Parameter-free analysis of digital surfaces with plane probing algorithms

Jui-Ting Lu

▶ **To cite this version:**

Jui-Ting Lu. Parameter-free analysis of digital surfaces with plane probing algorithms. Mathematics [math]. INSA de Lyon, 2023. English. NNT : 2023ISAL0123 . tel-04689444

HAL Id: tel-04689444
https://theses.hal.science/tel-04689444

Submitted on 5 Sep 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

N° d'ordre NNT : 2023ISAL0123

THÈSE de DOCTORAT DE L'UNIVERSITÉ DE LYON

opérée au sein de

l'Institut National des Sciences Appliquées de Lyon

École Doctorale ED512
École Doctorale InfoMaths

Spécialité de doctorat : Informatique

Soutenue publiquement le 21/12/2023, par :

Jui-Ting Lu

Parameter-free analysis of digital surfaces with plane probing algorithms

---

Devant le jury composé de :

M. Fabien Feschet	Professeur des universités	Université Clermont Auvergne	Rapporteur
Mme. Yukiko Kenmochi	Chargée de recherche	CNRS	Rapportrice
Mme. Isabelle Debled-Rennesson	Professeure des universités	Université de Lorraine	Examinatrice
Mme. Isabelle Sivignon	Chargée de recherche	CNRS	Examinatrice
M. David Coeurjolly	Directeur de recherche	CNRS	Directeur de thèse
M. Tristan Roussillon	Maître de conférence	INSA Lyon	Directeur de thèse
M. Jacques-Olivier Lachaud	Professeur des universités	Université Savoie Mont Blanc	Invité
M. Yan Gérard	Maitre de Conférences, HDR	Université Clermont Auvergne	Invité

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Département FEDORA – INSA Lyon - Ecoles Doctorales

SIGLE	ECOLE DOCTORALE	NOM ET COORDONNEES DU RESPONSABLE
CHIMIE	**CHIMIE DE LYON** **https://www.edchimie-lyon.fr** Sec. : Renée EL MELHEM Bât. Blaise PASCAL, 3e étage secretariat@edchimie-lyon.fr	**M. Stéphane DANIELE** C2P2-CPE LYON-UMR 5265 Bâtiment F308, BP 2077 43 Boulevard du 11 novembre 1918 69616 Villeurbanne **directeur@edchimie-lyon.fr**
E.E.A.	**ÉLECTRONIQUE, ÉLECTROTECHNIQUE, AUTOMATIQUE** **https://edeea.universite-lyon.fr** Sec. : Stéphanie CAUVIN Bâtiment Direction INSA Lyon Tél : 04.72.43.71.70 secretariat.edeea@insa-lyon.fr	**M. Philippe DELACHARTRE** INSA LYON Laboratoire CREATIS Bâtiment Blaise Pascal, 7 avenue Jean Capelle 69621 Villeurbanne CEDEX Tél : 04.72.43.88.63 **philippe.delachartre@insa-lyon.fr**
E2M2	**ÉVOLUTION, ÉCOSYSTÈME, MICROBIOLOGIE, MODÉLISATION** **http://e2m2.universite-lyon.fr** Sec. : Bénédicte LANZA Bât. Atrium, UCB Lyon 1 Tél : 04.72.44.83.62 secretariat.e2m2@univ-lyon1.fr	**Mme Sandrine CHARLES** Université Claude Bernard Lyon 1 UFR Biosciences Bâtiment Mendel 43, boulevard du 11 Novembre 1918 69622 Villeurbanne CEDEX **sandrine.charles@univ-lyon1.fr**
EDISS	**INTERDISCIPLINAIRE SCIENCES-SANTÉ** **http://ediss.universite-lyon.fr** Sec. : Bénédicte LANZA Bât. Atrium, UCB Lyon 1 Tél : 04.72.44.83.62 secretariat.ediss@univ-lyon1.fr	**Mme Sylvie RICARD-BLUM** Institut de Chimie et Biochimie Moléculaires et Supramoléculaires (ICBMS) - UMR 5246 CNRS - Université Lyon 1 Bâtiment Raulin - 2ème étage Nord 43 Boulevard du 11 novembre 1918 69622 Villeurbanne Cedex Tél : +33(0)4 72 44 82 32 **sylvie.ricard-blum@univ-lyon1.fr**
INFOMATHS	**INFORMATIQUE ET MATHÉMATIQUES** **http://edinfomaths.universite-lyon.fr** Sec. : Renée EL MELHEM Bât. Blaise PASCAL, 3e étage Tél : 04.72.43.80.46 infomaths@univ-lyon1.fr	**M. Hamamache KHEDDOUCI** Université Claude Bernard Lyon 1 Bât. Nautibus 43, Boulevard du 11 novembre 1918 69 622 Villeurbanne Cedex France Tél : 04.72.44.83.69 **hamamache.kheddouci@univ-lyon1.fr**
Matériaux	**MATÉRIAUX DE LYON** **http://ed34.universite-lyon.fr** Sec. : Yann DE ORDENANA Tél : 04.72.18.62.44 yann.de-ordenana@ec-lyon.fr	**M. Stéphane BENAYOUN** Ecole Centrale de Lyon Laboratoire LTDS 36 avenue Guy de Collongue 69134 Ecully CEDEX Tél : 04.72.18.64.37 **stephane.benayoun@ec-lyon.fr**
MEGA	**MÉCANIQUE, ÉNERGÉTIQUE, GÉNIE CIVIL, ACOUSTIQUE** **http://edmega.universite-lyon.fr** Sec. : Stéphanie CAUVIN Tél : 04.72.43.71.70 Bâtiment Direction INSA Lyon mega@insa-lyon.fr	**M. Jocelyn BONJOUR** INSA Lyon Laboratoire CETHIL Bâtiment Sadi-Carnot 9, rue de la Physique 69621 Villeurbanne CEDEX **jocelyn.bonjour@insa-lyon.fr**
ScSo	**ScSo*** **https://edsciencessociales.universite-lyon.fr** Sec. : Mélina FAVETON INSA : J.Y. TOUSSAINT Tél : 04.78.69.77.79 melina.faveton@univ-lyon2.fr	**M. Christian MONTES** Université Lumière Lyon 2 86 Rue Pasteur 69365 Lyon CEDEX 07 **christian.montes@univ-lyon2.fr**

*ScSo : Histoire, Géographie, Aménagement, Urbanisme, Archéologie, Science politique, Sociologie, Anthropologie

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Résumé

Les volumes 3D discrets proviennent de diverses sources, notamment la segmentation d'images, la simulation numérique, et les éditeurs basés sur les voxels. Notre intérêt réside dans le traitement de la géométrie des surfaces discrètes entourant ces volumes, permettant la reconnaissance de structures locales telles que des segments de plans discrets. Cependant, les surfaces discrètes ont une géométrie pauvre, composée de surfels carrés parallèles aux axes. Pour analyser ces surfaces, des algorithmes de type plane-probing adaptent le voisinage autour d'un point en développant itérativement une approximation de plan, souvent sous forme de triangles, en fonction des informations locales. Notre objectif est d'analyser ces surfaces discrètes en utilisant les méthodes de type plane-probing.

Nous introduisons les algorithmes de type plane-probing existants dans un cadre général. De plus, nous proposons une nouvelle variante de l'algorithme de type plane-probing qui prend en compte un voisinage plus étendu que ceux des algorithmes existants. Nous proposons également une implémentation efficace de cette nouvelle variante.

Une découverte importante est que la suite de tétraèdres formée à partir de deux triangles consécutifs crée une triangulation de Delaunay dans une partie du plan discret. Cette propriété est vérifiée pour la nouvelle variante introduite. En conséquence, le triangle final retourné par l'algorithme a trois angles aigus ou droits. Ce résultat nous permet de déterminer l'étendue du voisinage considéré au cours des calculs.

Enfin, nous proposons quelques ajustements afin d'adapter ce type d'algorithme à des surfaces discrètes, permettant ainsi de déduire un estimateur de vecteurs normaux. Nous nous concentrons notamment sur la convergence multigrille de cet estimateur, qui a été observée expérimentalement pour des positions bien identifiées sur des surfaces discrètes convexes.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Abstract

Discrete 3D volumes come from various sources, including image segmentation, numerical simulation, and voxel-based editors. Our focus is on processing the geometries of digital surfaces surrounding these volumes, enabling the recognition of local structures such as pieces of digital plane. Data is manipulated without alteration, making it easy to use octrees for efficient spatial representation. However, digital surfaces have an indigent geometry, consisting of square surfels parallel to the axes. To analyze these surfaces, plane-probing algorithms adapt the neighborhood around a point by iteratively developing a plane approximation, often in the form of triangles, based on local information. Our goal is to analyze these discrete surfaces using plane-probing methods.

In each iteration, these plane-probing algorithms select a point in a specific neighborhood and update one of the vertices of the currently constructed triangle. In this context, we introduce existing plane-probing algorithms in a general framework. Additionally, we propose a new variant of the plane-probing algorithm that considers a larger neighborhood than existing algorithms. We have also made improvements to the implementation of this new variant.

An important discovery is that the sequence of tetrahedra formed from two consecutive triangles creates a Delaunay triangulation in a part of the digital plane. This property holds for the introduced new variant. As a result, the final triangle of the algorithm is always acute, making it easier to find the minimal base. These results lead us to discussions about locality, helping us estimate the space required for the algorithm.

Finally, we suggest some adjustments to adapt this type of algorithm to discrete surfaces, thereby deducing an estimator of normal vectors. We particularly focus on the multigrid convergence of this estimator, which has been experimentally observed for well-defined positions on convex discrete surfaces.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Remerciements / Acknowledgments / 致謝

Mes plus sincères remerciements vont à David et Tristan pour leur encadrement exceptionnel et leur expertise inestimable tout au long de ces trois années de thèse. Je les remercie également pour leur patience et leur soutien lors des moments difficiles.

Grâce à l'encadrement de Julie et Raphaëlle lors d'un stage, j'ai pu découvrir le laboratoire LIRIS. Je suis reconnaissante envers les autres membres de l'équipe ORIGAMI, permanents, doctorants, stagiaires, notamment Léon, Olivier, Pierre, Thomas, Tong, pour les pauses bien nécessaires, les rires partagés et les liens d'amitié qui ont rendu cette expérience d'études doctorales encore plus mémorable.

J'aimerais remercier tous les chercheurs que j'ai pu rencontrer lors des conférences et séminaires. J'aimerais remercier notamment Bastien, Jacques-Olivier, Laurent, et Phuc pour les échanges intéressants et enrichissants pour la thèse.

Je tiens à remercier Stéphanie et Yan pour leur accompagnement lors des comités de suivi de la thèse. Je souhaite exprimer également ma profonde reconnaissance aux rapporteurs, Fabien et Yukiko pour leur temps et leurs commentaires constructifs et leur précieuse contribution à l'amélioration de cette thèse. J'aimerais aussi exprimer ma reconnaissance aux membres du jury Isabelle D.-R., Isabelle S. pour leur présence à ma soutenance.

Je tiens à exprimer ma gratitude envers l'administration de l'INSA Lyon pour son assistance dans la préparation de la soutenance. De plus, j'aimerais remercier Aude, Audrey, Joseph, Julie, Laura, Nicolas, Oriane, Shai-Ing, Thierry et Yang, qui m'ont énormément aidée à m'intégrer à la vie à Lyon.

Pendant cette thèse, j'ai eu l'opportunité d'effectuer un stage de recherche au Japon. I would like to thank Prof. Sugimoto for his guidance during the six months in Japan. I would also like to thank Antoine, Chang, Chung Yun, Domenico, Ghali, Hajar, Laura, Liang, Marta, Sayaka, Samuel, Sou, Vittorio for the unforgettable life experience in Japan.

Enfin, j'aimerais remercier les membres de ma famille. 感謝我的父母及其他長輩，不論血緣與否，還有最最最親的姊妹。Ti^{7}-kong, tsap^{8} ni^{5} u^{7} li^{2} sann-thin^{7}, gua^{2} tsiok ho^{2}-mia^{7}. To-sia^{7}–li^{2}!

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Contents

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

CHAPTER 1

Introduction

Diverse sources give rise to 3D discrete volumes, encompassing image segmentation techniques employed in materical science [Fli+05], medical imaging [Hil+99], simulations replicating physical processes [Mar+10], voxel-based editing tools,and various other sources. Efficient octree-type spatial data structures facilitate the manipulation of this data. However, many high-level tasks in computer graphics, computer vision and 3D image analysis require a rich geometry. Our focus lies in processing the geometry of digital surfaces that surround a set of voxels. This allows us to recognize local structures, such as segments of digital planes, or estimate differential quantities. A drawback of working with a digital surface is the inherent geometric limitation, as, at any resolution, it comprises square surface elements that are parallel to one of the axes.

To overcome this while analyzing digital surfaces, it is often necessary to capture local geometric properties surrounding a specific point. This can be achieved by either considering a fixed neighborhood (e.g., employing a Euclidean ball with a predetermined radius) or by adapting this neighborhood based on local geometric characteristics. When the depicted neighborhood is too small, we lack sufficient information to generate meaningful results. Conversely, an excessively large neighborhood could lead to the loss of fine details in the digital surface. Thus the choice of the neighborhood is very important.

In many methods, the size of the neighborhood is a user-defined parameter, but here we focus on parameter-free methods. In particular, we focus on the plane-probing algorithms, which iteratively construct a plane approximation using prescribed updating rules. What makes plane-probing algorithms promising is their ability to dynamically determine the most effective way to probe the digital surface. The algorithm achieves this by extending a local pattern, typically in the form of a triangle, from a point of interest so that it aligns with an estimated tangent plane.

In this thesis, our objective is to investigate various geometric properties of the plane-probing algorithms. We aspire to enhance the estimation of first-order geometric quantities associated with the algorithm through the development of these theoretical findings. We implement estimators for normal vectors based on plane-probing

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

algorithms, eliminating the need for user-defined parameters. Additionally, we seek for precision improvement when the resolution is sufficiently increased.

PARameter-free Analysis of DIgital Surfaces This thesis is funded by the French *Agence Nationale de la Recherche* with the project, PARameter-free Analysis of DIgital Surfaces (PARADIS).

Organization of the manuscrit In this thesis, we present an exploration of the plane-probing algorithms and their associated properties. The organization of this work is structured into six chapters. Chapter 2 establishes the foundational knowledge by reviewing the state of the art and introducing essential notations. In chapter 3, we introduce our novel algorithm variant, providing insights into its practical implementation and performance. Chapter 4 narrows our focus to a critical invariant of the L-algorithm, namely, the Delaunay property. Chapter 5 is dedicated to a detailed examination of various aspects related to the locality of our algorithm. Finally, in chapter 6, we present empirical results and discuss the multigrid convergence of our algorithm.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

CHAPTER 2

Preliminary

In this chapter, we first explain why plane-probing algorithms have emerged as a potential tool in the context of digital surface analysis. Then, we present *tetrahedron-based* plane-probing algorithms in a uniform way and, in particular, prove their termination and correctness on digital planes. Based on those fundamental elements, our contributions will be presented in the next chapters.

2.1 Digital volume and surface analysis

In the field of digital geometry, we explore the digitized version of continuous objects as discrete sets, much like pixels on a screen. The research does not limit to 2D but also nD. Given a Euclidean space, a *digital volume* is a set of *volume elements*, i.e., *voxels*. The concept of a voxel can be visualized as small cubes, then a digital volume can be perceived as a collection of identical small cubes that share the sames axes of direction [KR04] (see Fig. 2.1).

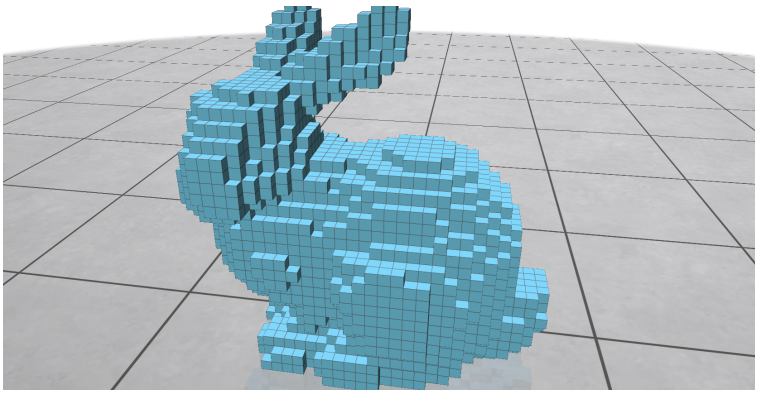


FIGURE 2.1: Example of a digital volume: a digital bunny

Non-invasive acquisition tools such as magnetic resonance, X-ray tomography or micro-tomography are essential for observation and measurements in various fields,

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

for instance, in material sciences [Fli+05] or medical imaging [Hil+99]. These devices typically produce 3D images. Digital volumes come from the segmentation of such 3D images. They are also generated in voxel-based editing tools and in scientific modeling [Dar+09; Jon+10; Mar+10]. Indeed, digital volumes offer advantages: regularity of the support, existence of efficient octree-based spatial data structures [KSA13; VMG17], integer-only and exact computations, etc.

We consider a closed *digital surface* as boundary of a digital volume [RKW91]. A digital surface consists of *surface elements*, i.e., *surfels*, and can be visually interpreted as a quadrangular mesh with squares.

A drawback of working with a digital surface is its inherent geometric limitation, as, at any resolution, it comprises square surface elements that are parallel to one of the axes, while many high-level tasks in computer graphics, computer vision and 3D image analysis require an accurate normal vector field. The context of this thesis is the estimation of such normal vector field from a digital surface, when considering it as the digitization of an unknown continuous surface. The accuracy of the estimates is expected to improve as the digitization step becomes increasingly finer. This is the idea of *multigrid convergence* [Kle00; CLR12]. In the next section, we review the most common methods for normal estimation not only on digital surfaces, but also on meshes and point clouds in order to show why plane-probing algorithms have emerged as a potential tool in this context.

2.2 Normal estimation

There are only six potential directions for the normal of a surfel, including the three axes and their opposites. In order to obtain a more relevant direction, it is necessary to aggregate the geometrical information contained in a larger surface patch. This can be achieved by either considering a fixed neighborhood (sec. 2.2.1) or by adapting this neighborhood based on local geometric characteristics (sec. 2.2.2).

2.2.1 Kernel methods

The local geometric information can be aggregated in various ways.

Fitting. [Hop+92] devised a method for estimating normal vectors at specific data points. This involved computing the least squares best-fitting plane using a set of points from the local neighborhood surrounding the point of interest. Various other fitting surfaces have also been explored, such as jets, which are essentially truncated Taylor expansions of surface expressions [CP05; CP08]. In the initial stages of all fitting methods, the essential step involves gathering the points to be used in the fitting process. In the case of meshes, a breadth-first search is employed to visit neighboring points until a sufficient set of points is collected. Conversely, in point-cloud scenarios, the k-nearest-neighbors strategy is commonly employed. In both situations, the number of points to collect typically remains a user-defined parameter, despite the existence of some proposed heuristics for automatic selection [Hop+92; CP05]. All fitting methods tend to smooth sharp features, and thus fail to correctly estimate normal vectors near edges.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Voronoi diagram. Instead of approximating the tangent space, an alternative approach relies on Voronoi diagrams to provide a more accurate estimation of the orthogonal space. [AB99] first used the furthest vertex of the Voronoi cell to estimate the normal vectors for point clouds. To enhance the stability of these estimations, [All+07] proposed to apply linear fitting to the Voronoi cell or to the union of Voronoi cells within a neighborhood. In further advancements, [MOG11] proposed the Voronoi Covariance Measure (VCM), that computes a weighted average of covariance matrices derived from Voronoi cells, as opposed to calculating the covariance matrix of their union. In their method, only the intersection between the Voronoi diagram and a ball around the data point is taken into account in order to get purely local information about the surface geometry. A digital variant was proposed in [Cue+15].

Integral invariants. In mesh-based approaches, an alternative method involves aggregating surface geometry within a spherical region. This is accomplished by computing integrals over the intersection between the sphere and the volume bounded by the mesh [Pot+09]. The covariance matrix of this intersection set serves as a means to estimate principal curvatures, principal directions, and the normal direction. There exists also a digital adaptation [LCL17]. The choice of the ball radius represents a user-defined parameter in these methods. In addition, the ball radius is usually the same for the whole digital surface and completely ignores the features. The above-mentioned digital variants [Cue+15; LCL17] are multigrid-convergent for digitization of smooth shapes if the radius is conveniently chosen with respect to the grid step.

Convolution. Starting from initial normal vector estimates (e.g., unit vectors perpendicular to surfels), several methods refine these estimates by applying convolution using a specified smoothing kernel. This technique has been employed in both digital curves [EMC11; EM16] and surfaces [FM09]. However, it comes with certain limitations: The user must specify the kernel parameters, including its size, and the treatment, which is usually isotropic, smooths out sharp features. However, if the kernel size is some function of the grid step, multigrid convergence can be obtained in two dimensions [EMC11]. In mesh denoising, more advanced local filters, such as the bilateral one, are used to preserve sharp edges and corners. However, these filters typically involve several user-defined parameters [FDC03; Zhe+11].

2.2.2 Geometric methods

The methods described above aggegrate the geometrical information within a neighborhood of predetermined size. It is crucial to wisely select that size and find the right balance between obtaining accurate estimates and preserving sharp features. Indeed, when the neighborhood is too small, we lack sufficient information to generate meaningful results. Conversely, an excessively large neighborhood could lead to lose fine details.

In digital geometry, researchers try to process the geometry in another way. The neighborhoods are defined from a primitive so that their shape and size depend on the local geometry. The primitives are typically segments of digital lines or planes.

Digital straight segment (DSS) The set of maximal DSSs, i.e., inextensible straight parts, can be computed by one scan of the digital curve [FT99; Fes05] and provides a multigrid-convergent normal estimator [LVV07]. In addition to convergence, this normal estimator has several sought-after properties: (i) it preserves sharp

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

features (ii) local convexity [RS11], and (iii) without any input parameter. Asymptotic properties of maximal DSSs can also be used to estimate the local amount of noise along the digital curve [KL12]. A last benefit is that normal integration over the digital curve provides a multigrid-convergent length estimator [CK04].

Maximal segments on 2D slices. In higher dimension, to the best of our knowledge, only one parameter-free normal estimator has been proposed in 3D [TD99] and extended to nD [LV03]. It is based on maximal DSSs on 2D slices. Maximal DSSs provide windows of adaptive size but the slicing truncates the geometric information and leads to an artificial spatial variability because two neighbor surfels only share one slice over two.

Digital plane segments (DPS) Another possible approach is to mimic the 2D tool box by computing the set of DPSs that locally fit the digital surface. This approach has been used for surface area estimation [KS01], reversible polyhedrization [SDC04] and normal estimation [CL11]. In the latter approach, DPSs are initiated by a maximal circular neighborhood around a seed, with no adjustments made to their normal vector during extension. While this constitutes a consistent definition of maximality, the approach struggles to recovers the geometry of the digital surface near sharp edges and corners. The primary challenge lies in developing an efficient scanning technique to identify DPSs that can effectively reveal the local geometry through their size and shape.

Recognition of digital plane segments Recognizing digital planes is a way to ensure the validity of a digital plane segment. Researchers initially addressed this challenge in finite scenarios, such as identifying digital planes within digital cubes [KS91]. The authors of [GDZ05] treated this problem as a linear programming issue and proposed an algorithm with high theoretical complexity but apparent linear time execution concerning the input size of integer points. In the field of digital plane recognition, [DR94] introduced the concepts of "leaning points" and "leaning planes" and provide an algorithm that recognize a piece of digital plane around a given point. Simultaneously, other research efforts have contributed algorithms emphasizing computational efficiency [Buz03]. Some researchers have also explored the fitting of planes between two sets of points, rather than examining the entire point set [Vee12].

2.2.3 Prior research on plane-probing algorithms

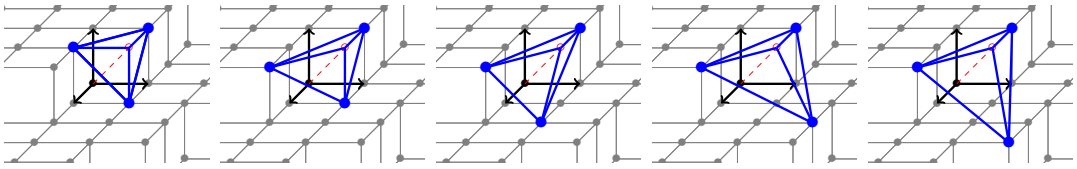
The main challenge is not really to recognize DPSs, but more to find which data points should be taken into account during the recognition process to obtain DPSs tangent to the digital surface. In this thesis, we mainly focus on plane-probing algorithms that allow to decide on-the-fly how to probe the digital surface and make grow an implicit DPS tangent by construction. The growth direction is given by both arithmetic and geometric properties.

Plane-probing algorithms can mainly be categorized into two types: *tetrahedron-based* plane-probing algorithms [LPR16a; LPR16b; LPR17; RL19] and *parallelepiped-based* plane-probing algorithms [LMR20].

We compare side by side the *tetrahedron-based* plane-probing algorithms (see Fig. 2.2) and the *parallelepiped-based* plane-probing algorithms (see Fig. 2.3). For example, we consider the normal $\mathbf{N} = (1, 2, 5)$. Both algorithms return the same normal at the end

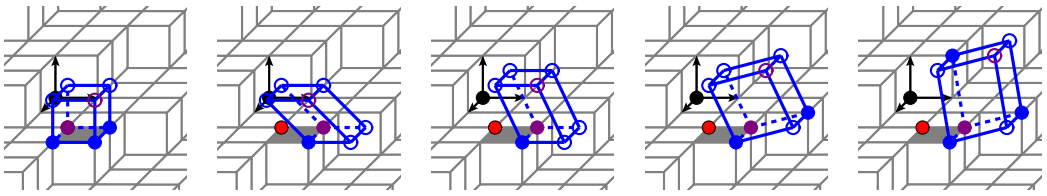
Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

of iteration, but have very different behavior. *Tetrahedron-based* plane-probing algorithms always have its base triangle inside the digital plane, whereas *parallelepiped-based* plane-probing algorithms shows a parallelepiped that strolls between the border.



(A) $i = 0$ (B) $i = 1$ (C) $i = 2$ (D) $i = 3$ (E) $i = 4$

FIGURE 2.2: The evolution (from left to right) of a *tetrahedron-based* plane-probing algorithm on a digital plane of normal $(1, 2, 5)$.



(A) $i = 0$ (B) $i = 1$ (C) $i = 2$ (D) $i = 3$ (E) $i = 4$

FIGURE 2.3: The evolution (from left to right) of a *parallelepiped-based* plane-probing algorithm on a digital plane of normal $(1, 2, 5)$.

Parallelepiped-based plane-probing algorithms are essentially wrappers around *tetrahedron-based* plane-probing algorithms and will be described in chapter 6. Except in [LPR16a], *tetrahedron-based* plane-probing algorithms are local by construction, because they update the three vertices of the tetrahedron base until it matches the normal of the digital plane, while the apex of the tetrahedron remains fixed. That is why we will focus on this kind of algorithms in the upcoming section.

2.3 Plane-probing algorithms

A *digital plane* refers to an infinite digital set that can be defined using a normal vector $\mathbf{N} \in \mathbb{Z}^3 \setminus \{\mathbf{0}\}$ and a shift value $\mu \in \mathbb{Z}$ as follows [Rev91]:

$$\mathbf{P}_{\mu,\mathbf{N}} := \{\mathbf{x} \in \mathbb{Z}^3 \mid \mu \leq \mathbf{x} \cdot \mathbf{N} < \mu + \|\mathbf{N}\|_1\}. \tag{2.1}$$

To put it differently, it comprises all integer points that lie between two parallel, infinite continuous planes. Fig. 2.4 represents an illustration of a digital plane.

In this thesis, when we mention any digital plane, we suppose w.l.o.g. that $\mu = 0$ and that the components of $\mathbf{N}$ are positive, i.e., $\mathbf{N} \in \mathbb{N}^3 \setminus \{\mathbf{0}\}$:

$$\mathbf{P}_{\mathbf{N}} := \{\mathbf{x} \in \mathbb{Z}^3 \mid 0 < \mathbf{x} \cdot \mathbf{N} < \|\mathbf{N}\|_1\}. \tag{2.2}$$

In this section, we focus on *tetrahedron-based* plane-probing algorithms applied on a digital plane [LPR16b; LPR17; RL19]. Given a digital plane $\mathbf{P} \in \{\mathbf{P}_{\mathbf{N}} \mid \mathbf{N} \in \mathbb{N}^3 \setminus \{\mathbf{0}\}\}$ of unknown normal vector, a *plane-probing algorithm* computes the normal vector $\mathbf{N}$ of $\mathbf{P}$ by sparsely probing it with the predicate "is $\mathbf{x}$ in $\mathbf{P}$?", denoted by InPlane($\mathbf{x}$) hereafter.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

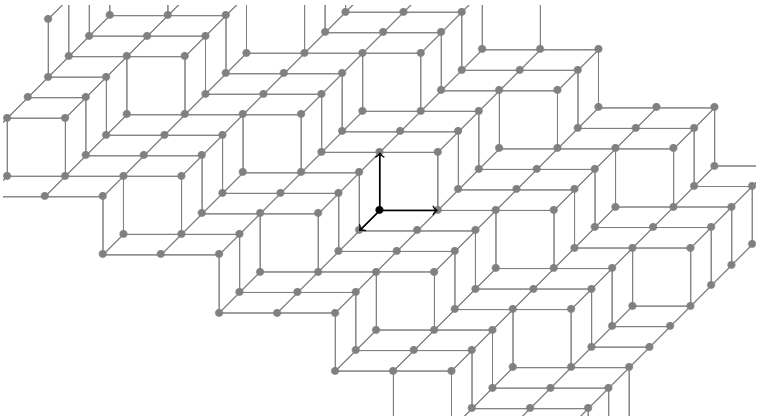


FIGURE 2.4: A segment of a digital plane

In the following, we describe a general framework for *tetrahedron-based* plane-probing algorithms (see also Algorithm 1).

2.3.1 Initialization

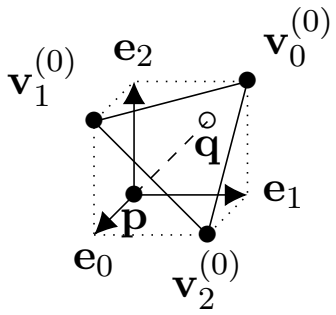


FIGURE 2.5: Illustration of the starting triangle

Let $(\mathbf{e}_0, \mathbf{e}_1, \mathbf{e}_2)$ be the canonical basis of $\mathbb{Z}^3$. Given a starting point $\mathbf{p}$ that belongs to the plane $\mathbf{P}$, let $\mathbf{q}$ be equal to $\mathbf{p} + (1, 1, 1)$. The point $\mathbf{q}$ is by definition not in $\mathbf{P}$. We define the initial triangle as follow

$$\mathbf{T}^{(0)} := (\mathbf{v}_k^{(0)})_{k\in\{0,1,2\}} = (\mathbf{q} - \mathbf{e}_k)_{k\in\{0,1,2\}}. \tag{2.3}$$

We suppose that every vertex of the first triangle is in the digital plane, i.e. $\mathbf{T}^{(0)} \subset \mathbf{P}$ (see Fig. 2.5 and Algorithm 1, line 1).

2.3.2 Neighborhoods

At every step $i \in \mathbb{N}$, the triangle $\mathbf{T}^{(i)} \subset \mathbf{P}$ represents the current approximation of the plane $\mathbf{P}$. One vertex of $\mathbf{T}^{(i)} = (\mathbf{v}_k^{(i)})_{k\in\{0,1,2\}}$ is updated per iteration. That vertex is replaced by a point of $\mathbf{P}$ from a candidate set, also called *neighborhood*.

We first define three sets of integer pairs as follow:

$$S_H := \{(\alpha, \beta) \in \{(1, 0), (0, 1)\}. \tag{2.4}$$

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

$$S_R := \{(\alpha, \beta) \in \{(1, \lambda), (\lambda, 1) \mid \lambda \in \mathbb{N}\}. \tag{2.5}$$

$$S_L := \{(\alpha, \beta) \in \mathbb{N}^2 \setminus (0, 0)\}. \tag{2.6}$$

Let S be one of the three sets above. At every step i, the neighborhood is now defined as follows:

$$\mathcal{N}_S^{(i)} := \left\{ \mathbf{v}_k^{(i)} + \alpha \mathbf{m}_{k+1}^{(i)} + \beta \mathbf{m}_{k+2}^{(i)} \mid k \in \mathbb{Z}/3\mathbb{Z},\ (\alpha, \beta) \in S \right\}, \tag{2.7}$$

where $\forall i, k, \quad \mathbf{m}_k^{(i)} := \mathbf{q} - \mathbf{v}_k^{(i)}$.

The point $\mathbf{q}$ is fixed, and we note $\mathbf{p}^{(i)} = \mathbf{q} - \sum_{k \in \{0,1,2\}} \mathbf{m}_k^{(i)}$. We drop the index i when there is no ambiguity.

We denote respectively $\mathcal{N}_{S_H}^{(i)}, \mathcal{N}_{S_R}^{(i)}$, and $\mathcal{N}_{S_L}^{(i)}$ the neighborhood associated with the sets S_H, S_R and S_L. See Fig. 2.6 for an illustration of the neighborhoods. The H-algorithm is based on $\mathcal{N}_{S_H}^{(i)}$, which looks like an Hexagon, whereas the R-algorithm is based on $\mathcal{N}_{S_R}^{(i)}$, which consists of Rays. In this paper, we propose a lattice-based algorithm, denoted by the letter L, for lattice, and which uses the largest neighborhood $\mathcal{N}_{S_L}^{(i)}$. We indeed have $S_H \subset S_R \subset S_L$ and $\forall i,\ \mathcal{N}_{S_H}^{(i)} \subset \mathcal{N}_{S_R}^{(i)} \subset \mathcal{N}_{S_L}^{(i)}$.

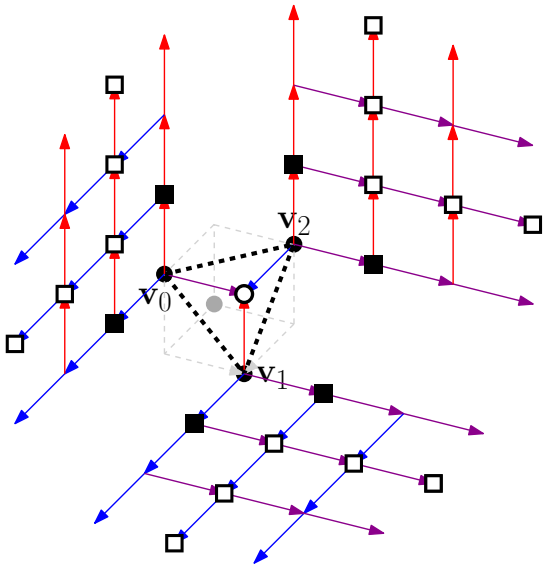


FIGURE 2.6: Illustration of the neighborhoods. $\mathcal{N}_{S_H}^{(i)}$ (black square), $\mathcal{N}_{S_R}^{(i)}$ (squares) and $\mathcal{N}_{S_L}^{(i)}$ includes every point on the lattices.

We discuss more about neighborhood in Sec. 3.1

The algorithm terminates at a step n, when the neighborhood has an empty intersection with the plane, i.e., when $\mathcal{N}_S^{(n)} \bigcap \mathbf{P} = \emptyset$ (Algorithm 1, line 3). The proof of termination is postponed to Sec. 2.5.

2.3.3 Pre-order

At every step i, let $\mathcal{H}_+^{(i)}$ be the open half-space delimited by $\mathbf{T}^{(i)}$ and containing the neighborhood $\mathcal{N}_S^{(i)}$. In addition, let $\mathcal{B}(\mathbf{T}, \mathbf{x})$ be the closed ball defined by $\mathbf{T}^{(i)}$ and a

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

fourth point $\mathbf{x}$ not in the plane passing by $\mathbf{T}^{(i)}$.

As in [RL19], for any pair $\mathbf{x}, \mathbf{x}' \in \mathcal{H}_+^{(i)}$, we say that $\mathbf{x}'$ is *closer* to $\mathbf{T}^{(i)}$ than $\mathbf{x}$, denoted $\mathbf{x}' \leq_{\mathbf{T}^{(i)}} \mathbf{x}$, if and only if we have the inclusion: $(\mathcal{B}(\mathbf{T}^{(i)}, \mathbf{x}') \cap \mathcal{H}_+^{(i)}) \subseteq (\mathcal{B}(\mathbf{T}^{(i)}, \mathbf{x}) \cap \mathcal{H}_+^{(i)})$ (See Fig. 2.7). By examining the link between the relation $\leq_{\mathbf{T}^{(i)}}$ and the intersection of balls with half-spaces, we obtain the following claim:

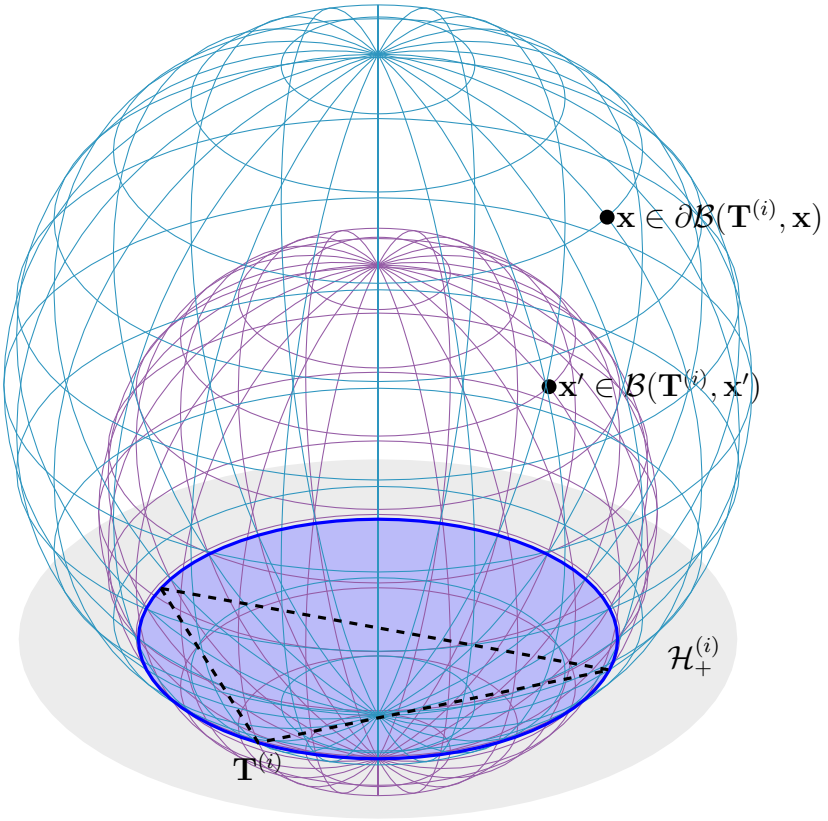


FIGURE 2.7: Illustration of $\mathcal{B}(\mathbf{T}^{(i)}, \mathbf{x})$ and $\mathcal{B}(\mathbf{T}^{(i)}, \mathbf{x}')$ for two points $\mathbf{x}, \mathbf{x}' \in \mathcal{H}_+^{(i)}$.

Remark 2.1. *For all $i \in \{0, \ldots, n\}$, $\leq_{\mathbf{T}^{(i)}}$ is a total preorder on $\mathcal{H}_+^{(i)}$.*

Proof. Here, we focus on a step $i \in \{0, \ldots, n\}$ and ignore the exponent (i) in the notations. For any pair $\mathbf{x}, \mathbf{x}' \subset \mathcal{H}_+$, we remind that $\mathbf{x}' \leq_{\mathbf{T}} \mathbf{x}$ if and only if $(\mathcal{B}(\mathbf{T}, \mathbf{x}') \cap \mathcal{H}_+) \subseteq (\mathcal{B}(\mathbf{T}, \mathbf{x}) \cap \mathcal{H}_+)$.

- **Reflexivity**: the ball defined by $\mathbf{T}$ and $\mathbf{x} \in \mathcal{H}_+$ is unique, thus $(\mathcal{B}(\mathbf{T}, \mathbf{x}) \cap \mathcal{H}_+) \subseteq (\mathcal{B}(\mathbf{T}, \mathbf{x}) \cap \mathcal{H}_+)$.
- **Transitivity**: it is induced from the transitivity of the order $\subseteq$.
- **Totality**: $\leq_{\mathbf{T}}$ is total if $(\mathcal{B}(\mathbf{T}, \mathbf{x}') \cap \mathcal{H}_+) \subseteq (\mathcal{B}(\mathbf{T}, \mathbf{x}) \cap \mathcal{H}_+)$ or $(\mathcal{B}(\mathbf{T}, \mathbf{x}) \cap \mathcal{H}_+) \subseteq (\mathcal{B}(\mathbf{T}, \mathbf{x}') \cap \mathcal{H}_+)$ for all $\mathbf{x}, \mathbf{x}' \in \mathcal{H}_+$. We have two cases according to the pair $\mathbf{x}, \mathbf{x}'$:
  - if $\mathcal{B}(\mathbf{T}, \mathbf{x}) = \mathcal{B}(\mathbf{T}, \mathbf{x}')$, both alternatives are obviously true.
  - if $\mathcal{B}(\mathbf{T}, \mathbf{x}) \neq \mathcal{B}(\mathbf{T}, \mathbf{x}')$, the intersection of the boundary of the two balls is a curve lying in a plane, which is by definition, the one containing $\mathbf{T}$. Since $\mathcal{H}_+$ does not contain that plane, we necessarily have either $(\mathcal{B}(\mathbf{T}, \mathbf{x}') \cap \mathcal{H}_+) \subseteq (\mathcal{B}(\mathbf{T}, \mathbf{x}) \cap \mathcal{H}_+)$ or $(\mathcal{B}(\mathbf{T}, \mathbf{x}) \cap \mathcal{H}_+) \subseteq (\mathcal{B}(\mathbf{T}, \mathbf{x}') \cap \mathcal{H}_+)$.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

□

Remark 2.2. *The order is* not antisymmetric *because there exist co-spherical cases where* $\mathbf{x} \neq \mathbf{x}'$ *but* $\mathcal{B}(\mathbf{T}, \mathbf{x}) = \mathcal{B}(\mathbf{T}, \mathbf{x}')$.

2.3.4 Update rule

At each iteration, the algorithm replaces a vertex of $\mathbf{T}^{(i)}$ with a point of the set $\mathcal{N}_S^{(i)} \bigcap \mathbf{P}$ that is a closest one according to $\leq_{\mathbf{T}^{(i)}}$. More precisely, if the set $\mathcal{N}_S^{(i)} \bigcap \mathbf{P}$ is not empty, there is at least an index $k \in \{0, 1, 2\}$ and a pair of integers $(\alpha, \beta) \in \mathbb{N}^2 \setminus (0, 0)$ that satisfies the following relation:

$$\forall \mathbf{x} \in \mathcal{N}_S^{(i)} \cap \mathbf{P},\ \mathbf{v}_k^{(i)} + \alpha \mathbf{m}_{k+1}^{(i)} + \beta \mathbf{m}_{k+2}^{(i)} \leq_{\mathbf{T}} \mathbf{x}. \tag{2.8}$$

Remark 2.3. *The triple* (k, α, β) *may not be unique when several points are cospherical.*

The update rule is then [LPR17, Lemma 2]:

$$\begin{cases} \mathbf{v}_k^{(i+1)} := \mathbf{v}_k^{(i)} + \alpha \mathbf{m}_{k+1}^{(i)} + \beta \mathbf{m}_{k+2}^{(i)}, \\ \mathbf{v}_{k+1}^{(i+1)} := \mathbf{v}_{k+1}^{(i)}, \\ \mathbf{v}_{k+2}^{(i+1)} := \mathbf{v}_{k+2}^{(i)}. \end{cases} \tag{2.9}$$

As shown in Algorithm 1, lines 4 to 6, equations (2.8) and (2.9) are used to update the current triangle.

Algorithm 1: Plane-probing algorithms H, R ([LPR17]) and L[LRC22]

Input: The predicate InPlane := "Is a point $\mathbf{x} \in \mathbf{P}$?", a point $\mathbf{p} \in \mathbf{P}$, the type of neighborhood $S \in \{S_H, S_R, S_L\}$ (see equations (2.7))

Output: A normal vector $\hat{\mathbf{N}}$ and a basis of the lattice $\{\mathbf{x} \mid \mathbf{x} \cdot \hat{\mathbf{N}} = \|\mathbf{N}\|_1 - 1\}$.

1 $\mathbf{q} \leftarrow \mathbf{p} + (1, 1, 1)$; $(\mathbf{v}_k^{(0)})_{k \in \{0,1,2\}} \leftarrow (\mathbf{q} - \mathbf{e}_k)_{k \in \{0,1,2\}}$; `// initialization`

2 $i \leftarrow 0$;

3 **while** $\mathcal{N}_S^{(i)} \cap \{\mathbf{x} \mid \text{InPlane}(\mathbf{x})\} \neq \emptyset$ **do**

4 Let (k, α, β) be such that, for all $\mathbf{y} \in \mathcal{N}_S^{(i)} \cap \{\mathbf{x} \mid \text{InPlane}(\mathbf{x})\}$, $\mathbf{v}_k^{(i)} + \alpha(\mathbf{q} - \mathbf{v}_{k+1}^{(i)}) + \beta(\mathbf{q} - \mathbf{v}_{k+2}^{(i)}) \leq_{\mathbf{T}^{(i)}} \mathbf{y}$; `// equation (2.8)`

5 $\mathbf{v}_k^{(i+1)} \leftarrow \mathbf{v}_k^{(i)} + \alpha(\mathbf{q} - \mathbf{v}_{k+1}^{(i)}) + \beta(\mathbf{q} - \mathbf{v}_{k+2}^{(i)})$; `// equation (2.9)`

6 $\forall l \in \{0, 1, 2\} \setminus k,\ \mathbf{v}_l^{(i+1)} \leftarrow \mathbf{v}_l^{(i)}$;

7 $i \leftarrow i + 1$;

8 $B \leftarrow \{\mathbf{v}_0^{(i)} - \mathbf{v}_1^{(i)}, \mathbf{v}_1^{(i)} - \mathbf{v}_2^{(i)}, \mathbf{v}_2^{(i)} - \mathbf{v}_0^{(i)}\}$;

9 Let $\mathbf{b}_1$ and $\mathbf{b}_2$ be the shortest and second shortest vectors of B ;

10 **return** $\mathbf{b}_1 \times \mathbf{b}_2, (\mathbf{b}_1, \mathbf{b}_2)$; `// × denotes the cross product`

2.4 Invariant properties

Before proving that the algorithms terminate on a digital plane, we gather in this section some properties that the L-algorithm shares with its predecessors studied in

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

[LPR17]. These properties are invariant at every iteration and they are essential for the proof of termination in sec. 2.5.

Theorem 2.1 ([LPR17], Lemma 5)**.** *For all* $i \in \{0, \dots, n\}$, $\forall k \in \{0, 1, 2\}$, $\mathbf{m}_k \cdot \mathbf{N} > 0$.

Proof. For any $k \in \{0, 1, 2\}$ and $i \in \{0, \dots, n\}$, we recall the definition $\mathbf{m}_k^{(i)} = \mathbf{q} - \mathbf{v}_k^{(i)}$ in equation (2.7). We can conclude since $\mathbf{q}$ is not in the plane $\mathbf{P}$ ($\mathbf{q} \cdot \mathbf{N} \geq \|\mathbf{N}\|_1$) and $\mathbf{v}_k^{(i)}$ is in the plane $\mathbf{P}$ ($\mathbf{v}_k^{(i)} < \|\mathbf{N}\|_1$). □

Lemma 2.1 ([LPR17], Lemma 6)**.** *For all* $i \in \{0, \dots, n-1\}$*, let* $k^\star$ *be the updated vertex such that* $\mathbf{v}_{k^\star}^{(i+1)} \neq \mathbf{v}_{k^\star}^{(i)}$ *(2.9). Then* $\mathbf{v}_{k^\star}^{(i+1)} \cdot \mathbf{N} > \mathbf{v}_{k^\star}^{(i)} \cdot \mathbf{N}$*, i.e.,* $\mathbf{m}_{k^\star}^{(i+1)} \cdot \mathbf{N} < \mathbf{m}_{k^\star}^{(i)} \cdot \mathbf{N}$.

Proof. For all $i \in \{0, \dots, n-1\}$, by equation (2.9), only one vertex changes at each step, $\mathbf{v}_{k^\star}^{(i)}$. Then, we have $\mathbf{v}_{k^\star}^{(i+1)} = \mathbf{v}_{k^\star}^{(i)} + \alpha \mathbf{m}_{k^\star+1}^{(i)} + \beta \mathbf{m}_{k^\star+2}^{(i)}$, which is equivalent to $\mathbf{m}_{k^\star}^{(i+1)} = \mathbf{m}_{k^\star}^{(i)} - \alpha \mathbf{m}_{k^\star+1}^{(i)} - \beta \mathbf{m}_{k^\star+2}^{(i)}$, for some integers α, β. We can conclude with theorem 2.1. □

We say a point $\mathbf{x}$ is *higher* than another one $\mathbf{y}$ when $\mathbf{x} \cdot \mathbf{N} \geq \mathbf{y} \cdot \mathbf{N}$. Thus, the algorithm always replaces a vertex with a *higher* candidate point in direction $\mathbf{N}$. That property is a key point in the proof of Theorem 2.2. It also implies that the set $\mathcal{N}_S^{(i)} \cap \mathbf{P}$ is always finite.

Lemma 2.2. *For all* $i \in \{0, \dots, n\}$*, the set of candidate points* $\mathcal{N}_S^{(i)} \cap \mathbf{P}$ *is finite.*

Proof. The scalar product $(\mathbf{v}_k^{(i)} + \alpha(\mathbf{m}_{k+1}^{(i)}) + \beta(\mathbf{m}_{k+2}^{(i)})) \cdot \mathbf{N}$ tends to infinity when α or β (or both) tend to infinity. That is to say, when α or β is large enough, the point $\mathbf{v}_k^{(i)} + \alpha(\mathbf{m}_{k+1}^{(i)}) + \beta(\mathbf{m}_{k+2}^{(i)})$ does not belong to $\mathbf{P}$. Therefore, the set $\left\{ \mathbf{v}_k^{(i)} + \alpha \mathbf{m}_{k+1}^{(i)} + \beta \mathbf{m}_{k+2}^{(i)} \mid (\alpha, \beta) \in S \right\} \cap \mathbf{P}$ is finite for all $k \in \mathbb{Z}/3\mathbb{Z}$ and we can conclude. □

Theorem 2.2 ([LPR17], Lemma 3)**.** *For all* $i \in \{0, \dots, n\}$, $\det(\mathbf{m}_0^{(i)}, \mathbf{m}_1^{(i)}, \mathbf{m}_2^{(i)}) = 1$.

Proof. Let $\mathbf{M}^{(i)}$ be the 3×3 matrix formed by the vectors $\mathbf{m}_0^{(i)}$, $\mathbf{m}_1^{(i)}$, $\mathbf{m}_2^{(i)}$.We have $\det(\mathbf{M}^{(i)}) = \det(\mathbf{m}_0^{(i)}, \mathbf{m}_1^{(i)}, \mathbf{m}_2^{(i)})$ and in particular, $\det(\mathbf{M}^{(0)}) = \det(\mathbf{e}_0, \mathbf{e}_1, \mathbf{e}_2) = 1$. We now prove that if $\det(\mathbf{M}^{(i)}) = 1$ for $\forall i \in \{0, \dots, n-1\}$, then $\det(\mathbf{M}^{(i+1)}) = 1$. By equation (2.9), only one vertex changes at each step, let us say $\mathbf{v}_k^{(i)}$. Then, we have $\mathbf{v}_k^{(i+1)} = \mathbf{v}_k^{(i)} + \alpha \mathbf{m}_{k+1}^{(i)} + \beta \mathbf{m}_{k+2}^{(i)}$, which is equivalent to $\mathbf{m}_k^{(i+1)} = \mathbf{m}_k^{(i)} - \alpha \mathbf{m}_{k+1}^{(i)} - \beta \mathbf{m}_{k+2}^{(i)}$, for some non-negative integers α, β such that not both are null. The other vertices are not modified so the remaining two rows of $\mathbf{M}^{(i+1)}$ are not modified. We get

$$\begin{aligned} \det(\mathbf{M}^{(i+1)}) &= \det(\mathbf{m}_k^{(i)} - \alpha \mathbf{m}_{k+1}^{(i)} - \beta \mathbf{m}_{k+2}^{(i)}, \mathbf{m}_{k+1}^{(i)}, \mathbf{m}_{k+2}^{(i)}) \\ &= \det(\mathbf{m}_k^{(i)}, \mathbf{m}_{k+1}^{(i)}, \mathbf{m}_{k+2}^{(i)}) && \text{(by linearity)} \\ &= \det(\mathbf{M}^{(i)}) = 1, && \text{(by induction hypothesis)} \end{aligned}$$

□

This shows that, for all steps $i \in \{0, \dots, n\}$, $\left\{ \mathbf{m}_0^{(i)}, \mathbf{m}_1^{(i)}, \mathbf{m}_2^{(i)} \right\}$ is a basis of $\mathbb{Z}^3$, which is especially useful in sec. 4.1.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

2.5 Termination and correctness of the algorithm

On a digital plane, we have an upper bound for the number of steps. The proof of the following theorem requires some invariant properties introduced in the previous section. For sake of clarity, we use the bar notation whenever a scalar product with $\mathbf{N}$ is required, i.e., $\overline{\mathbf{x}}$ instead of $\mathbf{x} \cdot \mathbf{N}$ for any vector $\mathbf{x} \in \mathbb{Z}^3$.

Theorem 2.3 ([LPR17],Theorem 1)**.** *The number of steps, n, is less than or equal to $\|\mathbf{N}\|_1 - 3$.*

Proof. The result comes from the fact that the sequence $(\sum_k \overline{\mathbf{m}}_k^{(i)})_{i=0,\dots,n}$ is a strictly decreasing sequence of integers between $\|\mathbf{N}\|_1$ and 3 because:

- $\forall k, \mathbf{m}_k^{(0)} = \mathbf{e}_{k+2}$ and $\sum_k \overline{\mathbf{m}}_k^{(0)} = \|\mathbf{N}\|_1$.
- by theorem 2.1,

$$\forall i \in \{0, \dots, n\},\ \forall k,\ \overline{\mathbf{m}}_k^{(i)} \geq 1 \text{ then } \sum_k \overline{\mathbf{m}}_k^{(i)} \geq 3.$$

- by lemma 2.1,

$$\forall i \in \{0, \dots, n-1\},\ \sum_k \overline{\mathbf{m}}_k^{(i)} > \sum_k \overline{\mathbf{m}}_k^{(i+1)}.$$

□

This bound is reached for any normal of components $(1, 1, r)$ with $r \in \mathbb{N} \setminus \{0\}$.

The correctness of the algorithm is studied in [LPR17] for the H-algorithm and the R-algorithm. The proofs are actually valid for any neighborhood S. Since we focus below on the last step n, we omit the exponent (n) in the proofs to improve their readability.

Theorem 2.4 ([LPR17],Theorem 2)**.** *If $\mathbf{p}$ is a lower leaning point (i.e. $\overline{\mathbf{p}} = 0$ and thus $\overline{\mathbf{q}} = \|\mathbf{N}\|_1$), the vertices of the last triangle are upper leaning points, i.e. $\forall k \in \{0, 1, 2\}, \overline{\mathbf{v}}_k^{(n)} = \|\mathbf{N}\|_1 - 1$.*

Proof. Let us introduce the following notation for the sides of the triangles: $\mathbf{d}_k := \mathbf{m}_{k+1} - \mathbf{m}_{k+2}$ for all $k \in \mathbb{Z}/3\mathbb{Z}$.

The first step of the proof is to show that the vertices of the last triangle are all at the same height, i.e., $\overline{\mathbf{m}}_0 = \overline{\mathbf{m}}_1 = \overline{\mathbf{m}}_2$. If not, then there exist $k \in \mathbb{Z}/3\mathbb{Z}$ such that $\overline{\mathbf{d}}_k := \overline{\mathbf{m}}_{k+1} - \overline{\mathbf{m}}_{k+2} \neq 0$. In this case, either (i) $\overline{\mathbf{d}}_k < 0$ or (ii) $\overline{\mathbf{d}}_k > 0$. Since $\overline{\mathbf{q}} = \|\mathbf{N}\|_1$ and $|\overline{\mathbf{d}}_k| < \|\mathbf{N}\|_1$, either (i) $\mathbf{q} + \mathbf{d}_k \in \mathbf{P}$ or (ii) $\mathbf{q} - \mathbf{d}_k \in \mathbf{P}$. This implies that $\mathcal{N}_S \cap \mathbf{P} \neq \emptyset$, which is a contradiction because $\mathcal{N}_S \cap \mathbf{P} = \emptyset$ at the last step. As a consequence, $\forall k, \overline{\mathbf{d}}_k = 0$ and $\forall k, \overline{\mathbf{m}}_k = \gamma$, a strictly positive integer.

The second step of the proof is to show that $\gamma = 1$. Let us denote by $\mathbf{1}$ the vector containing ones. We can write the last system as $\mathbf{MN} = \gamma\mathbf{1}$. Since $\mathbf{M}$ is invertible (because $\det(\mathbf{M}) = 1$ by theorem 2.2), $\mathbf{N} = \mathbf{M}^{-1}\gamma\mathbf{1}$ and as a consequence $\gamma = 1$ (because the components of $\mathbf{N}$ are relatively prime and $\mathbf{M}^{-1}$ is unimodular).

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

We conclude that $\forall k, \overline{\mathbf{m}}_k = 1$ and, straightforwardly, $\overline{\mathbf{v}}_k = \|\mathbf{N}\|_1 - 1$. □

The above theorem leads to two important results:

Corollary 2.1 ([LPR17],Corollary 4)**.** *If* $\mathbf{p}$ *is a lower leaning point, the normal of the last triangle is equal to* $\mathbf{N}$*, i.e.,* $\hat{\mathbf{N}}(\mathbf{T}^{(n)}) = \mathbf{N}$*.*

Proof. On one hand, $\mathbf{M}\hat{\mathbf{N}}(\mathbf{T}) = \mathbf{1}$ because $\forall k, ((\mathbf{m}_0 - \mathbf{m}_1) \times (\mathbf{m}_1 - \mathbf{m}_2)) \cdot \mathbf{m}_k = (\mathbf{m}_{k+1} \times \mathbf{m}_{k+2}) \cdot \mathbf{m}_k = \det(\mathbf{M})$, which is equal to 1 by theorem 2.2.

On the other hand, $\mathbf{MN} = \mathbf{1}$ by theorem 2.4. Since $\mathbf{M}$ is invertible, we have $\hat{\mathbf{N}}(\mathbf{T}) = \mathbf{N}$. □

Corollary 2.2 ([LPR17],Corollary 5)**.** *If* $\overline{\mathbf{p}} = 0$*, the normal of* $\mathbf{T}^{(n)}$ *is equal to* $\mathbf{N}$ *and any two edges form a basis of the 2D lattice of* upper leaning points*, i.e., the lattice* $\{\mathbf{x} \in \mathbb{Z}^3 \mid \mathbf{x} \cdot \mathbf{N} = \|\mathbf{N}\|_1 - 1\}$*.*

Proof. By theorem 2.2, the lattice

$$\left\{\mathbf{q} + a\mathbf{m}_0, \mathbf{q} + b\mathbf{m}_1, \mathbf{q} + c\mathbf{m}_2 \mid (a, b, c) \in \mathbb{Z}^3\right\}$$

is equivalent to $\mathbb{Z}^3$. Thus, the fundamental domain of this 3D lattice does not contain any integer point. This implies that the facet $conv(\mathbf{T})$, does not contain any integer point neither. We also know that the vertices of $\mathbf{T}$ are at the same height theorem 2.4 Therefore for any $k \in \mathbb{Z}/3\mathbb{Z}$, $(\mathbf{m}_{k+1} - \mathbf{m}_{k+2}, \mathbf{m}_{k+2} - \mathbf{m}_k)$ forms a basis of the 2D lattice of upper leaning points. □

2.6 Conclusion

In this chapter, we introduce the fundamental concept of digital surfaces. We explore various techniques for estimating normal vector and perform geometric analyses on discrete point sets. Within the more specific context of the *tetrahedron-based* plane-probing algorithms on a digital plane, we establish the notation that will be utilized in subsequent chapters and offer theoretical insights into the algorithm's validity. Notably, our upcoming focus in the next chapter is on a novel variant, the L-algorithm.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

CHAPTER 3

L-algorithm

Plane-probing algorithms are methods which adapt the neighborhood progressively. The first plane-probing algorithm was proposed in [LPR16a]. It probes some points in the digital surface and the output represents locally an approximation on the digital surface. Other *tetrahedron-based* plane-probing algorithms, such as the H-algorithm and the R-algorithm, were proposed later [LPR17]. The L-algorithm is one of the *tetrahedron-based* algorithms that considers more points than other variants. It is introduced in our paper that has been presented in the 2nd International Conference on Discrete Geometry and Mathematical Morphology [LRC22]. In this chapter, we discuss about the implementation of L-algorithm and its performance.

3.1 L-neighborhood

As defined in Sec. 2.3.2, the H, R and L-algorithm differ in terms of neighborhood. At every step i, from (2.6) and (2.7), the L-neighborhood can be rewritten as:

$$\mathcal{N}_{S_L}^{(i)} := \left\{ \mathbf{v}_k^{(i)} + \alpha \mathbf{m}_{k+1}^{(i)} + \beta \mathbf{m}_{k+2}^{(i)} \mid k \in \mathbb{Z}/3\mathbb{Z},\ (\alpha, \beta) \in S_L = \{(\alpha, \beta) \in \mathbb{N}^2 \setminus (0,0)\} \right\}. \tag{3.1}$$

Since the set $\mathcal{N}_{S_L}^{(i)} \cap \mathbf{P}$ is finite (by lemma 2.2), this ensures that the algorithm can find the *closest* point with regard to the relation $\leq_{\mathbf{T}}$. However, finding a closest point in $\mathcal{N}_{S_L}^{(i)} \cap \mathbf{P}$ with exhaustive search could require a lot of probes and becomes an expensive task for the L-algorithm (see lines 4 and 5 of Algorithm 1). In practice, one does not need to probe so much, because one can safely discard a large part of $\mathcal{N}_{S_L}^{(i)} \cap \mathbf{P}$. The aim of the next few sections is to show and prove how to reduce the actual number of probes.

In following sections, we focus on a step $i \in \{0, \ldots, n\}$ and for the sake of simplicity, we drop the exponent (i) in the notations. Furthermore, we focus on the 2D lattice

$$\forall k \in \mathbb{Z}/3\mathbb{Z},\ \mathcal{L}_k := \{ \mathbf{v}_k + \alpha \mathbf{m}_{k+1} + \beta \mathbf{m}_{k+2} \mid (\alpha, \beta) \in S_L \}.$$

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

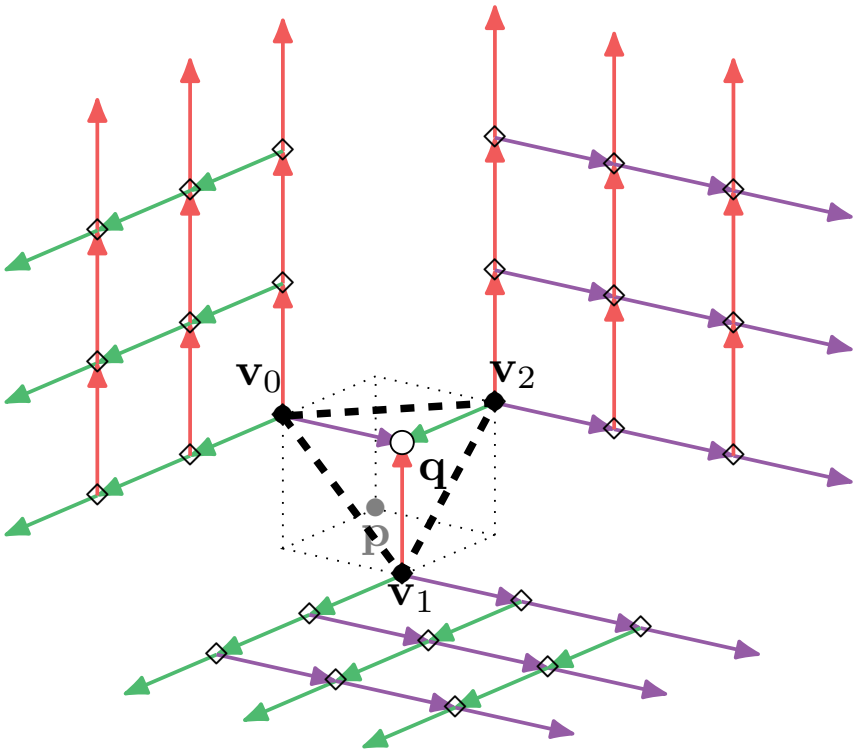
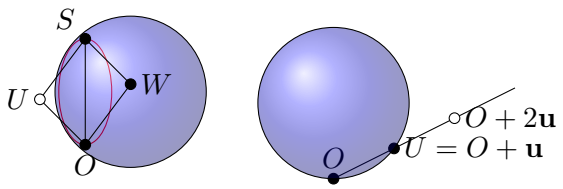


FIGURE 3.1: Illustration of the L-neighborhood $\mathcal{N}_{S_L}^{(i)}$ that includes every point on the lattices (marked as rhombus).

3.2 L-neighborhood navigation

The first idea is to discard the points "further" than others with respect to $\leq_{\mathbf{T}}$. To begin, we introduce two circumsphere related results:



(A) Lemma 3.1 (B) Lemma 3.2

FIGURE 3.2: Illustrations for circumsphere related lemmas. A point is depicted as a black disk if it is inside the ball of interest, and as a hollow disk if it is not in the closed ball.

Lemma 3.1. *Let* $\mathbf{u}, \mathbf{w}$ *be two non-zero vectors of* $\mathbb{R}^3$*. Let* B *be a closed ball whose border passes through the origin* O *and the point* $S := O + \mathbf{u} + \mathbf{w}$*. If* $\mathbf{u} \cdot \mathbf{w} \geq 0$*, at least one of the two points* $U := O + \mathbf{u}$ *and* $W := O + \mathbf{w}$ *lies in the ball.*

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Proof. Consider a ball B with centre c and radius r. Using the equivalence $O \in \partial B \Leftrightarrow r^2 = \|c\|^2$ it follows:

$$S \in \partial B \Leftrightarrow (O + \mathbf{u} + \mathbf{w} - c)^2 = \|c\|^2 \Leftrightarrow (\mathbf{u} + \mathbf{w})^2 = 2(\mathbf{u} + \mathbf{w}) \cdot c, \tag{3.2}$$

$$U \notin B \Leftrightarrow (O + \mathbf{u} - c)^2 > \|c\|^2 \Leftrightarrow 2c \cdot \mathbf{u} < \mathbf{u}^2, \tag{3.3}$$

$$W \in B \Leftrightarrow (O + \mathbf{w} - c)^2 > \|c\|^2 \Leftrightarrow 2c \cdot \mathbf{w} > \mathbf{w}^2. \tag{3.4}$$

If $\mathbf{u} \cdot \mathbf{w} \geq 0$ and suppose that $U \notin B$, then

$$\mathbf{u}^2 + \mathbf{w}^2 \underbrace{\leq}_{\mathbf{u}\cdot\mathbf{w}\geq 0} (\mathbf{u} + \mathbf{w})^2 \underbrace{<}_{(3.2) \text{ and } (3.3)} \mathbf{u}^2 + 2\mathbf{w} \cdot c. \tag{3.5}$$

Subtracting $\mathbf{u}^2$ from both sides gives (3.4), thus $W \in B$. With similar proof, if $\mathbf{u} \cdot \mathbf{w} \geq 0$ and $W \notin B$, we will have $U \in B$. □

Lemma 3.2. *Let* $\mathbf{u}$ *be a non-zero vector in* $\mathbb{R}^3$*. Let* B *be a closed ball whose border passes through the origin* O*. If* $U := O + \mathbf{u}$ *does not lie in the interior* $\mathring{B}$ *of the ball* B*, then no point* $O + \delta\mathbf{u}$ *such that* $\delta > 1$ *lies in the ball* B*.*

Proof. Consider a ball B with centre c and radius r, and recalling that $O \in \partial B$ is equivalent to $r^2 = \|c\|^2$, we get

$$O + \mathbf{u} \notin \mathring{B} \Leftrightarrow (\mathbf{u} - c)^2 \geq \|c\|^2 \Leftrightarrow 2c \cdot \mathbf{u} \leq \mathbf{u}^2. \tag{3.6}$$

We then compute:

$$\begin{aligned}
(O + \delta\mathbf{u} - c)^2 &= \delta^2\mathbf{u}^2 - 2\delta\mathbf{u} \cdot c + \|c\|^2 \\
&\geq \delta(\delta - 1)\mathbf{u}^2 + \|c\|^2 && \text{(by (3.6) and factoring } \mathbf{u}^2\text{)} \\
&> r^2. && \text{(since } \delta > 1, \mathbf{u}^2 > 0 \text{ and } \|c\|^2 = r^2\text{)}
\end{aligned}$$

We conclude since $(O + \delta\mathbf{u} - c)^2 > r^2$ is equivalent to $O + \delta\mathbf{u} \notin B$. □

An elementary application of the above two lemmas is the following result:

Lemma 3.3. *For all* $k \in \mathbb{Z}/3\mathbb{Z}$*, let* Λ_k *be the set* $\{\mathbf{v}_k + \alpha\mathbf{u} + \beta\mathbf{w} \mid (\alpha, \beta) \in S_L\}$*, where* $\mathbf{u}$*,* $\mathbf{w}$ *are any two non-zero vectors of* $\mathbb{Z}^3$ *such that* $\mathbf{v}_k + \mathbf{u}, \mathbf{v}_k + \mathbf{w} \in \mathcal{H}_+$*. If* $\mathbf{u} \cdot \mathbf{w} \geq 0$*, we have either* $\mathbf{v}_k + \mathbf{u} \leq_{\mathbf{T}} \mathbf{x}$ *for all* $\mathbf{x} \in \Lambda_k$ *or* $\mathbf{v}_k + \mathbf{w} \leq_{\mathbf{T}} \mathbf{x}$ *for all* $\mathbf{x} \in \Lambda_k$*.*

Proof. Let us consider the ball $\mathcal{B}(\mathbf{T}, \mathbf{x})$ for a point $\mathbf{x} := \mathbf{v}_k + \alpha\mathbf{u} + \beta\mathbf{w}$, with $\alpha, \beta \geq 1$. Since $\alpha\mathbf{u} \cdot \beta\mathbf{w} \geq 0$, by Lemma 3.1, we know that either $\mathbf{v}_k + \alpha\mathbf{u}$ or $\mathbf{v}_k + \beta\mathbf{w}$ lies in $\mathcal{B}(\mathbf{T}, \mathbf{x})$. Let us assume w.l.o.g. that $\mathbf{v}_k + \alpha\mathbf{u} \in \mathcal{B}(\mathbf{T}, \mathbf{x})$, which means that $\mathbf{v}_k + \alpha\mathbf{u} \leq_{\mathbf{T}} \mathbf{x}$. By Lemma 3.2, we then conclude that $\mathbf{v}_k + \mathbf{u} \leq_{\mathbf{T}} \mathbf{v}_k + \alpha\mathbf{u} \leq_{\mathbf{T}} \mathbf{x}$. □

From now on, let us denote the angle between two vectors $\mathbf{u}$ and $\mathbf{w}$ as $\angle\mathbf{uw}$. By lemma 3.3, if the angle between $\mathbf{m}_{k+1}$ and $\mathbf{m}_{k+2}$, denoted as $\angle\mathbf{m}_{k+1}\mathbf{m}_{k+2}$ is acute or right, the number of points that one needs to consider in $\mathcal{L}_k$ is only two: $\mathbf{v}_k + \mathbf{m}_{k+1}$ and $\mathbf{v}_k + \mathbf{m}_{k+2}$.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

When the angle is obtuse we divide the angle into two with the vector $\mathbf{m}_{k+1}+\mathbf{m}_{k+2}$. Then we know that at least one of the following angle is acute: $\angle\mathbf{m}_{k+1}(\mathbf{m}_{k+1}+\mathbf{m}_{k+2})$ or $\angle(\mathbf{m}_{k+1}+\mathbf{m}_{k+2})\mathbf{m}_{k+2}$. If one of the two angles is obtuse, we continue the divide operation with this angle until we do not spot any obtuse angle (see Algorithm 2).

Definition 3.1. *For any pair of linearly independent non-zero vectors* $(\mathbf{u},\mathbf{w})\in\mathbb{Z}^3\times\mathbb{Z}^3$*, we define a sequence of vector pairs* $\Omega^0_{\mathbf{u},\mathbf{w}}=\{(\mathbf{u}_j,\mathbf{w}_j)\}$ *as follow:*

1. $\mathbf{u}_0=\mathbf{u}$ *and* $\mathbf{w}_0=\mathbf{w}$.
2. *For any* $j\geq 0$*, the pair* $(\mathbf{u}_{j+1},\mathbf{w}_{j+1})$ *is defined in the following cases:*

$$\begin{cases}(\mathbf{u}_{j+1},\mathbf{w}_{j+1}):=(\mathbf{u}_j+\mathbf{w}_j,\mathbf{w}_j) & \text{if } (\mathbf{u}_j+\mathbf{w}_j)\cdot\mathbf{w}_j\leq 0,\\ (\mathbf{u}_{j+1},\mathbf{w}_{j+1}):=(\mathbf{u}_j,\mathbf{u}_j+\mathbf{w}_j) & \text{if } \mathbf{u}_j\cdot(\mathbf{u}_j+\mathbf{w}_j)\leq 0.\end{cases} \tag{3.7}$$

We remark that the above cases in (3.7) only exist when $\mathbf{u}_j\cdot\mathbf{w}_j\leq 0$.

For $k\in\mathbb{Z}/3\mathbb{Z}$, we define the (naive) candidate set as

$$Cand_0^{(k)}=\bigcup_{(\mathbf{u},\mathbf{w})\in\Omega_{\mathbf{m}_{k+1},\mathbf{m}_{k+2}}}\{\mathbf{u},\mathbf{w}\}. \tag{3.8}$$

Algorithm 2: CreateCandidateListNaive(InPlane, $\mathbf{T}$, $\mathbf{q}$, k)

Input: The predicate InPlane, the triangle $\mathbf{T}$, the point $\mathbf{q}$ and an index $k\in\{0,1,2\}$
Output: A list $Cand_0^{(k)}$ of candidate points around vertex $\mathbf{v}_k$

```
Initialize Cand_0^(k); (u, w) ← (q − v_{k+1}, q − v_{k+2});
Add v_k + u to Cand_0^(k);
Add v_k + w to Cand_0^(k);
while u · w < 0 do
    Add v_k + u + w to Cand_0^(k);
    if u · (u + w) ≤ 0 then
        (u, w) ← (u + w, w);
    else
        if w · (u + w) ≤ 0 then
            (u, w) ← (u, u + w);
return Cand_0^(k);
```

This process will end because:

Lemma 3.4. *For any* $k\in\mathbb{Z}/3\mathbb{Z}$*,* $Cand_0^{(k)}$ *is finite.*

Proof. In fact, for any $j\geq 0$, if the pair $(\mathbf{u}_{j+1},\mathbf{w}_{j+1})$ exists, then we have either $\mathbf{u}_{j+1}\cdot\mathbf{w}_{j+1}=\mathbf{u}_j\cdot\mathbf{w}_j+|\mathbf{u}_j|^2$ or $\mathbf{u}_{j+1}\cdot\mathbf{w}_{j+1}=\mathbf{u}_j\cdot\mathbf{w}_j+|\mathbf{w}_j|^2$. The negative integer sequence $(\mathbf{u}_j\cdot\mathbf{w}_j)_j$ is thus strictly increasing and bounded by zero, thus finite. □

We have just shown how to navigate through the L-neighborhood naively and select a finite set of candidate points $Cand_0^{(k)}$. The algorithm will compare and find the *closest* point in $Cand_0^{(k)}\cap\mathbf{P}$ and update one vertex of the triangle. This method works, but

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

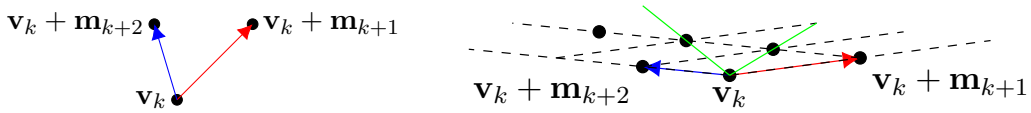
there exist a faster way to explore the neighborhood, which we will introduce in the following section.

3.3 Reduction of the L-neighborhood

We propose a more efficient algorithm (Algorithm 3) that selects a small and sufficient set of candidate points included in $\mathcal{L}_k$.

3.3.1 A Smaller Candidate Set

In the preceding section, we divided the target angle into two parts, and then proceeded only if there was an obtuse angle remaining. This process can be optimized because, in any angle partition containing an obtuse angle, there is at most one obtuse angle. In the following, we will explain how to identify and locate this obtuse angle, if it exists. Then, we will show how to recursively divide the angle between the two vectors $\mathbf{m}_{k+1}$ and $\mathbf{m}_{k+2}$ issued from $\mathbf{v}_k$.



(A) (B)

FIGURE 3.3: Angle between $\mathbf{m}_{k+1}$ and $\mathbf{m}_{k+2}$: (a) when $\mathbf{m}_{k+1} \cdot \mathbf{m}_{k+2} \geq 0$, (b) when $\mathbf{m}_{k+1} \cdot \mathbf{m}_{k+2} < 0$ and $\mathbf{m}_{k+2} \cdot (\mathbf{m}_{k+1} + \mathbf{m}_{k+2}) < 0$. Here, we also have $(\mathbf{m}_{k+1} + \gamma \mathbf{m}_{k+2}) \cdot (\mathbf{m}_{k+1} + (\gamma + 1)\mathbf{m}_{k+2}) < 0$ with $\gamma = 1$ (see Lemma 3.5 and the green angle).

When tracing along the points on a ray of the 2D lattice $\mathcal{L}_k$ and considering the angles formed with the point $\mathbf{v}_k$, we observe that there can be at most one obtuse angle (see for instance Fig. 3.3 (b)). Furthermore, if the obtuse angle exists, we would know its location (lemma 3.5) and we can prove that other angles are all acute (lemma 3.6).

Lemma 3.5. *Let* $\mathbf{u}$, $\mathbf{w}$ *be two non-zero and linearly independent vectors in* $\mathbb{Z}^3$. *If there exists* $\gamma \geq 1$ *such that*

$$(\mathbf{u} + \gamma \mathbf{w}) \cdot (\mathbf{u} + (\gamma + 1)\mathbf{w}) < 0, \tag{3.9}$$

then γ *is the unique integer greater than or equal to* 1 *that verifies*

$$(\mathbf{u} + (\gamma + 1)\mathbf{w}) \cdot \mathbf{w} > 0 > (\mathbf{u} + \gamma \mathbf{w}) \cdot \mathbf{w}. \tag{3.10}$$

In this case, $\gamma = \left\lfloor \frac{-\mathbf{u} \cdot \mathbf{w}}{\|\mathbf{w}\|^2} \right\rfloor$.

Proof. We refer to Fig. 3.3(b) for an example where $\mathbf{u} = \mathbf{m}_{k+1}$ and $\mathbf{w} = \mathbf{m}_{k+2}$. By rewritting the left-hand side of (3.9) as $(\mathbf{u} + \gamma \mathbf{w}) \cdot ((\mathbf{u} + \gamma \mathbf{w}) + \mathbf{w})$ and developing, we get

$$\|\mathbf{u} + \gamma \mathbf{w}\|^2 + (\mathbf{u} + \gamma \mathbf{w}) \cdot \mathbf{w} < 0 \Rightarrow (\mathbf{u} + \gamma \mathbf{w}) \cdot \mathbf{w} < 0\,,$$

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

which is the right-hand side of (3.10). Similarly, by rewritting the left-hand side of (3.9) as $((\mathbf{u}+(\gamma+1)\mathbf{w})-\mathbf{w})\cdot(\mathbf{u}+(\gamma+1)\mathbf{w})<0$ and developing, we have

$$\|(\mathbf{u}+(\gamma+1)\mathbf{w})\|^2-\mathbf{w}\cdot(\mathbf{u}+(\gamma+1)\mathbf{w})<0.$$

As this expression is strictly negative by (3.9), we obtain $(\mathbf{u}+(\gamma+1)\mathbf{w})\cdot\mathbf{w}>0$, which is the left-hand side of (3.10). To end, by developing (3.10) and isolating the γ, we obtain $\gamma+1>\frac{-\mathbf{u}\cdot\mathbf{w}}{\|\mathbf{w}\|^2}>\gamma$, thus unicity. □

Lemma 3.6. *Let* $\mathbf{u}$, $\mathbf{w}$ *be two non-zero and linearly independent vectors in* $\mathbb{Z}^3$. *If there exists* $\gamma\geq 1$ *verifying (3.9), then for all* $c\in\{0,1,\ldots,\gamma-1\}$, $(\mathbf{u}+c\mathbf{w})\cdot(\mathbf{u}+(c+1)\mathbf{w})>0$.

Proof. First, observe that for all $c\in\mathbb{N}\setminus\{0\}$,

$$(\mathbf{u}+(c-1)\mathbf{w})\cdot(\mathbf{u}+c\mathbf{w})=(\mathbf{u}+c\mathbf{w})\cdot(\mathbf{u}+(c+1)\mathbf{w})-2\mathbf{w}\cdot(\mathbf{u}+c\mathbf{w}). \tag{3.11}$$

To determine the sign of $-2\mathbf{w}\cdot(\mathbf{u}+c\mathbf{w})$, note that we obviously have $c\mathbf{w}^2<\gamma\mathbf{w}^2$ and, from the right-hand side of (3.10), $\gamma\mathbf{w}^2<-\mathbf{u}\cdot\mathbf{w}$. As a result,

$$c\mathbf{w}^2<-\mathbf{u}\cdot\mathbf{w}\Leftrightarrow\mathbf{w}\cdot(\mathbf{u}+c\mathbf{w})<0.$$

Since $-2\mathbf{w}\cdot(\mathbf{u}+c\mathbf{w})>0$, it is enough to show that the statement is true for $c=\gamma-1$ because the result for the smaller values of c then follows by induction.

By (3.10), we also have $\mathbf{w}\cdot(\mathbf{u}+\gamma\mathbf{w})<0<(\mathbf{u}+\gamma\mathbf{w})^2$. Therefore,

$$2\mathbf{w}\cdot(\mathbf{u}+\gamma\mathbf{w})<(\mathbf{u}+\gamma\mathbf{w})^2+\mathbf{w}\cdot(\mathbf{u}+\gamma\mathbf{w})=(\mathbf{u}+\gamma\mathbf{w})\cdot(\mathbf{u}+(\gamma+1)\mathbf{w}).$$

From this lower bound and replacing c by γ in (3.11), we finally obtain $(\mathbf{u}+(\gamma-1)\mathbf{w})\cdot(\mathbf{u}+\gamma\mathbf{w})>0$, which concludes the proof. □

The two previous lemmas provide a set of lattice bases whose vectors form an acute angle. Indeed, with $\mathbf{u}=\mathbf{m}_{k+1}$ and $\mathbf{w}=\mathbf{m}_{k+2}$ and assuming that γ exists, we have $(\mathbf{m}_{k+1}+(\gamma+1)\mathbf{m}_{k+2})\cdot\mathbf{m}_{k+2}>0$ (lemma 3.5) and for all $c\in\{0,1,\ldots,\gamma-1\}$, $(\mathbf{m}_{k+1}+c\mathbf{m}_{k+2})\cdot(\mathbf{m}_{k+1}+(c+1)\mathbf{m}_{k+2})>0$ (lemma 3.6). Then, it straightforwardly follows from lemma 3.3 that the closest points in the set

$$\{\mathbf{v}_k+\mathbf{m}_{k+2}\}\cup\{\mathbf{v}_k+\mathbf{m}_{k+1}+c\mathbf{m}_{k+2}\mid c\in\{0,\ldots,\gamma+1\}\}$$

are closer than any other points in the set

$$\mathcal{L}_k\setminus\{\mathbf{v}_k+\alpha(\mathbf{m}_{k+1}+\gamma\mathbf{m}_{k+2})+\beta(\mathbf{m}_{k+1}+(\gamma+1)\mathbf{m}_{k+2})\mid(\alpha,\beta)\in S_L\}$$

(see fig. 3.4).

One part of $\mathcal{L}_k$ cannot be covered because $(\mathbf{m}_{k+1}+\gamma\mathbf{m}_{k+2})\cdot(\mathbf{m}_{k+1}+(\gamma+1)\mathbf{m}_{k+2})<0$. In this case, we can again apply previous lemmas to the pair of vectors $(\mathbf{u},\mathbf{w})=(\mathbf{m}_{k+2},\mathbf{m}_{k+1}+\gamma+\mathbf{m}_{k+2})$. We recursively apply the lemmas until we can no longer

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

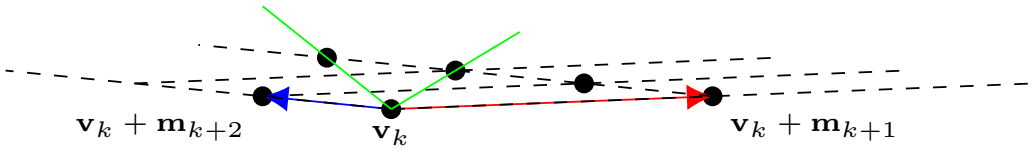


FIGURE 3.4: When $\gamma = 2$, i.e. $(\mathbf{m}_{k+1} + 2\mathbf{m}_{k+2}) \cdot (\mathbf{m}_{k+1} + 3\mathbf{m}_{k+2}) < 0$ (green angle). For any point in $\mathcal{L}_k$ that is not within the green angle, it is "further" than one of the points of the set $\{\mathbf{m}_{k+1} + c\mathbf{m}_{k+2} \mid 0 \leq c \leq \gamma + 1\}$ or the point $\mathbf{m}_{k+2}$.

find an obtuse angle. Here, we define formally a sequence of points in order to describe the candidate set of points deduced from previous results.

Definition 3.2. *For any pair of linearly independent non-zero vectors* $(\mathbf{u}, \mathbf{w}) \in \mathbb{Z}^3 \times \mathbb{Z}^3$*, we define a sequence of vector pairs* $\Omega_{\mathbf{u},\mathbf{w}} = \{(\mathbf{u}_j, \mathbf{w}_j)\}_{j\geq 0}$ *as follows:*

1. $\mathbf{u}_0 = \mathbf{u}$ *and* $\mathbf{w}_0 = \mathbf{w}$.
2. *For any* $j \geq 0$*, the pair* $(\mathbf{u}_{j+1}, \mathbf{w}_{j+1})$ *exists if and only if there exists* $\gamma_j \geq 1$ *such that*
$$(\mathbf{u}_j + \gamma_j \mathbf{w}_j) \cdot (\mathbf{u}_j + (\gamma_j + 1)\mathbf{w}_j) < 0, \tag{3.12}$$
then
$$\mathbf{u}_{j+1} := \mathbf{w}_j\,, \quad \mathbf{w}_{j+1} := \mathbf{u}_j + \gamma_j \mathbf{w}_j. \tag{3.13}$$

Definition 3.3 (Candidate set)**.** *For* $k \in \mathbb{Z}/3\mathbb{Z}$ *and for any pair of vectors* $(\mathbf{u}, \mathbf{w})$ *in the set* $\{(\mathbf{m}_{k+1}, \mathbf{m}_{k+2}), (\mathbf{m}_{k+2}, \mathbf{m}_{k+1})\}$*, we define*

$$C_k := \bigcup_{(\mathbf{u}_j, \mathbf{w}_j) \in \Omega(\mathbf{u},\mathbf{w})} \{\mathbf{v}_k + \mathbf{w}_j\} \cup \{\mathbf{v}_k + \mathbf{u}_j + c\mathbf{w}_j \mid c \in \{0, \ldots, \gamma_j + 1\}\}.$$

The finiteness of C_k stems from the finiteness of $\Omega_{\mathbf{u},\mathbf{w}}$:

Lemma 3.7. *The sequence* $\Omega_{\mathbf{u},\mathbf{w}} = \{(\mathbf{u}_j, \mathbf{w}_j)\}_{j\geq 0}$ *is finite.*

Proof. From (3.13), we have for any $j \geq 0$, $-\mathbf{u}_{j+1} \cdot \mathbf{w}_{j+1} = -\mathbf{w}_j \cdot (\mathbf{u}_j + \gamma_j \mathbf{w}_j)$. Developing the last expression, we obtain $-\mathbf{u}_j \cdot \mathbf{w}_j - \gamma_j \|\mathbf{w}_j\|^2$, which is strictly less than $-\mathbf{u}_j \cdot \mathbf{w}_j$. Therefore, the sequence of natural numbers $\{-\mathbf{u}_j \cdot \mathbf{w}_j\}_{j\geq 0}$ is strictly decreasing. Since, in addition, $-\mathbf{u}_j \cdot \mathbf{w}_j \geq \|\mathbf{w}_j\|^2$ while there exists $\gamma_j \geq 1$, there is a lower bound and it follows that the sequence $\Omega_{\mathbf{u},\mathbf{w}}$ is finite. □

We have just proven that only a small part of $\mathcal{L}_k$ has to be taken into account. In the subsequent section, we demonstrate a further reduction of the candidate set.

3.3.2 Technical details related to the preorder

We could further decrease the number of points, which will be detailed in sec. 3.3.3. To achieve this, we need to understand more about the preorder $\leq_{\mathbf{T}}$ and some technical details. The following result will serve further in the proof of lemma 3.9 and Theorem 4.1.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Another preoreder First, we observe that there is another preorder in the half-space $\mathcal{H}_-$, the open half-space lying below the plane incident to $\mathbf{T}$.

Lemma 3.8. *For all* $\mathbf{x}, \mathbf{y} \in \mathcal{H}_+$*, if* $\mathbf{y} \leq_{\mathbf{T}} \mathbf{x}$ *(i.e.* $(\mathcal{B}(\mathbf{T}, \mathbf{y}) \cap \mathcal{H}_+) \subseteq (\mathcal{B}(\mathbf{T}, \mathbf{x}) \cap \mathcal{H}_+)$*), then* $(\mathcal{B}(\mathbf{T}, \mathbf{x}) \cap \mathcal{H}_-) \subseteq (\mathcal{B}(\mathbf{T}, \mathbf{y}) \cap \mathcal{H}_-)$.

Proof. Again, we have two cases according to the pair $\mathbf{x}, \mathbf{y}$:

- if $\mathcal{B}(\mathbf{T}, \mathbf{x}) = \mathcal{B}(\mathbf{T}, \mathbf{y})$, the statement is obviously true.
- if $\mathcal{B}(\mathbf{T}, \mathbf{x}) \neq \mathcal{B}(\mathbf{T}, \mathbf{y})$, we first note that we can symmetrically define a total preorder with $\mathcal{H}_-$ instead of $\mathcal{H}_+$, which means that there are only two possible cases: either $(\mathcal{B}(\mathbf{T}, \mathbf{x}) \cap \mathcal{H}_-) \subseteq (\mathcal{B}(\mathbf{T}, \mathbf{y}) \cap \mathcal{H}_-)$ or $(\mathcal{B}(\mathbf{T}, \mathbf{y}) \cap \mathcal{H}_-) \subseteq (\mathcal{B}(\mathbf{T}, \mathbf{x}) \cap \mathcal{H}_-)$. We now show by contradiction that the second one is impossible. Indeed, if we have the two following inclusions

$$(\mathcal{B}(\mathbf{T}, \mathbf{y}) \cap \mathcal{H}_+) \subseteq (\mathcal{B}(\mathbf{T}, \mathbf{x}) \cap \mathcal{H}_+),$$
$$(\mathcal{B}(\mathbf{T}, \mathbf{y}) \cap \mathcal{H}_-) \subseteq (\mathcal{B}(\mathbf{T}, \mathbf{x}) \cap \mathcal{H}_-),$$

by taking the union of both sides of the inclusions, we obtain $\mathcal{B}(\mathbf{T}, \mathbf{y}) \subseteq \mathcal{B}(\mathbf{T}, \mathbf{x})$, which raises a contradiction as the two balls are assumed to be disinct and, by definition, intersect.

□

Understanding the Preorder with Determinants The implicit equation of the sphere can be written as a determinant. Thus, we can describe the algebraic distance of a point $\mathbf{x}'$ to the circumsphere of $\mathbf{T} \cup \{\mathbf{x}\}$ as a 5×5 determinant:

$$\delta_{\mathbf{T}}(\mathbf{x}, \mathbf{x}') := \begin{vmatrix} \mathbf{v}_0 & \mathbf{v}_1 & \mathbf{v}_2 & \mathbf{x} & \mathbf{x}' \\ {\mathbf{v}_0}^2 & {\mathbf{v}_1}^2 & {\mathbf{v}_2}^2 & \mathbf{x}^2 & {\mathbf{x}'}^2 \\ 1 & 1 & 1 & 1 & 1 \end{vmatrix}.$$

This notation $\delta_{\mathbf{T}}(\cdot, \cdot)$ is first introduced in [RL19]. We remark that $\delta_{\mathbf{T}}(\mathbf{x}, \mathbf{x}') \leq 0 \Leftrightarrow \mathbf{x}' \leq_{\mathbf{T}} \mathbf{x}$ means that $\mathbf{x}'$ is inside or on the circumball of $\mathbf{T} \cup \{\mathbf{x}\}$. By the update rule of the algorithm, if $\mathbf{x}$ is the chosen point, then $\delta_{\mathbf{T}}(\mathbf{y}, \mathbf{x}) \leq 0$ for all points in the neighborhood $\mathbf{y} \in \mathcal{N}_S$ (see algorithm 1 - line 4).

Let us consider the point $\mathbf{v}_0$ as the origin. In order to simplify notations, we set $\mathbf{y} := \mathbf{x} - \mathbf{v}_0$, $\mathbf{y}' := \mathbf{x}' - \mathbf{v}_0$ and using $\mathbf{d}_k = \mathbf{v}_{k+1} - \mathbf{v}_k = \mathbf{m}_k - \mathbf{m}_{k+1}$ for all k, we have:

$$\delta^0_{\mathbf{T}}(\mathbf{y}, \mathbf{y}') := \delta_{\mathbf{T}}(\mathbf{v}_0 + \mathbf{y}, \mathbf{v}_0 + \mathbf{y}') = \begin{vmatrix} \mathbf{d}_0 & -\mathbf{d}_2 & \mathbf{y} & \mathbf{y}' \\ {\mathbf{d}_0}^2 & {\mathbf{d}_2}^2 & \mathbf{y}^2 & {\mathbf{y}'}^2 \end{vmatrix}. \tag{3.14}$$

The relation between the algebraic distance of a point $\mathbf{y}$ to the circumball of $\mathbf{T} \cup \{\mathbf{x}\}$ converts to the following:

$$\forall \mathbf{y}, \mathbf{y}' \in \mathcal{H}_+, \quad \delta^0_{\mathbf{T}}(\mathbf{y}, \mathbf{y}') \geq 0 \Leftrightarrow \mathbf{v}_0 + \mathbf{y} \leq_{\mathbf{T}} \mathbf{v}_0 + \mathbf{y}' \tag{3.15}$$

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Let us denote by $[\mathbf{z}, \mathbf{z}', \mathbf{z}'']$ the 3×3 matrix composed of columns $\mathbf{z}, \mathbf{z}', \mathbf{z}''$. We give below a formula for $\delta^0_{\mathbf{T}}(\mathbf{z}, \mathbf{z}' + \alpha\mathbf{z}'')$ for any $\mathbf{z}, \mathbf{z}', \mathbf{z}'' \in \mathbb{R}^3$ using the cofactor expansion of the determinant (3.14):

$$\begin{aligned}\delta^0_{\mathbf{T}}(\mathbf{z}, \mathbf{z}' + \alpha\mathbf{z}'') = -{\mathbf{d}_0}^2 \det[-\mathbf{d}_2, \mathbf{z}, \mathbf{z}' + \alpha\mathbf{z}''] + {\mathbf{d}_2}^2 \det[\mathbf{d}_0, \mathbf{z}, \mathbf{z}' + \alpha\mathbf{z}''] \\ - \mathbf{z}^2 \det[\mathbf{d}_0, -\mathbf{d}_2, \mathbf{z}' + \alpha\mathbf{z}''] + (\mathbf{z}' + \alpha\mathbf{z}'')^2 \det[\mathbf{d}_0, -\mathbf{d}_2, \mathbf{z}].\end{aligned}$$

Due to the multilinearity inherent in the determinant's nature, we can derive the following identity [RL19, equation (6)]:

$$\begin{aligned}\delta^0_{\mathbf{T}}(\mathbf{z}, \mathbf{z}' + \alpha\mathbf{z}'') = \delta^0_{\mathbf{T}}(\mathbf{z}, \mathbf{z}') + \alpha\delta^0_{\mathbf{T}}(\mathbf{z}, \mathbf{z}'') \\ + \Big(\alpha^2({\mathbf{z}''}^2) + \alpha(-{\mathbf{z}''}^2 + 2\mathbf{z}' \cdot \mathbf{z}'')\Big) \det[\mathbf{d}_0, -\mathbf{d}_2, \mathbf{z}].\end{aligned} \tag{3.16}$$

In particular, when $\alpha = 1$, we have:

$$\delta^0_{\mathbf{T}}(\mathbf{z}, \mathbf{z}' + \mathbf{z}'') = \delta^0_{\mathbf{T}}(\mathbf{z}, \mathbf{z}') + \delta^0_{\mathbf{T}}(\mathbf{z}, \mathbf{z}'') + (2\mathbf{z}' \cdot \mathbf{z}'') \det[\mathbf{d}_2, -\mathbf{d}_1, \mathbf{z}]. \tag{3.17}$$

The above equation is used below as well as in sec. 4.2.3.

3.3.3 Even smaller candidate set

The set C_k described in sec. 3.3.1 is a union of subsets of aligned points. We show below that, for each subset, the last point is always closer than the other ones:

Lemma 3.9. *For any vectors $\mathbf{u}, \mathbf{w} \in \mathcal{L}_k$ such that there exists $\gamma \geq 1$ such that $(\mathbf{u}+\gamma\mathbf{w}) \cdot (\mathbf{u}+(\gamma+1)\mathbf{w}) < 0$, then for any $0 \leq c \leq \gamma - 1$, we have $\mathbf{u}+\gamma\mathbf{w} \leq_{\mathbf{T}} \mathbf{u}+c\mathbf{w}$.*

Proof. We assume w.l.o.g. that $k = 0$ and we use the notation $\delta^0_{\mathbf{T}}(\mathbf{x}, \mathbf{y})$ introduced in [RL19], where $\mathbf{x}$ and $\mathbf{y}$ are relative points of $\mathbb{Z}^3$ when considering $\mathbf{v}_0$ as origin. We recall that if $\mathbf{v}_0 + \mathbf{x} \in \mathcal{H}_+$, then $\mathbf{v}_0 + \mathbf{x} \leq_{\mathbf{T}} \mathbf{v}_0 + \mathbf{y} \Leftrightarrow \delta^0_{\mathbf{T}}(\mathbf{x}, \mathbf{y}) \geq 0$.

In order to show that for all $0 \leq c \leq \gamma - 1$, $\delta^0_{\mathbf{T}}(\mathbf{u} + \gamma\mathbf{w}, \mathbf{u} + c\mathbf{w}) \geq 0$, we use the identity described in (3.17).

Indeed, as $c = \gamma - (\gamma - c)$, we obtain (with $\mathbf{z} = \mathbf{z}' = \mathbf{u} + \gamma\mathbf{w}$ and $\mathbf{z}'' = -(\gamma - c)\mathbf{w}$):

$$\begin{aligned}\delta^0_{\mathbf{T}}(\mathbf{u} + \gamma\mathbf{w}, \mathbf{u} + c\mathbf{w}) = \underbrace{\delta^0_{\mathbf{T}}(\mathbf{u} + \gamma\mathbf{w}, \mathbf{u} + \gamma\mathbf{w})}_{=0} + \underbrace{\delta^0_{\mathbf{T}}(\mathbf{u} + \gamma\mathbf{w}, -(\gamma - c)\mathbf{w})}_{\geq 0, \text{ see item 1}} \\ - 2(\gamma - c) \quad \underbrace{(\mathbf{u} + \gamma\mathbf{w}) \cdot \mathbf{w}}_{\leq 0 \text{ by Lemma 3.5 (3.10)}} \quad \underbrace{\det[\mathbf{m}_0 - \mathbf{m}_1, \mathbf{m}_0 - \mathbf{m}_2, \mathbf{u} + \gamma\mathbf{w}]}_{>0, \text{ see item 2}}.\end{aligned}$$

1. Let $\mathcal{H}_-$ be the open half-space lying below the plane incident to $\mathbf{T}$. Let us set $\mathbf{x} := \mathbf{u} + \gamma\mathbf{w}$ and $\mathbf{y} := -(\gamma - c)\mathbf{w}$. By definition, $\mathbf{v}_0 + \mathbf{x} \in \mathcal{H}_+$ and $\mathbf{v}_0 + \mathbf{y} \in \mathcal{H}_-$. We have to prove that $\mathbf{v}_0 + \mathbf{x} \leq_{\mathbf{T}} \mathbf{v}_0 + \mathbf{y}$. Let $\mathbf{x}^\star$ be the closest point chosen for update. By definition, $\mathbf{x}^\star \leq_{\mathbf{T}} \mathbf{v}_0 + \mathbf{x}$, which implies that $(\mathcal{H}_- \cap \mathcal{B}(\mathbf{T}, \mathbf{v}_0 + \mathbf{x})) \subseteq (\mathcal{H}_- \cap \mathcal{B}(\mathbf{T}, \mathbf{x}^\star))$ (see Lemma 3.8). Due to the above inclusion relation, since $\mathbf{v}_0 + \mathbf{y}$ is not in the interior of $\mathcal{B}(\mathbf{T}, \mathbf{x}^\star)$ (by Delaunay property, see chapter 4), $\mathbf{v}_0 + \mathbf{y}$ is not in the interior of $\mathcal{B}(\mathbf{T}, \mathbf{v}_0 + \mathbf{x})$ either, i.e., $\mathbf{v}_0 + \mathbf{x} \leq_{\mathbf{T}} \mathbf{v}_0 + \mathbf{y}$.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

2. For any $(\alpha, \beta) \in S_L$, $\det[\mathbf{m}_0 - \mathbf{m}_1, \mathbf{m}_0 - \mathbf{m}_2, \alpha\mathbf{m}_1 + \beta\mathbf{m}_2] = \alpha + \beta > 0$, because $\det[\mathbf{m}_0, \mathbf{m}_1, \mathbf{m}_2] = 1$ (see Theorem 2.2). Notably,

$$\det[\mathbf{m}_0 - \mathbf{m}_1, \mathbf{m}_0 - \mathbf{m}_2, \mathbf{u} + \gamma\mathbf{w}] > 0.$$

□

Lemma 3.9 shows that the last point should be the closest. However, in the case where this last point is not in $\mathbf{P}$, we can resort to a binary search as in [LPR17, Algorithm 4].

3.4 Algorithm

Fig. 3.5 sums up the process of filtering the set $\mathcal{L}_k$. However, we have to discard the points that are not in $\mathbf{P}$. For this purpose, we use the predicate InPlane in the whole procedure detailed in Algorithm 3.

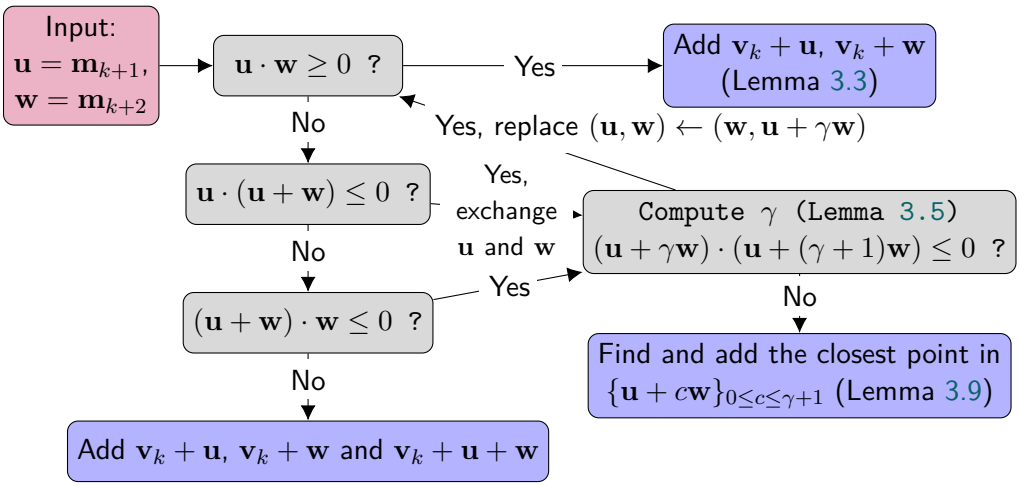


FIGURE 3.5: Roadmap

Theorem 3.1. *Algorithm 3 requires $O(\log \|\mathbf{N}\|_1)$ calls to the predicate* InPlane.

Proof. We consider the sequence of vectors $(\mathbf{u}_j, \mathbf{w}_j)_{0 \leq j \leq j_{max}}$. For any $j \geq 2$, if we rewrite the equation (3.13) with only $\mathbf{u}_{j-2}$, $\mathbf{u}_{j-1}$ and $\mathbf{u}_j$, we obtain the relation $\mathbf{u}_j = \mathbf{u}_{j-2} + \gamma_j \mathbf{u}_{j-1}$. We use the bar notation $\bar{\cdot}$ above any vector $\mathbf{x}$ to denote its height relative to $\mathbf{N}$. Otherwise said, $\overline{\mathbf{x}} := \mathbf{x} \cdot \mathbf{N}$. Then, we have $\overline{\mathbf{u}}_j = \overline{\mathbf{u}}_{j-2} + \gamma_j \overline{\mathbf{u}}_{j-1} \geq \overline{\mathbf{u}}_{j-2} + \overline{\mathbf{u}}_{j-1}$ (because $\gamma_j \geq 1$ and $\overline{\mathbf{u}}_{j-1} \geq 0$ by recurrence). By induction, we have for all $2 \leq j \leq j_{max}$, $\overline{\mathbf{u}}_j \geq 2^{\lfloor \frac{j}{2} \rfloor}(\overline{\mathbf{u}}_0 + \overline{\mathbf{u}}_1)$, which leads to $j_{max} \in O(\log \|\mathbf{N}\|_1)$, because the last point must be in $\mathbf{P}$, i.e., $\overline{\mathbf{u}}_{j_{max}} \leq \|\mathbf{N}\|_1$. Note that there is only one call to the predicate at each rank $2 \leq j \leq j_{max}$ (and at most four calls before), hence a total of $O(\log \|\mathbf{N}\|_1)$ calls at the last rank. It remains to notice that the final search also requires at most $O(\log \|\mathbf{N}\|_1)$ calls with an appropriate procedure such as [LPR17, Algorithm 4]. □

The overall complexity of the L-algorithm is discussed in the next section.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Algorithm 3: CreateCandidateList(InPlane, $\mathbf{T}$, $\mathbf{q}$, k)

Input: The predicate InPlane, the triangle $\mathbf{T}$, the point $\mathbf{q}$ and an index $k \in \{0, 1, 2\}$
Output: A list $Cand_k$ of candidate points around vertex $\mathbf{v}_k$

Initialize $Cand_k$; $(\mathbf{m}_1, \mathbf{m}_2) \leftarrow (\mathbf{q} - \mathbf{v}_{k+1}, \mathbf{q} - \mathbf{v}_{k+2})$;
Add $\mathbf{v}_k + \mathbf{m}_1$ (resp. $\mathbf{v}_k + \mathbf{m}_2$) to $Cand_k$ if InPlane($\mathbf{v}_k + \mathbf{m}_1$) (resp. InPlane($\mathbf{v}_k + \mathbf{m}_2$));
if InPlane($\mathbf{v}_k + \mathbf{m}_1$) *and* InPlane($\mathbf{v}_k + \mathbf{m}_2$) **then**
    $(\mathbf{u}, \mathbf{w}) \leftarrow (\mathbf{m}_1, \mathbf{m}_2)$;
    while $\mathbf{u} \cdot \mathbf{w} < 0$ **do**
        if $\mathbf{u} \cdot (\mathbf{u} + \mathbf{w}) \leq 0$ *or* $\mathbf{w} \cdot (\mathbf{u} + \mathbf{w}) \leq 0$ **then**
            if $\mathbf{u} \cdot (\mathbf{u} + \mathbf{w}) \leq 0$ **then**
                $(\mathbf{u}, \mathbf{w}) \leftarrow (\mathbf{w}, \mathbf{u})$;
            Compute $\gamma = \left\lfloor \frac{-\mathbf{u} \cdot \mathbf{w}}{\|\mathbf{w}\|^2} \right\rfloor$;
            if $(\mathbf{u} + \gamma\mathbf{w}) \cdot (\mathbf{u} + (\gamma + 1)\mathbf{w}) < 0$ **then**
                if InPlane($\mathbf{u} + \gamma\mathbf{w}$) **then**
                    Add $\mathbf{v}_k + \mathbf{u} + \gamma\mathbf{w}$ to $Cand_k$;
                    $(\mathbf{u}, \mathbf{w}) \leftarrow (\mathbf{w}, \mathbf{u} + \gamma\mathbf{w})$;
                else
                    Find a closest point $\mathbf{x}^\star \in \{\mathbf{v}_k + \mathbf{u} + c\mathbf{w}\}_{0 \leq c \leq \gamma+1}$ such that InPlane($\mathbf{x}^\star$) and add it to $Cand_k$; **break**;
            else
                Find a closest point $\mathbf{x}^\star \in \{\mathbf{v}_k + \mathbf{u} + c\mathbf{w}\}_{0 \leq c \leq \gamma-1}$ such that InPlane($\mathbf{x}^\star$) and add it to $Cand_k$; **break**;
        else
            Add $\mathbf{v}_k + \mathbf{u} + \mathbf{w}$ to $Cand_k$ if InPlane($\mathbf{v}_k + \mathbf{u} + \mathbf{w}$); **break**;
return $Cand_k$;

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

	n	$\mathcal{N}^i_{call}$		$\sum_{i=0}^{n-1} \mathcal{N}^i_{call}$
alg.	avg.	avg.	max.	avg.
H	25.3756	6.00	6	152.25
R	19.2534	17.73	25	271.31
R^1	19.2534	9.77	15	131.23
L	19.2529	12.03	21	144.85

TABLE 3.1: Statistics of plane-probing algorithms (on planes whose normal is in χ). $\mathcal{N}^i_{call}$ denotes the number of calls to predicate at a step i and n is the number of steps.

3.5 Overall complexity and performance

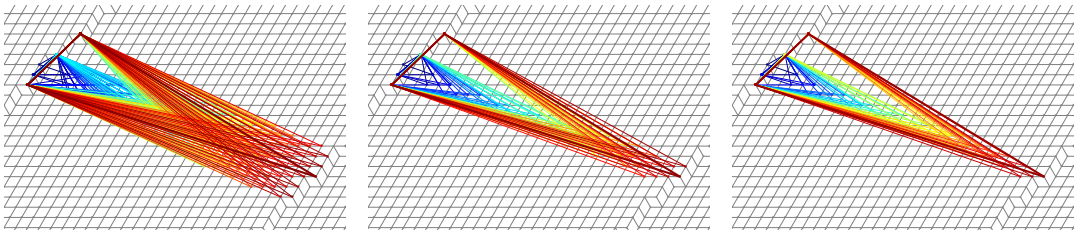
Thanks to Theorem 3.1, we obtain the global complexity of the L-algorithm:

Corollary 3.1. *For L-algorithm, the total number of predicate calls is in* $O(\|\mathbf{N}\|_1 \log \|\mathbf{N}\|_1)$.

Proof. There are $O(\|\mathbf{N}\|_1)$ steps (see Sec. 2.5) and $O(\log \|\mathbf{N}\|_1)$ calls to the predicate at every step due to Algorithm 3 (Theorem 3.1). □

For comparison, the total number of predicate calls in worst cases is in $O(\|\mathbf{N}\|_1)$ for the H-algorithm, $O(\|\mathbf{N}\|_1 \log \|\mathbf{N}\|_1)$ for the R-algorithm [LPR17] and $O(\|\mathbf{N}\|_1)$ for the R^1-algorithm [RL19].

To provide more statistics, we have considered a large collection of implicit digital planes with normal vectors in a set χ with relatively prime components, in the range $(1, 1, 1)$ to $(200, 200, 200)$. The cardinal of the set χ is 6578833. For all variants of *tetrahedron-based* plane-probing algorithms, we compare in Tab. 3.1: the number n of steps, the number $\mathcal{N}^i_{call}$ of calls to the predicate per iteration and the total number of calls $\sum_{i=0}^{n-1} \mathcal{N}^i_{call}$. The results are obtained from a C++ implementation using the DGtal Library [The10]. The numbers do not perfectly match with the table shown in [RL19] due to different implementation choices (*e.g.*, the ordering in case of co-spherical points).



(A) (B) (C)

FIGURE 3.6: The evolution for normal $(2, 5, 156)$ with H-algorithm (a), R-algorithm (b) and L-algorithm (c). Every triangle of the evolution is superimposed. The initial triangle is blue. The last one is red.

In terms of the number of steps, the L-algorithm requires a fewer number of steps to obtain the exact normal vector of the plane. In particular, we observe that the steps of L-algorithm are included in the ones of R-algorithm for all vectors in χ. For

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

example, for the digital plane of normal vector $\mathbf{N} = (2, 5, 6)$ (see Fig. 3.6), the L-algorithm uses 40 steps while the R-algorithm uses 50 steps to find the exact normal vector. Visibly, the H-algorithm requires more steps.

We also remark that, in practice, the L-algorithm usually examines fewer points at each step than the R-algorithm. However, it does not outperform the R^1-algorithm, the optimized version of R-algorithm. The experimental results are shown in tab. 3.1.

In Fig. 3.7, we compare the complexity of different variants with a family of normal vectors $\{(3, 19, r), 1 \leq r \leq 500\}$. We measure the number of calls to the predicate InPlane at each iteration. In the upper image, we divide the number by the number of the iteration for each normal vector and the curves correspond to the statement in Theorem 3.1. In the bottom image, we look at the sum of the numbers of calls of all iterations. We remark that for this particular example, the H-algorithm requires a constant number of call to the predicate but the total number of calls might be more than other variants because it needs more steps.

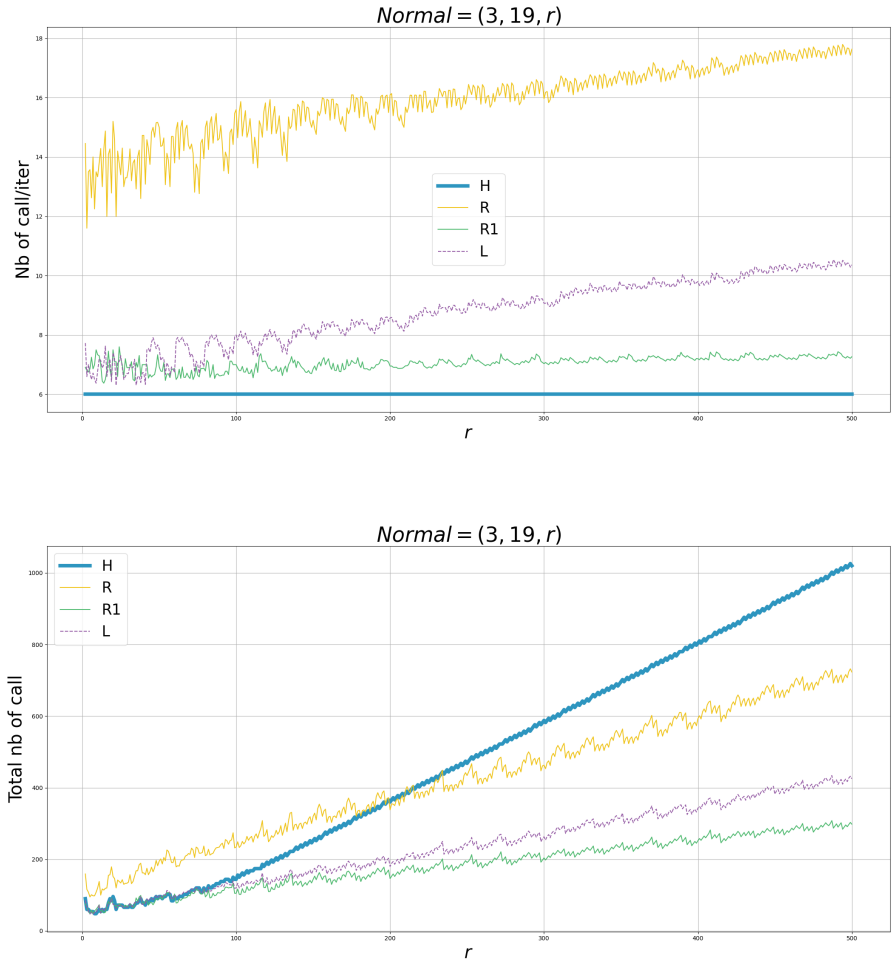


FIGURE 3.7: Number of calls to predicate (upper: per iteration; bottom: in total) for normal vectors of form $\{(3, 19, r), 1 \leq r \leq 500\}$.

Furthermore, there is a possible improvement for the implementation. In fact, the H-neighborhood is included in the R-neighborhood, and the latter is in the L-neighborhood

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

(sec. 2.3.2). In the computation for the Fig. 3.7, no matter which kind of neighborhood is chosen, the six points of the H-neighborhood are necessarily probed at each iteration. However, some of the points remain inside the H-neighborhood after the vertex update. Or, to put it differently, the H-neighborhood of two consecutive steps have points in common. Therefore, instead of probing all of them repeatedly at each iteration, one can use a cache so as not to probe twice the same point.

Fig. 3.8 compares the number of predicate calls for different plane-probing algorithms in a simple family of digital planes. The figure also shows the result of an optimized variant of the L-algorithm, denoted L-opt, that decreases the number of calls at each step by some values that are bounded by a constant. We remark that this optimization is not reserved for the L-algorithm, it can also be applied to the H-algorithm and the R-algorithm.

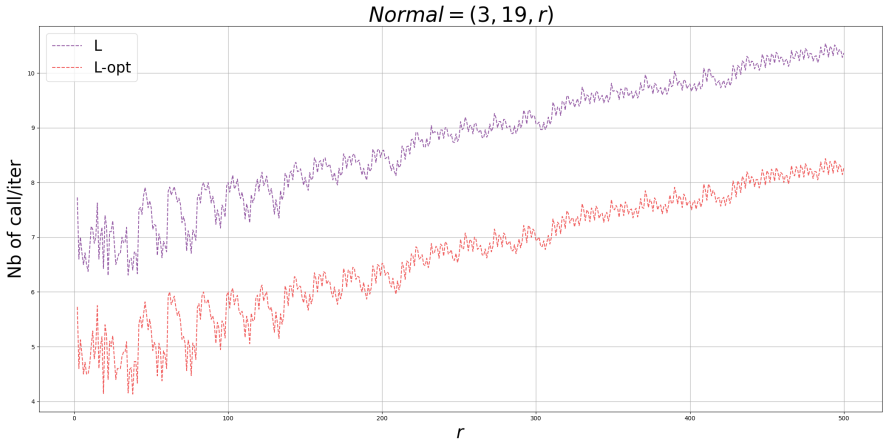


FIGURE 3.8: Number of calls to predicate per iteration for $\mathbf{N} \in \{(3, 19, r), 1 \leq r \leq 500\}$.

3.6 Properties

Besides comparing the complexity among the different variants of plane-probing algorithms, we also carry out several experiments on some geometrical properties. Particularly, there are two of them worth mentioning:

The Delaunay property Here, we consider the set of normal vector of coprime coordinates between $(1, 1, 1)$ and $(80, 80, 80)$, and the results are shown in tab. 3.2. At each step, the algorithm updates one of the three vertices of the base triangle. Thus, the cardinal of the union of the vertices of two consecutive triangles is equal to four. Let $\mathcal{B}^{(i)}$ be the ball determined by the four distinct points of $\mathbf{T}^{(i-1)} \cup \mathbf{T}^{(i)}$. In tab. 3.2, we compute the number of points $C_{\mathbf{N}}$ that are found both in the digital plane and in the ball $\mathcal{B}^{(i)}$. No points are found in any balls for the L-algorithm while 75235972 points are found for H-algorithm and 424 points for R-algorithm.

The fact that there is no point found for the L-algorithm is a very interesting property. We name it the *Delaunay* property and we will prove that it is indeed an invariant for the L-algorithm in the next chapter.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

	$\mathbf{N}$ s.t. $C_{\mathbf{N}} > 0$	$C_{\mathbf{N}}$	
alg.	total	total	avg.
H	247457	75235972	471.46
R	90	424	2.44
L	0	0	0

TABLE 3.2: Statistics of plane-probing algorithms (on planes whose normal is inbetween $(1, 1, 1)$ and $(80, 80, 80)$). $C_{\mathbf{N}}$ denotes the number of points lying both in $\mathbf{P}$ and in $\mathcal{B}^{(i)}$, the closed ball that passes through the vertices of two consecutive triangles.

Non-decreasing ball radius We consider the same collection of implicit digital planes, as in the previous subsection, with normal vectors in a set χ with relatively prime components, in the range $(1, 1, 1)$ to $(200, 200, 200)$. In addition, we tested for all normal vectors in the set χ that, for the L-algorithm, the sequence of radii of the balls $\left\{\mathcal{B}^{(i)}\right\}_{0 \leq i \leq n-1}$ is non-decreasing.

In fig. 3.9, we show an example where the balls' radius are not monotone for the H-algorithm and R^1-algorithm.

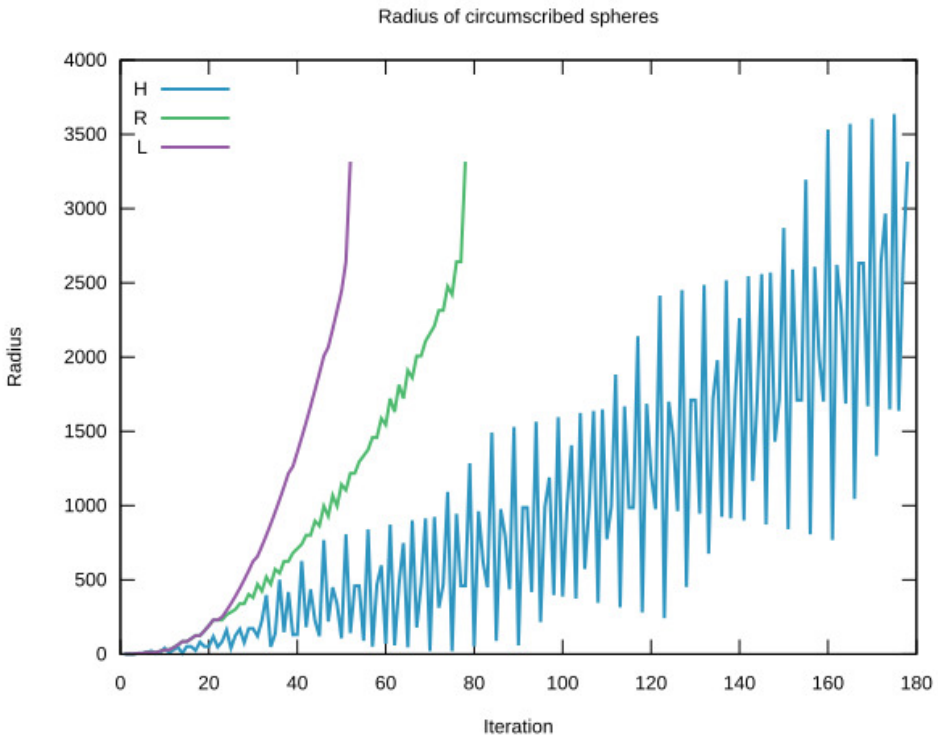


FIGURE 3.9: Radius of $\mathcal{B}^{(i)}$ for $\mathbf{N} = (198, 195, 193)$ during probings with the H, the R^1 and our L-algorithm.

This is also a property that is true only for the L-algorithm. We postpone its proof also to the next chapter (see sec. 4.3).

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

3.7 Conclusion

In this chapter, we have introduced the L-algorithm, which is a new *tetrahedron-based* plane-probing algorithm. It considers a larger neighborhood compared to previous methods. Efficiently probing this expanded neighborhood was a key focus of our discussion, ensuring that the algorithm can be implemented effectively. We also delved into the complexity analysis of the L-algorithm, which helps us understand how it performs in terms of computational resources.

Additionally, we discussed invariant properties of the algorithm, which are characteristics that remain consistent throughout its execution. In the upcoming chapter, we will dive deeper into one specific invariant property known as the Delaunay property, further enhancing our understanding of the L-algorithm and its applications.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

CHAPTER 4

Delaunay property

One major property that the L-algorithm satisfies is the Delaunay property (see property 4.1). In this chapter, we introduce this property and prove that it is an invariant property for the L-algorithm. The majority of the work conducted in this chapter has been organized and presented in a paper [Lu+23] that we have submitted to the journal, Theoretical Computer Science.

For convenience let $\mathbf{T}^{(-1)}$ denotes the degenerated triangle whose three vertices are all at $\mathbf{o}$. For all $i \in \{0, \dots, n\}$, let $\mathcal{B}^{(i)}$ be the ball uniquely determined by the four distinct points of $\mathbf{T}^{(i-1)} \cup \mathbf{T}^{(i)}$. In previous chapter, we have experimentally observed that the following property is verified by the L-Algorithm, but neither by the H-Algorithm, nor by the R-Algorithm:

Property 4.1 (Delaunay property for plane-probing algorithms)**.** *For all $i \in \{0, \dots, n\}$, the ball $\mathcal{B}^{(i)}$ does not contain any point of $\mathbf{P}$ in its interior.*

If a point is randomly chosen in the set $\mathcal{N}_{S_L}^{(i)} \cap \mathbf{P}$ at each iteration, our procedure would still terminate and return a triangle whose normal is equal to the normal of the plane. However, that triangle might possess a significantly bad aspect ratio, with vertices potentially distant from the initial point. The Delaunay property is noteworthy as it ensures both minimal basis and computational locality for the L-algorithm. It is a strong geometric result which ensures that the last triangle has only acute or right angles (Corollary 5.1, in chapter 5), and that its vertices stay close enough to the starting point (Sec. 5.3, in chapter 5).

The main purpose of this chapter is to prove the following theorem:

Theorem 4.1. *The L-algorithm verifies the Delaunay property (Property 4.1).*

Remind that for all $i \in \{0, \dots, n\}$, $\mathcal{H}_+^{(i)}$ (resp. $\mathcal{H}_-^{(i)}$) is the open half-space delimited by $\mathbf{T}^{(i)}$ that contains (resp. does not contain) the neighborhood $\mathcal{N}_{S_L}^{(i)}$.

The proof of Theorem 4.1 requires the following lemma whose quite technical and lengthy proof is postponed to Sec. 4.1:

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Lemma 4.1. *For all $i \in \{0, \dots, n-1\}$, if the interior of $\mathcal{B}^{(i)}$ contains no point of $\mathbf{P}$, then the interior of $\mathcal{B}^{(i+1)}$ contains no point of $\mathbf{P} \cap \mathcal{H}_+^{(i)}$.*

By assuming Lemma 4.1 as true, we can establish the proof of Theorem 4.1 through an inductive approach.

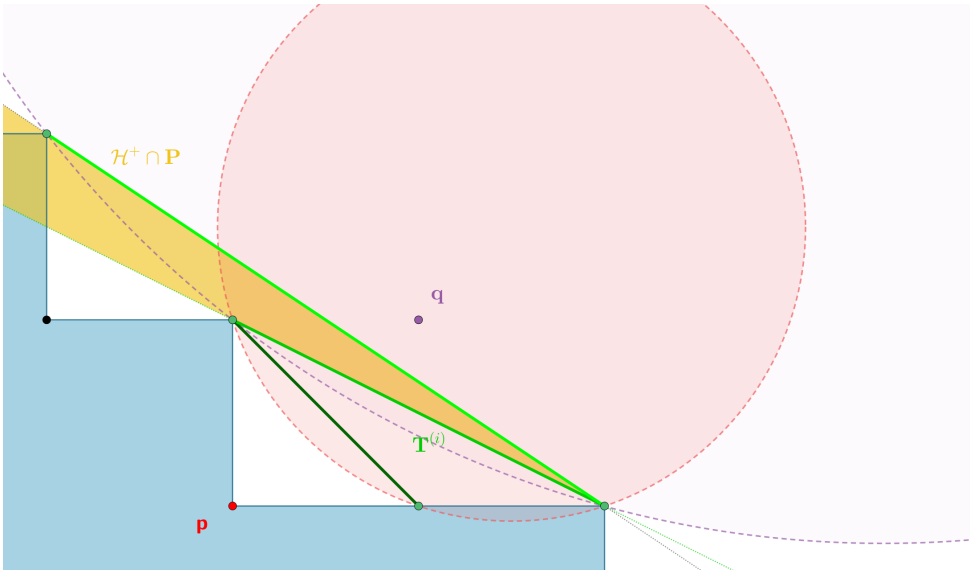


FIGURE 4.1: An 2D representation of an induction step: The (dark to light) green segments indicates the triangles at each step. The last triangle (lightest green) aligns with the digital plane $\mathbf{P}$. $\mathcal{H}_+^{(i)}$ correspond to the half-plane above the triangle $\mathbf{T}^{(i)}$. The yellow region indicates the intersection $\mathcal{H}_+^{(i)} \cap \mathbf{P}$. The balls $\mathcal{B}^{(i)}$ and $\mathcal{B}^{(i+1)}$ are respectively in red and purple.

Proof. of Theorem 4.1

Base case $\mathcal{B}^{(0)}$, which passes through all the vertices of a unit cube, contains no integer point in its interior and as a consequence, no point of $\mathbf{P}$.

Induction step We assume that $\mathcal{B}^{(i)}$ contains no point of $\mathbf{P}$ in its interior for any $i \in \{0, \dots, n-1\}$ and we want to show that no point of $\mathbf{P}$ lies in the interior of $\mathcal{B}^{(i+1)}$. Fig. 4.1 serves as visual aid.

By definition, the boundary of $\mathcal{B}^{(i)}$ and the boundary of $\mathcal{B}^{(i+1)}$ pass through the vertices of $\mathbf{T}^{(i)}$ and there is a point $\mathbf{x}^\star$, chosen by the algorithm, lying in $\mathcal{H}_+^{(i)}$ and such that $\mathbf{x}^\star \in \mathbf{T}^{(i+1)} \setminus \mathbf{T}^{(i)}$.

First, we can safely discard the points of $\mathbf{P}$ that are located in $\mathcal{H}_-^{(i)}$. Indeed, $\mathbf{x}^\star \in \mathcal{H}_+^{(i)}$ (by definition) and $\mathbf{x}^\star \notin \mathcal{B}^{(i)}$ (by hypothesis) together imply that $(\mathcal{B}^{(i)} \cap \mathcal{H}_+^{(i)}) \subseteq (\mathcal{B}^{(i+1)} \cap \mathcal{H}_+^{(i)})$, thus $(\mathcal{B}^{(i+1)} \cap \mathcal{H}_-^{(i)}) \subseteq (\mathcal{B}^{(i)} \cap \mathcal{H}_-^{(i)})$ (see Lemma 3.8). We conclude that the interior of $(\mathcal{B}^{(i+1)} \cap \mathcal{H}_-^{(i)})$ contains no point of $\mathbf{P}$, because it is included in the interior of $(\mathcal{B}^{(i)} \cap \mathcal{H}_-^{(i)})$, itself included in the interior of $\mathcal{B}^{(i)}$, which is assumed to contain no point of $\mathbf{P}$.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

If we denote by $\mathcal{H}^{(i)}$ the plane containing $\mathbf{T}^{(i)}$, we can similarly show that the interior of $(\mathcal{B}^{(i+1)} \cap \mathcal{H}^{(i)})$ contains no point of $\mathbf{P}$, because $(\mathcal{B}^{(i+1)} \cap \mathcal{H}^{(i)}) = (\mathcal{B}^{(i)} \cap \mathcal{H}^{(i)})$ by the definition $\mathcal{B}^{(i)}$ and $\mathcal{B}^{(i+1)}$.

Finally, regarding the points of $\mathbf{P}$ that are located in $\mathcal{H}_+^{(i)}$, by Lemma 4.1, we know that none of them are in the interior of $\mathcal{B}^{(i+1)}$, which concludes. □

4.1 Proof of Lemma 4.1

This section is dedicated to the proof of lemma 4.1. For a fixed step i, the main idea is to partition the points of $\mathcal{H}_+^{(i)}$ into different categories according to their position and we treat each case with distinct lemmas before concluding.

Let us consider the parallelepiped that is generated from the point $\mathbf{q}$ and the vectors $\left\{-\mathbf{m}_k^{(i)}\right\}_{k\in\{0,1,2\}}$. We first introduce the following lemma that shows that the lowest point of the parallelogram is always in the digital plane.

Lemma 4.2. *For all* $i \in \{0,\dots,n\}$, $\mathbf{p}^{(i)} \cdot \mathbf{N} \geq 0$, *with* $\mathbf{p}^{(i)} := \mathbf{q} - \sum_k \mathbf{m}_k^{(i)}$.

Proof. By definition $\mathbf{p}^{(0)} = \mathbf{o}$ and $\mathbf{o}$ is assumed to belong to $\mathbf{P}$. As a consequence, $\mathbf{p}^{(0)} \cdot \mathbf{N} \geq 0$.

For any $i \in \{1,\dots,n-1\}$, there is (k,α,β) such that $\mathbf{m}_k^{(i+1)} = \mathbf{m}_k^{(i)} - \alpha\mathbf{m}_{k+1}^{(i)} - \beta\mathbf{m}_{k+2}^{(i)}$, $\mathbf{m}_{k+1}^{(i+1)} = \mathbf{m}_{k+1}^{(i)}$ and $\mathbf{m}_{k+2}^{(i+1)} = \mathbf{m}_{k+2}^{(i)}$ by (2.9). Then, we remark that $\mathbf{p}^{(i+1)} \cdot \mathbf{N} - \mathbf{p}^{(i)} \cdot \mathbf{N} = \alpha\mathbf{m}_{k+1}^{(i)} \cdot \mathbf{N} + \beta\mathbf{m}_{k+2}^{(i)} \cdot \mathbf{N}$, which is strictly positive by Theorem 2.1. We can therefore conclude by induction. □

Since we now focus on a step $i \in \{0,\dots,n-1\}$, for sake of simplicity, we drop the exponent (i) in the notations of this section.

4.1.1 Outline of the proof and notations

Remind that $\mathbf{p}$ is equal to $\mathbf{q} - \sum_k \mathbf{m}_k$. We conveniently describe any integer point $\mathbf{y} \in \mathbb{Z}^3$ as a linear combination of $\mathbf{m}_0$, $\mathbf{m}_1$ and $\mathbf{m}_2$, which form a basis of $\mathbb{Z}^3$ (by Theorem 2.2), i.e., $\mathbf{y} := \mathbf{p} + \sum_k c_k\mathbf{m}_k$, with $c_k \in \mathbb{Z}$ for all $k \in \mathbb{Z}/3\mathbb{Z}$. By construction, the bounding plane of $\mathcal{H}_+$ is defined by the vertices $\{\mathbf{p}+\mathbf{m}_0+\mathbf{m}_1, \mathbf{p}+\mathbf{m}_1+\mathbf{m}_2, \mathbf{p}+\mathbf{m}_0+\mathbf{m}_2\}$. All lattice points $\mathbf{y}$ on this plane are such that $\sum_k c_k = 2$. Hence, for any lattice point $\mathbf{y}$, we have $\mathbf{y} \in \mathcal{H}_+ \Leftrightarrow \sum_k c_k \geq 3$. In this section, we always assume that $\mathbf{y} \in \mathcal{H}_+$.

We consider several cases:

(Case 1) the coefficients c_0, c_1, c_2 are all strictly positive (see Lemma 4.3),

(Case 2) one coefficient is zero and the others are strictly positive; these points are exactly the ones probed in the L-algorithm (see also the definition of the candidate points, equation (3.1)),

(Case 3) one coefficient is strictly negative and the others are strictly positive (see Lemma 4.4 and Lemma 4.5),

(Case 4) one coefficient is strictly positive and the others are strictly negative or null (see Lemma 4.6 and Lemma 4.7).

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

To check that any $\mathbf{y} \in \mathcal{H}_+$ is in one of the previous cases, it is enough to consider the partition of $\mathbb{Z}^3$ into eight octants depending on the signs of the coefficients and with a convention for null coefficients (see Fig. 4.2). The *negative* octant, in red, does not intersect $\mathcal{H}_+$ and is therefore discarded. The *positive* octant is itself divided into two regions, the interior, in yellow, corresponds to (Case 1), whereas the boundary faces, in green, correspond to (Case 2). Among the last six octants, three of them, in blue, correspond to (Case 3), whereas the other three, in purple, correspond to (Case 4).

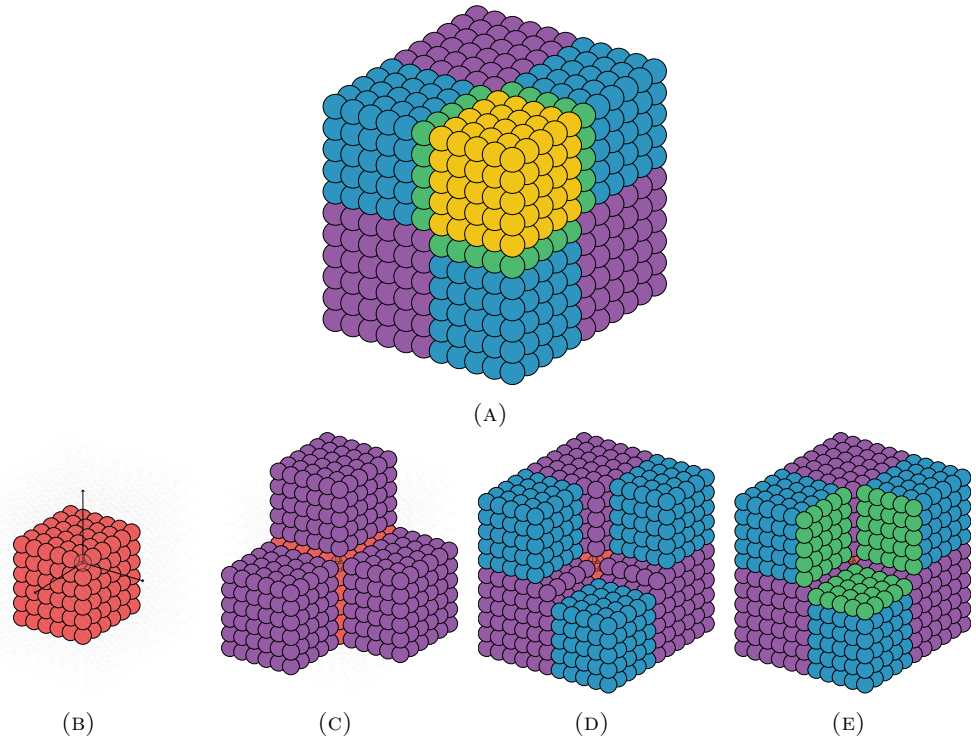


FIGURE 4.2: The discrete space $\mathbb{Z}^3$ (intersected with the box $[-5, 5]^3$ for the illustration) is partitioned into five regions: the yellow, green, blue and purple regions respectively correspond to (Case 1), (Case 2), (Case 3) and (Case 4), the red one is discarded because none of its points lie in $\mathcal{H}_+$ (the three black arrows indicate the direction of the grid axes).

The proofs of the following lemmas require a lot of technical details that are postponed in Sec. 4.2 for the sake of readability. They also require the following notation (see Fig. 4.3):

$$\forall k \in \mathbb{Z}/3\mathbb{Z},\ \mathbf{d}_k := \mathbf{m}_{k+1} - \mathbf{m}_{k+2} = \mathbf{v}_{k+2} - \mathbf{v}_{k+1}. \tag{4.1}$$

For sake of clarity, we use the bar notation whenever a scalar product with $\mathbf{N}$ is required, i.e., $\overline{\mathbf{y}}$ instead of $\mathbf{y} \cdot \mathbf{N}$ for any vector $\mathbf{y} \in \mathbb{Z}^3$.

Lemma 4.2 ensures that $\overline{\mathbf{p}} \geq 0$. Since $\forall k \in \mathbb{Z}/3\mathbb{Z}$, $\overline{\mathbf{m}}_k > 0$ by Theorem 2.1, all points of the form $\mathbf{p} + \sum_k c_k \mathbf{m}_k$ with positive coefficients are such that $\overline{\mathbf{p}} + \sum_k c_k \overline{\mathbf{m}}_k > 0$. That is why we will only check if $\overline{\mathbf{p}} + \sum_k c_k \overline{\mathbf{m}}_k < \|\mathbf{N}\|_1$, whenever we want to determine whether such a point is in $\mathbf{P}$ or not.

Finally, let Σ be the set of all permutations over $\{0, 1, 2\}$. Permutations will be useful to describe in a uniform way the various sign combinations of the coefficients.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

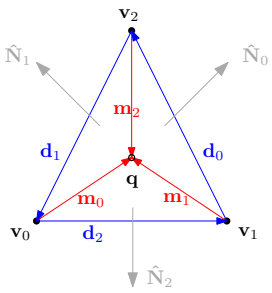


FIGURE 4.3: Notations for $\mathbf{v}_k$ (black), $\mathbf{m}_k$ (red) and $\mathbf{d}_k$ (blue). Note that $\hat{\mathbf{N}}_k$ (grey) is used only in Sec. 4.2.1.

4.1.2 (Case 1)

The following lemma indicates that the points $\mathbf{y}$ corresponding to (Case 1) do not need to be considered because they are not in $\mathbf{P}$.

Lemma 4.3. *Let* $\mathbf{y} = \mathbf{p} + \sum_k c_k \mathbf{m}_k$ *be such that* $\sum_k c_k \geq 3$. *If* $c_0, c_1, c_2 > 0$, *then* $\mathbf{y} \notin \mathbf{P}$.

Proof. Note that $\mathbf{y} = \mathbf{q} + \sum_k (c_k - 1)\mathbf{m}_k$. Since $(c_k - 1) \geq 0$ for all $k \in \mathbb{Z}/3\mathbb{Z}$ (by hypothesis), $\overline{\mathbf{q}} = \|\mathbf{N}\|_1$ (by definition) and $\overline{\mathbf{m}}_k > 0$ (by Theorem 2.1), then $\overline{\mathbf{y}} = \overline{\mathbf{q}} + \sum_k (c_k - 1)\overline{\mathbf{m}}_k \geq \overline{\mathbf{q}} = \|\mathbf{N}\|_1$ and $\mathbf{y} \notin \mathbf{P}$. □

4.1.3 (Case 3)

This section contains Lemma 4.4 and Lemma 4.5 that focus on (Case 3). More precisely, they indicate that the points $\mathbf{y}$ corresponding to (Case 3) do not need to be considered because if they are in $\mathbf{P}$, then there is at least one specific point $\mathbf{x} \in \mathcal{N}_S \cap \mathbf{P}$ (Lemma 4.4) such that $\mathbf{x} \leq_{\mathbf{T}} \mathbf{y}$ (Lemma 4.5).

Lemma 4.4. *Let* $\mathbf{y} = \mathbf{p} + \sum_k c_k \mathbf{m}_k$ *be such that* $\sum_k c_k \geq 3$. *Let* $\sigma \in \Sigma$ *be such that* $c_{\sigma(0)} < 0$ *and* $c_{\sigma(1)}, c_{\sigma(2)} > 0$. *If* $\mathbf{y} \in \mathbf{P}$, *then* $\mathbf{p} + 2\mathbf{m}_{\sigma(1)} + \mathbf{m}_{\sigma(2)} \in \mathbf{P}$ *or* $\mathbf{p} + \mathbf{m}_{\sigma(1)} + 2\mathbf{m}_{\sigma(2)} \in \mathbf{P}$ *(the two points can be both in* $\mathbf{P}$*).*

In addition, if $-c_{\sigma(0)} < \min(c_{\sigma(1)}, c_{\sigma(2)}) - 1$, *then* $\mathbf{p} + (c_{\sigma(0)} + c_{\sigma(1)})\mathbf{m}_{\sigma(1)} + (c_{\sigma(0)} + c_{\sigma(2)})\mathbf{m}_{\sigma(2)} \in \mathbf{P}$.

Proof. We assume w.l.o.g. that σ is the identity, i.e., $\sigma(0) = 0$, $\sigma(1) = 1$ and $\sigma(2) = 2$.

Since $\mathbf{y} \in \mathbf{P}$, we have

$$\overline{\mathbf{y}} = \overline{\mathbf{p}} + \sum_k c_k \overline{\mathbf{m}}_k = \overline{\mathbf{q}} + \sum_k (c_k - 1)\overline{\mathbf{m}}_k < \|\mathbf{N}\|_1.$$

Since $\overline{\mathbf{q}} = \|\mathbf{N}\|_1$, the last inequality is equivalent to $\sum_k (c_k - 1)\overline{\mathbf{m}}_k < 0$.

With h set to $\min(\overline{\mathbf{m}}_1, \overline{\mathbf{m}}_2)$ and noticing that $c_0 < 0 \Leftrightarrow -(c_0 - 1) > 1$, we equivalently have

$$\frac{(c_1 + c_2 - 2)h}{-(c_0 - 1)} \leq \frac{(c_1 - 1)\overline{\mathbf{m}}_1 + (c_2 - 1)\overline{\mathbf{m}}_2}{-(c_0 - 1)} < \overline{\mathbf{m}}_0.$$

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

In addition, we have

$$\sum_k c_k \geq 3 \Leftrightarrow c_1 + c_2 - 2 \geq -c_0 + 1,$$

which means that $h < \overline{\mathbf{m}}_0$.

We conclude that if $h = \overline{\mathbf{m}}_1$ (resp. $h = \overline{\mathbf{m}}_2$), $\overline{\mathbf{p}} + 2\overline{\mathbf{m}}_1 + \overline{\mathbf{m}}_2$ (resp. $\overline{\mathbf{p}} + \overline{\mathbf{m}}_1 + 2\overline{\mathbf{m}}_2$) is strictly smaller than $\overline{\mathbf{p}} + \sum_k \overline{\mathbf{m}}_k = \overline{\mathbf{q}} = \|\mathbf{N}\|_1$ and thus, the point $\mathbf{p} + 2\mathbf{m}_1 + \mathbf{m}_2$ (resp. $\mathbf{p} + \mathbf{m}_1 + 2\mathbf{m}_2$) is in $\mathbf{P}$.

For the second part, we similarly derive from $\sum_k (c_k - 1)\overline{\mathbf{m}}_k < 0$:

$$\frac{(\min(c_1, c_2) - 1)}{-(c_0 - 1)}(\overline{\mathbf{m}}_1 + \overline{\mathbf{m}}_2) \leq \frac{(c_1 - 1)\overline{\mathbf{m}}_1 + (c_2 - 1)\overline{\mathbf{m}}_2}{-(c_0 - 1)} < \overline{\mathbf{m}}_0.$$

Since we assume $(\min(c_1, c_2) - 1) > -c_0$, we have $\frac{(\min(c_1, c_2) - 1)}{-(c_0 - 1)} \geq 1$ and it follows that $(\overline{\mathbf{m}}_1 + \overline{\mathbf{m}}_2) < \overline{\mathbf{m}}_0$.

As a consequence,

$$\overline{\mathbf{p}} + (c_0 + c_1)\overline{\mathbf{m}}_1 + (c_0 + c_2)\overline{\mathbf{m}}_2 < \overline{\mathbf{p}} + \sum_k c_k \overline{\mathbf{m}}_k = \overline{\mathbf{y}} < \|\mathbf{N}\|_1,$$

which concludes. □

Lemma 4.5. *Let* $\mathbf{y} = \mathbf{p} + \sum_k c_k \mathbf{m}_k$ *be such that* $\sum_k c_k \geq 3$. *Let* $\sigma \in \Sigma$ *be such that* $c_{\sigma(0)} < 0$ *and* $c_{\sigma(1)}, c_{\sigma(2)} > 0$. *If* $\mathbf{y} \in \mathbf{P}$ *and if the interior of* $\mathcal{B}$ *contains no point of* $\mathbf{P}$, *then there exists a point* $\mathbf{x} \in \mathcal{N}_S \cap \mathbf{P}$ *such that* $\mathbf{x} \leq_{\mathbf{T}} \mathbf{y}$.

Proof. We assume w.l.o.g. that σ is the identity. We also assume w.l.o.g. that $c_1 \leq c_2$ and consider three separate cases (see Fig. 4.4):

(i) $(c_1 - 1) \leq c_2 \leq -c_0$,

(ii) $(c_1 - 1) \leq -c_0 < c_2$,

(iii) $-c_0 < (c_1 - 1) < c_2$.

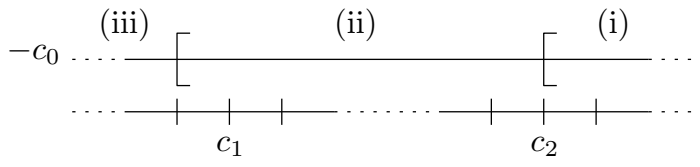


FIGURE 4.4: Relative position of $-c_0$ with respect to c_1 and c_2. Three separate cases, (i), (ii), and (iii), are considered in the proof.

Since $\mathbf{y} \in \mathbf{P}$, either $\mathbf{p} + \mathbf{m}_1 + 2\mathbf{m}_2$ or $\mathbf{p} + 2\mathbf{m}_1 + \mathbf{m}_2$ is in $\mathbf{P}$ by Lemma 4.4. For (i) and (ii), we suppose here that only $\mathbf{p} + \mathbf{m}_1 + 2\mathbf{m}_2 \in \mathbf{P}$, because the case where only $\mathbf{p} + 2\mathbf{m}_1 + \mathbf{m}_2 \in \mathbf{P}$ can be proven symmetrically. For (iii), Lemma 4.4 provides a stronger result, that is $\mathbf{y} \in \mathbf{P}$ implies $\mathbf{p} + (c_0 + c_1)\mathbf{m}_1 + (c_0 + c_2)\mathbf{m}_2 \in \mathbf{P}$, which in turn implies both $\mathbf{p} + \mathbf{m}_1 + 2\mathbf{m}_2 \in \mathbf{P}$ and $\mathbf{p} + 2\mathbf{m}_1 + \mathbf{m}_2 \in \mathbf{P}$ because $(c_0 + c_1) \geq 2$ and $(c_0 + c_2) \geq 2$ ($\overline{\mathbf{p}} + \overline{\mathbf{m}}_1 + 2\overline{\mathbf{m}}_2, \overline{\mathbf{p}} + 2\overline{\mathbf{m}}_1 + \overline{\mathbf{m}}_2 < \overline{\mathbf{p}} + (c_0 + c_1)\overline{\mathbf{m}}_1 + (c_0 + c_2)\overline{\mathbf{m}}_2 < \|\mathbf{N}\|_1$ by Theorem 2.1).

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Let $\mathbf{u} := -\mathbf{m}_0 + \mathbf{m}_1 + \mathbf{m}_2$. By equation (4.1), $\mathbf{u}$ is also equal to $\mathbf{d}_1 + \mathbf{m}_1$. The first step of the proof is to show the following results:

$$\mathbf{p} + \mathbf{m}_1 + 2\mathbf{m}_2 \in \mathbf{P} \Rightarrow \begin{cases} \mathbf{d}_1 \cdot \mathbf{m}_2 \geq 0, & (4.2) \\ \mathbf{m}_2 \cdot \mathbf{u} \geq 0, & (4.3) \\ \mathbf{d}_1 \cdot \mathbf{u} \geq 0, & (4.4) \\ (-\mathbf{d}_2) \cdot \mathbf{u} \geq 0, & (4.5) \end{cases}$$

and

$$\mathbf{p} + 2\mathbf{m}_1 + \mathbf{m}_2 \in \mathbf{P} \Rightarrow \mathbf{m}_1 \cdot \mathbf{u} \geq 0. \quad (4.6)$$

Those results are used in a second step to complete the proof: (4.2), (4.3), (4.4), (4.5) are used in cases (i) and (ii), while (4.3) and (4.6) are used in case (iii).

First step If $\mathbf{p}+\mathbf{m}_1+2\mathbf{m}_2$ is in $\mathbf{P}$, so is $\mathbf{p}+2\mathbf{m}_2$ ($\overline{\mathbf{p}}+2\overline{\mathbf{m}}_2 < \overline{\mathbf{p}}+\overline{\mathbf{m}}_1+2\overline{\mathbf{m}}_2 < \|\mathbf{N}\|_1$ by Theorem 2.1). As the interior of $\mathcal{B}$ does not contain any point of $\mathbf{P}$ by hypothesis, $\mathbf{p} + 2\mathbf{m}_2 \notin \mathcal{B}$. By rewriting

$$\mathbf{p} + 2\mathbf{m}_2 = \mathbf{v}_0 - \mathbf{d}_0 = \mathbf{v}_2 + \mathbf{d}_1 - \mathbf{d}_0\,,$$

we can apply Lemma 4.13 with the two vectors $(-\mathbf{d}_0), \mathbf{d}_1$ and the point $\mathbf{v}_2$ as origin. Since $\mathbf{v}_2$, $\mathbf{v}_2 - \mathbf{d}_0 = \mathbf{v}_1$ and $\mathbf{v}_2 + \mathbf{d}_1 = \mathbf{v}_0$ are indeed on the boundary of $\mathcal{B}$, we get $(-\mathbf{d}_0) \cdot \mathbf{d}_1 \geq 0$. From that, we finally get (4.2) because $(-\mathbf{d}_0) \cdot \mathbf{d}_1 \geq 0$ implies $\mathbf{d}_1 \cdot \mathbf{m}_2 > 0$ by Lemma 4.10.

We can similarly get (4.3) and (4.6). To explain why, we focus on the case where $\mathbf{p} + \mathbf{m}_1 + 2\mathbf{m}_2$ is assumed to be in $\mathbf{P}$ because the other case is symmetric. Note first that $\mathbf{p} + \mathbf{m}_0 \in \mathbf{P}$ (using the same arguments as in the previous paragraph for $\mathbf{p} + 2\mathbf{m}_2$). As a consequence, both $\mathbf{p} + \mathbf{m}_1 + 2\mathbf{m}_2$ and $\mathbf{p} + \mathbf{m}_0$ are not in $\mathcal{B}$ by hypothesis. We can then apply Lemma 4.14 with the two vectors $\mathbf{m}_2, (-\mathbf{u})$ and the point $\mathbf{v}_0$ as origin. Since $\mathbf{v}_0$ and $\mathbf{v}_0 + \mathbf{m}_2 - \mathbf{u} = \mathbf{v}_1$ are indeed on the boundary of $\mathcal{B}$, we get $\mathbf{m}_2 \cdot (-\mathbf{u}) < 0$ and thus (4.3).

By Lemma 4.12, $\mathbf{m}_2 \cdot (\mathbf{d}_1 + \mathbf{m}_1) \geq 0$ implies $\mathbf{d}_1 \cdot (\mathbf{d}_1 + \mathbf{m}_1) > 0$. Since $\mathbf{u} = \mathbf{d}_1 + \mathbf{m}_1$, (4.4) is actually a simple consequence of (4.3).

It remains (4.5), whose proof is separated into two distinct cases.

If $\mathbf{d}_1 \cdot \mathbf{d}_2 \leq 0$, we have $\mathbf{d}_2 \cdot (\mathbf{d}_1 + \mathbf{m}_1) < 0$ and thus (4.5) by Lemma 4.11.

Otherwise, i.e., if $(-\mathbf{d}_1) \cdot \mathbf{d}_2 < 0$, we apply Lemma 4.13 with vectors $(-\mathbf{d}_1)$, $\mathbf{d}_2$ and the point $\mathbf{v}_0$ as origin. Since the points $\mathbf{v}_0$, $\mathbf{v}_0 - \mathbf{d}_1 = \mathbf{v}_2$ and $\mathbf{v}_0 + \mathbf{d}_2 = \mathbf{v}_1$ are on the boundary of $\mathcal{B}$, we deduce that the point $\mathbf{v}_0 - \mathbf{d}_1 + \mathbf{d}_2 = \mathbf{q} - \mathbf{u}$ is necessarily in the interior of $\mathcal{B}$. Moreover, since no point of $\mathcal{B}$ belongs to $\mathbf{P}$, we deduce that $\mathbf{q} - \mathbf{u}$ is not in $\mathbf{P}$.

We have therefore $\overline{\mathbf{q}} - \overline{\mathbf{u}} \geq \|\mathbf{N}\|_1 \Leftrightarrow \overline{\mathbf{u}} \leq 0$. It follows that $\overline{\mathbf{v}}_2 < \|\mathbf{N}\|_1 \Rightarrow \overline{\mathbf{v}}_2 + \overline{\mathbf{u}} < \|\mathbf{N}\|_1$, which means that the point $\mathbf{v}_2+\mathbf{u} = \mathbf{p}+2\mathbf{m}_1+\mathbf{m}_2$ is in $\mathbf{P}$. In this case, we have (4.6) and as a consequence, (4.5), because $\mathbf{m}_1 \cdot (\mathbf{d}_1+\mathbf{m}_1) \geq 0$ implies $\mathbf{d}_2 \cdot (\mathbf{d}_1+\mathbf{m}_1) < 0$ by Lemma 4.12.

Second step

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

(i) We assume first that $(c_1 - 1) \leq c_2 \leq -c_0$. One can check that

$$\begin{aligned}\mathbf{y} &= \mathbf{p} + \sum_k c_k \mathbf{m}_k \\ &= \mathbf{v}_0 + c_0\mathbf{m}_0 + (c_1 - 1)\mathbf{m}_1 + (c_2 - 1)\mathbf{m}_2 \\ &= \mathbf{v}_0 + (-c_0 - c_1 + 1)(\mathbf{d}_1) + (-c_0 - c_2 + 1)(-\mathbf{d}_2) + (\sum_k c_k - 2)\mathbf{u}.\end{aligned}$$

Let $\mathbf{w} := (-c_0 - c_1 + 1)(\mathbf{d}_1) + (-c_0 - c_2 + 1)(-\mathbf{d}_2) + (\sum_k c_k - 2)\mathbf{u}$. All its coefficients, i.e., $(-c_0 - c_1 + 1)$, $(-c_0 - c_2 + 1)$, $(\sum_k c_k - 2)$, are positive by hypothesis. Since we also have (4.4) and (4.5), we can apply Lemma 4.17 to show that $\delta^0_{\mathbf{T}}(\mathbf{m}_2, \mathbf{w}) \geq 0$, which is equivalent to $\mathbf{v}_0 + \mathbf{m}_2 \leq_{\mathbf{T}} \mathbf{v}_0 + \mathbf{w}$ by (3.15). As a result, there exists a point in $\mathcal{N}_S \cap \mathbf{P}$, namely $\mathbf{v}_0 + \mathbf{m}_2$, which is closer than $\mathbf{y}$ according to $\leq_{\mathbf{T}}$.

(ii) We assume now $0 \leq (c_1 - 1) \leq -c_0 < c_2$ and we rewrite $\mathbf{y}$ as another positive linear combination:

$$\begin{aligned}\mathbf{y} &= \mathbf{v}_0 + c_0\mathbf{m}_0 + (c_1 - 1)\mathbf{m}_1 + (c_2 - 1)\mathbf{m}_2 \\ &= \mathbf{v}_0 + (-c_0 - c_1 + 1)(\mathbf{d}_1) + (c_0 + c_2 - 1)(\mathbf{m}_2) + (c_1 - 1)(\mathbf{u}).\end{aligned}$$

By assumptions, all coefficients, i.e., $(-c_0 - c_1 + 1)$, $(c_0 + c_2 - 1)$, $(c_1 - 1)$, are positive. From that and (4.2), (4.3), (4.4), (4.5), we can use Lemma 4.18 to get $\mathbf{v}_0 + \mathbf{m}_2 \leq_{\mathbf{T}} \mathbf{y}$. Again, there exists a point in $\mathcal{N}_S \cap \mathbf{P}$, namely $\mathbf{v}_0 + \mathbf{m}_2$, which is closer than $\mathbf{y}$ according to $\leq_{\mathbf{T}}$.

(iii) We finally assume $0 < -c_0 < (c_1 - 1) < c_2$ and we rewrite $\mathbf{y}$ as:

$$\begin{aligned}\mathbf{y} &= \mathbf{v}_0 + c_0\mathbf{m}_0 + (c_1 - 1)\mathbf{m}_1 + (c_2 - 1)\mathbf{m}_2 \\ &= \mathbf{v}_0 + (c_0 + c_1 - 1)(\mathbf{m}_1) + (c_0 + c_2 - 1)(\mathbf{m}_2) + (-c_0)(\mathbf{u}).\end{aligned}$$

By assumptions, all coefficients, i.e., $(c_0 + c_1 - 1)$, $(c_0 + c_2 - 1)$, $(-c_0)$, are positive. From that and (4.3), (4.6), Lemma 4.19 shows that there exists a point $\mathbf{x} := \mathbf{v}_0 + \alpha\mathbf{m}_1 + \beta\mathbf{m}_2$, with $\alpha, \beta \in \mathbb{N} \setminus (0, 0), \alpha \leq (c_0 + c_1 - 1), \beta \leq (c_0 + c_2 - 1)$, such that $\mathbf{x} \leq_{\mathbf{T}} \mathbf{y}$. To conclude (iii), it remains to check that such a point is in $\mathbf{P}$. Indeed, since $\mathbf{p} + (c_0 + c_1)\mathbf{m}_1 + (c_0 + c_2)\mathbf{m}_2 \in \mathbf{P}$ (Lemma 4.4), we have:

$$\overline{\mathbf{x}} = \overline{\mathbf{p}} + (\alpha + 1)\overline{\mathbf{m}}_1 + (\beta + 1)\overline{\mathbf{m}}_2 \leq \overline{\mathbf{p}} + (c_0 + c_1)\overline{\mathbf{m}}_1 + (c_0 + c_2)\overline{\mathbf{m}}_2 < \|\mathbf{N}\|_1.$$

□

4.1.4 (Case 4)

This section contains Lemma 4.6 and Lemma 4.7 that focus on (Case 4). More precisely, they indicate that the points $\mathbf{y}$ corresponding to (Case 4) do not need to be considered because, as in the previous section, if they are in $\mathbf{P}$, then there is at least one specific point $\mathbf{x} \in \mathcal{N}_S \cap \mathbf{P}$ (Lemma 4.6) such that $\mathbf{x} \leq_{\mathbf{T}} \mathbf{y}$ (Lemma 4.7).

Lemma 4.6. *Let* $\mathbf{y} = \mathbf{p} + \sum_k c_k\mathbf{m}_k$ *be such that* $\sum_k c_k \geq 3$. *Let* $\sigma \in \Sigma$ *be such that* $c_{\sigma(0)}, c_{\sigma(1)} \leq 0$, *then* $\mathbf{y} \in \mathbf{P}$ *implies both:*

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

- $\mathbf{p} + \mathbf{m}_{\sigma(0)} + 2\mathbf{m}_{\sigma(2)} \in \mathbf{P}$ *or* $\mathbf{p} + \mathbf{m}_{\sigma(1)} + 2\mathbf{m}_{\sigma(2)} \in \mathbf{P}$,
- $\mathbf{p} + 2\mathbf{m}_{\sigma(2)} \in \mathbf{P}$.

Proof. We assume w.l.o.g. that σ is the identity.

Since $\mathbf{y} \in \mathbf{P}$, we have

$$\overline{\mathbf{y}} = \overline{\mathbf{p}} + \sum_k c_k \overline{\mathbf{m}}_k = \overline{\mathbf{q}} + \sum_k (c_k - 1)\overline{\mathbf{m}}_k < \|\mathbf{N}\|_1.$$

Since $\overline{\mathbf{q}} = \|\mathbf{N}\|_1$, the last inequality is equivalent to $\sum_k (c_k - 1)\overline{\mathbf{m}}_k < 0$.

With h set to $\max(\overline{\mathbf{m}}_0, \overline{\mathbf{m}}_1)$ and noting that $(c_2 - 1) \geq 2$ (since $\sum_k c_k \geq 3$ and $c_0, c_1 \leq 0$), we equivalently have

$$\overline{\mathbf{m}}_2 < \frac{-(c_0 - 1)\overline{\mathbf{m}}_0 - (c_1 - 1)\overline{\mathbf{m}}_1}{c_2 - 1} < \frac{(-c_0 - c_1 + 2)h}{c_2 - 1}.$$

In addition, we have

$$\sum_k c_k \geq 3 \Leftrightarrow c_2 - 1 \geq -c_0 - c_1 + 2,$$

which means that $\overline{\mathbf{m}}_2 < h$.

We conclude that if $h = \overline{\mathbf{m}}_0$ (resp. $h = \overline{\mathbf{m}}_1$), $\overline{\mathbf{p}} + \overline{\mathbf{m}}_1 + 2\overline{\mathbf{m}}_2$ (resp. $\overline{\mathbf{p}} + \overline{\mathbf{m}}_0 + 2\overline{\mathbf{m}}_2$) is strictly smaller than $\overline{\mathbf{p}} + \sum_k \overline{\mathbf{m}}_k = \overline{\mathbf{q}} = \|\mathbf{N}\|_1$ and thus, the point $\mathbf{p} + \mathbf{m}_1 + 2\mathbf{m}_2$ (resp. $\mathbf{p} + \mathbf{m}_0 + 2\mathbf{m}_2$) is in $\mathbf{P}$. *A fortiori* and whatever h is, $\mathbf{p} + 2\mathbf{m}_2 \in \mathbf{P}$. □

Lemma 4.7. *Let* $\mathbf{y} = \mathbf{p} + \sum_k c_k \mathbf{m}_k$ *be such that* $\sum_k c_k \geq 3$. *Let* $\sigma \in \Sigma$ *be such that* $c_{\sigma(0)}, c_{\sigma(1)} \leq 0$. *If* $\mathbf{y} \in \mathbf{P}$ *and if the interior of* $\mathcal{B}$ *contains no point of* $\mathbf{P}$, *then there exists a point* $\mathbf{x} \in \mathcal{N}_S \cap \mathbf{P}$ *such that* $\mathbf{x} \leq_{\mathbf{T}} \mathbf{y}$.

Proof. We assume w.l.o.g. that σ is the identity.

Since $\mathbf{y} \in \mathbf{P}$, $\mathbf{p} + 2\mathbf{m}_2 \in \mathbf{P}$ by Lemma 4.6. That point, which is also at $\mathbf{v}_0 - \mathbf{d}_0 = \mathbf{v}_1 + \mathbf{d}_1$, is not in the interior of $\mathcal{B}$ by hypothesis and we can apply Lemma 4.13 with the two vectors $(-\mathbf{d}_0), \mathbf{d}_1$ and the point $\mathbf{v}_2$ as origin to get $(-\mathbf{d}_0) \cdot \mathbf{d}_1 \geq 0$.

Furthermore, either $\mathbf{p} + \mathbf{m}_1 + 2\mathbf{m}_2 \in \mathbf{P}$ or $\mathbf{p} + 2\mathbf{m}_1 + \mathbf{m}_2 \in \mathbf{P}$ by Lemma 4.6. We assume below that $\mathbf{p} + \mathbf{m}_1 + 2\mathbf{m}_2 \in \mathbf{P}$, because the case where only $\mathbf{p} + 2\mathbf{m}_1 + \mathbf{m}_2 \in \mathbf{P}$ can be proven symmetrically.

One can check that

$$\begin{aligned}\mathbf{y} = \mathbf{p} + \sum_k c_k \mathbf{m}_k &= \mathbf{v}_0 + c_0\mathbf{m}_0 + (c_1 - 1)\mathbf{m}_1 + (c_2 - 1)\mathbf{m}_2 \\ &= \mathbf{v}_0 + (-c_1 + 1)(-\mathbf{d}_0) + (-c_0)\mathbf{d}_1 + (\sum_k c_k - 2)\mathbf{m}_2.\end{aligned}$$

Let $\mathbf{w} := (-c_1 + 1)(-\mathbf{d}_0) + (-c_0)\mathbf{d}_1 + (\sum_k c_k - 2)\mathbf{m}_2$. All coefficients, i.e., $(-c_1 + 1)$, $(-c_0)$, $(\sum_k c_k - 2)$, are positive. Since, in addition, $(-\mathbf{d}_0) \cdot \mathbf{d}_1 \geq 0$, we can use Lemma 4.20 to show that $\delta^0_{\mathbf{T}}(\mathbf{m}_2, \mathbf{w}) \geq 0$, which is equivalent to $\mathbf{v}_0 + \mathbf{m}_2 \leq_{\mathbf{T}} \mathbf{v}_0 + \mathbf{w}$, where $\mathbf{v}_0 + \mathbf{w} = \mathbf{y}$ and $\mathbf{v}_0 + \mathbf{m}_2 = \mathbf{p} + \mathbf{m}_1 + 2\mathbf{m}_2$. □

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

4.1.5 Conclusion of the proof

Now we have all the material required to prove Lemma 4.1:

Proof. For all $i \in \{0, \dots, n-1\}$, the interior of $\mathcal{B}^{(i)}$ is assumed to contain no point of $\mathbf{P}$.

Let $\mathbf{x}^\star$ be the point chosen by the algorithm at step i, i.e., $\mathbf{x}^\star = \mathbf{T}^{(i+1)} \setminus \mathbf{T}^{(i)}$. We want to show that $\mathbf{x}^\star \leq_{\mathbf{T}} \mathbf{y}$, for all $\mathbf{y} \in \mathbf{P} \cap \mathcal{H}_+^{(i)}$. Let $\mathbf{y}$ be denoted as $\mathbf{p}^{(i)} + \sum_k c_k \mathbf{m}_k^{(i)}$. Note that $\sum_k c_k \geq 3$ because $\mathbf{y} \in \mathcal{H}_+^{(i)}$. Moreover, since $\mathbf{y} \in \mathbf{P}$, the coefficients cannot be all strictly positive by Lemma 4.3.

- if one coefficient is zero and the others are strictly positive (Case 2), then $\mathbf{x}^\star \leq_{\mathbf{T}} \mathbf{y}$ by the design of the algorithm,
- if one coefficient is strictly negative and the others are strictly positive (Case 3), then there exists a point $\mathbf{x} \in \mathcal{N}_S \cap \mathbf{P}$ such that $\mathbf{x} \leq_{\mathbf{T}} \mathbf{y}$ by Lemma 4.5. Then, $\mathbf{x}^\star \leq_{\mathbf{T}} \mathbf{x}$ by the design of the algorithm and $\mathbf{x}^\star \leq_{\mathbf{T}} \mathbf{y}$ by transitivity.
- if one coefficient is strictly positive and the others are strictly negative or null (Case 4), then, similarly, there exists a point $\mathbf{x} \in \mathcal{N}_S \cap \mathbf{P}$ such that $\mathbf{x} \leq_{\mathbf{T}} \mathbf{y}$ by Lemma 4.7. Then, $\mathbf{x}^\star \leq_{\mathbf{T}} \mathbf{x}$ by the design of the algorithm and $\mathbf{x}^\star \leq_{\mathbf{T}} \mathbf{y}$ by transitivity.

Since there is no other possibility, the proof is complete. □

The proof is long and technical. Fig. 4.5 indicates the relationships between each lemmas. The color code follows the partition showed in Fig. 4.2. Respectively, we have yellow for (Case 1), blue for (Case 3) and purple for (Case 4). The mixed color between blue and purple indicates that the lemmas are used for both cases. The gray color is for citation from [LPR17], which we mentioned in Chapter 2.

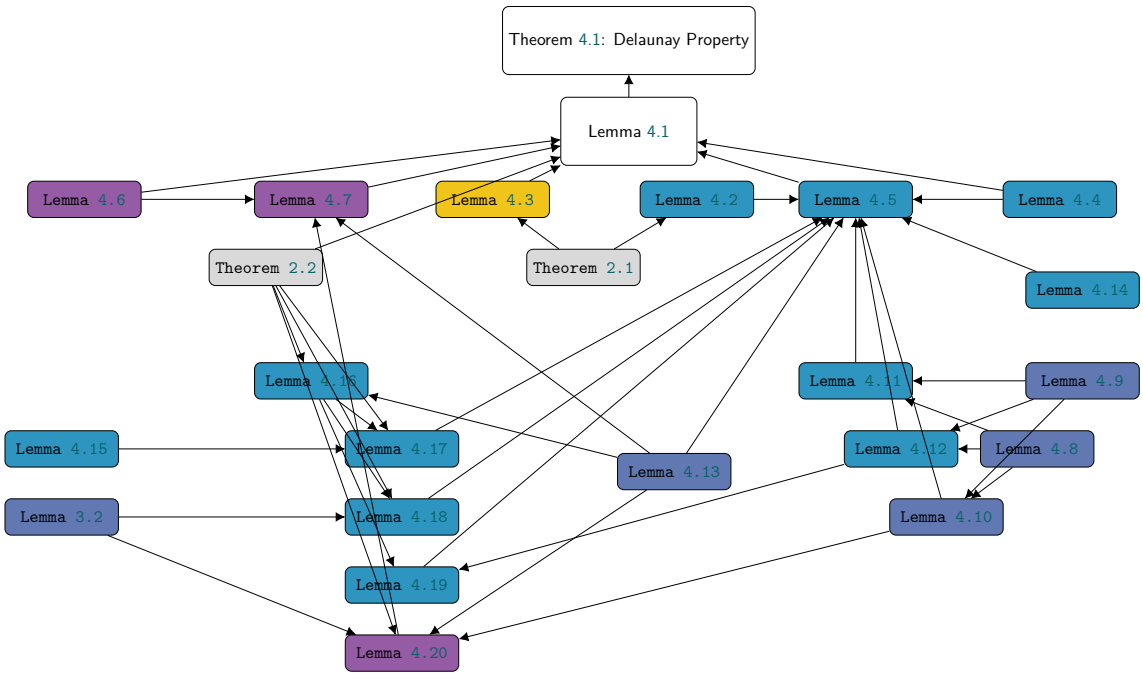


FIGURE 4.5: Lemmas used for the proof of Delaunay property.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

4.2 Technical details

The proof of Lemma 4.1 refers to several technical details that we elaborate in this section. The results are organized into three categories. First, we present, in Sec. 4.2.1, several useful angle relations in the tetrahedron formed by the current triangle and the fixed point $\mathbf{q}$. Then, we present several general and purely geometrical circumsphere-based properties in Sec. 4.2.2, because the relation $\leq_{\mathbf{T}}$ and the selection of a closest point according to $\leq_{\mathbf{T}}$ involves circumspheres. Finally, in sec. 4.2.3, we derive in an algebraic way several other results about the comparison of specific points according to $\leq_{\mathbf{T}}$. These results are used in Lemma 4.5 and Lemma 4.7, which are the main ingredients in the proof of Lemma 4.1.

4.2.1 Projection-based results

Remind that k is taken modulo 3. To keep notations short, we simply write $\forall k$ instead of $\forall k \in \mathbb{Z}/3\mathbb{Z}$ in this section. Let us introduce the following extra notations (see Fig. 4.3):

$$\forall i \in \{0,\dots,n\},\ \begin{cases} \forall k,\ \hat{\mathbf{N}}_k^{(i)} := \mathbf{m}_{k+1}^{(i)} \times \mathbf{m}_{k+2}^{(i)}, \\ \sum_{k\in\{0,1,2\}} \hat{\mathbf{N}}_k^{(i)} =: \hat{\mathbf{N}}(\mathbf{T}^{(i)}). \end{cases} \tag{4.7}$$

Note that the following equality also holds for the estimated normal vector, which is normal to the current triangle:

$$\forall i \in \{0,\dots,n\},\ \forall k,\ \hat{\mathbf{N}}(\mathbf{T}^{(i)}) = \mathbf{d}_k^{(i)} \times \mathbf{d}_{k+1}^{(i)}.$$

Lemma 4.8. *For all* $i \in \{0,\dots,n\}$, $\forall k$, $\hat{\mathbf{N}}_k^{(i)} \cdot \hat{\mathbf{N}}_{k+1}^{(i)} \geq 0$ *and* $\hat{\mathbf{N}}_k^{(i)} \cdot \hat{\mathbf{N}}(\mathbf{T}^{(i)}) > 0$.

Proof.

Base Case The triangle $\mathbf{T}^{(0)}$ and $\mathbf{q}$ forms a trirectangular tetrahedron. We have $\forall k$, $\hat{\mathbf{N}}_k^{(0)} \cdot \hat{\mathbf{N}}_{k+1}^{(0)} = 0$ and $\hat{\mathbf{N}}_k^{(0)} \cdot \hat{\mathbf{N}}(\mathbf{T}^{(0)}) > 0$.

Induction case We now assume that for any $i \in \{0,\dots,n-1\}$, $\forall k$, $\hat{\mathbf{N}}_k^{(i)} \cdot \hat{\mathbf{N}}_{k+1}^{(i)} \geq 0$ and $\hat{\mathbf{N}}_k^{(i)} \cdot \hat{\mathbf{N}}(\mathbf{T}^{(i)}) > 0$. By the update rule, equation 2.9, we straightforwardly have:

$$\hat{\mathbf{N}}_k^{(i+1)} = \hat{\mathbf{N}}_k^{(i)}, \quad \hat{\mathbf{N}}_{k+1}^{(i+1)} = \hat{\mathbf{N}}_{k+1}^{(i)} + \alpha\hat{\mathbf{N}}_k^{(i)}, \quad \hat{\mathbf{N}}_{k+2}^{(i+1)} = \hat{\mathbf{N}}_{k+2}^{(i)} + \beta\hat{\mathbf{N}}_k^{(i)},$$

and

$$\begin{aligned}
\hat{\mathbf{N}}_k^{(i+1)} \cdot \hat{\mathbf{N}}_{k+1}^{(i+1)} &= \hat{\mathbf{N}}_k^{(i)} \cdot \hat{\mathbf{N}}_{k+1}^{(i)} + \alpha\|\hat{\mathbf{N}}_k^{(i)}\|^2, \\
\hat{\mathbf{N}}_{k+1}^{(i+1)} \cdot \hat{\mathbf{N}}_{k+2}^{(i+1)} &= \hat{\mathbf{N}}_{k+1}^{(i)} \cdot \hat{\mathbf{N}}_{k+2}^{(i)} + \alpha(\hat{\mathbf{N}}_k^{(i)} \cdot \hat{\mathbf{N}}_{k+2}^{(i)}) + \beta(\hat{\mathbf{N}}_{k+1}^{(i)} \cdot \hat{\mathbf{N}}_k^{(i)}) + \alpha\beta\|\hat{\mathbf{N}}_k^{(i)}\|^2, \\
\hat{\mathbf{N}}_{k+2}^{(i+1)} \cdot \hat{\mathbf{N}}_k^{(i+1)} &= \hat{\mathbf{N}}_{k+2}^{(i)} \cdot \hat{\mathbf{N}}_k^{(i)} + \beta\|\hat{\mathbf{N}}_k^{(i)}\|^2.
\end{aligned}$$

Since we have $\forall k$, $\hat{\mathbf{N}}_k^{(i+1)} \cdot \hat{\mathbf{N}}_{k+1}^{(i+1)} \geq \hat{\mathbf{N}}_k^{(i)} \cdot \hat{\mathbf{N}}_{k+1}^{(i)}$ and $\hat{\mathbf{N}}_k^{(i+1)} \cdot \hat{\mathbf{N}}(\mathbf{T}^{(i+1)}) \geq \hat{\mathbf{N}}_k^{(i+1)} \cdot \hat{\mathbf{N}}(\mathbf{T}^{(i+1)})$, the induction hypothesis implies the result. □

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

From now on, we omit once again the exponent (i) for clarity. We go on with this purely geometrical result (see Fig. 4.6):

Lemma 4.9.

Let $\mathbf{d}$ *and* $\mathbf{d}'$ *be two vectors that span a plane of normal* $\mathbf{N} := \mathbf{d}' \times \mathbf{d}$. *Let* $\mathbf{m}$ *be another vector that projects along* $\mathbf{N}$ *into the interior of the convex combination of* $\mathbf{d}$ *and* $\mathbf{d}'$, *i.e.* $(\mathbf{N} \times \mathbf{d}) \cdot \mathbf{m} < 0$ *and* $(\mathbf{N} \times \mathbf{d}') \cdot \mathbf{m} > 0$. *If* $\mathbf{d} \cdot \mathbf{d}' \geq 0$, *then* $\mathbf{d} \cdot \mathbf{m} > 0$ *and* $\mathbf{d}' \cdot \mathbf{m} > 0$.

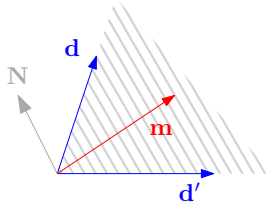


FIGURE 4.6: Illustration of Lemma 4.9. Note that $\mathbf{m}$ does not belong to the span of $\mathbf{d}$ and $\mathbf{d}'$. However, it projects along $\mathbf{N}$ into the interior of the convex combination of $\mathbf{d}$ and $\mathbf{d}'$ (hatched area).

Proof. We first expand $(\mathbf{N} \times \mathbf{d}) \cdot \mathbf{m} < 0$, which is equivalent to $(\mathbf{d} \times \mathbf{m}) \cdot (\mathbf{d} \times \mathbf{d}') > 0$, using the scalar quadruple product rule:

$$\|\mathbf{d}\|^2 \mathbf{d}' \cdot \mathbf{m} - (\mathbf{d} \cdot \mathbf{d}')\mathbf{d} \cdot \mathbf{m} > 0. \tag{4.8}$$

We then similarly expand $(\mathbf{N} \times \mathbf{d}') \cdot \mathbf{m} > 0$, equivalent to $(\mathbf{d}' \times \mathbf{m}) \cdot (\mathbf{d} \times \mathbf{d}') < 0$, as:

$$(\mathbf{d} \cdot \mathbf{d}')\mathbf{d}' \cdot \mathbf{m} - \|\mathbf{d}'\|^2 \mathbf{d} \cdot \mathbf{m} < 0. \tag{4.9}$$

If $\mathbf{d} \cdot \mathbf{d}' = 0$, we can conclude from (4.8) for $\mathbf{d}' \cdot \mathbf{m}$ and from (4.9) for $\mathbf{d} \cdot \mathbf{m}$.

If not, then $\mathbf{d} \cdot \mathbf{d}' > 0$ and we can derive lower and upper bounds for $\mathbf{d}' \cdot \mathbf{m}$, respectively from (4.8) and (4.9):

$$\frac{(\mathbf{d} \cdot \mathbf{d}')}{\|\mathbf{d}\|^2} \mathbf{d} \cdot \mathbf{m} < \mathbf{d}' \cdot \mathbf{m} < \frac{\|\mathbf{d}'\|^2}{(\mathbf{d} \cdot \mathbf{d}')} \mathbf{d} \cdot \mathbf{m}. \tag{4.10}$$

Multiplying both sides by $\|\mathbf{d}\|^2$ and $(\mathbf{d} \cdot \mathbf{d}')$ leads to:

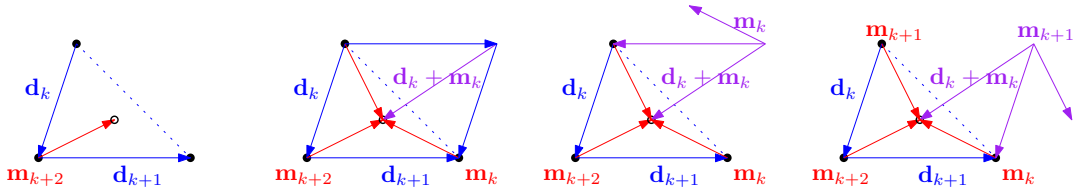
$$\|\mathbf{d} \cdot \mathbf{d}'\|^2 \mathbf{d} \cdot \mathbf{m} < \|\mathbf{d}'\|^2 \|\mathbf{d}\|^2 \mathbf{d} \cdot \mathbf{m} \Leftrightarrow \left(\|\mathbf{d} \cdot \mathbf{d}'\|^2 - \|\mathbf{d}'\|^2 \|\mathbf{d}\|^2 \right) (\mathbf{d} \cdot \mathbf{m}) < 0.$$

Since $\|\mathbf{d} \cdot \mathbf{d}'\|^2 \leq \|\mathbf{d}'\|^2 \|\mathbf{d}\|^2$, we conclude that $\mathbf{d} \cdot \mathbf{m} > 0$. In addition, since $\mathbf{d} \cdot \mathbf{m} > 0$ and $\mathbf{d} \cdot \mathbf{d}' > 0$, it follows from (4.10) that $\mathbf{d}' \cdot \mathbf{m} > 0$. □

We now combine the two preceding lemmas to find angular relations in the tetrahedron formed by the current triangle and $\mathbf{q}$, i.e., involving the vectors $(\mathbf{m}_k)$ and $(\mathbf{d}_k)$. See Fig. 4.7. These results are used in Lemma 4.5 and in sec. 4.2.3.

Lemma 4.10. *For all* k, *if* $\mathbf{d}_k \cdot \mathbf{d}_{k+1} \leq 0$, *then* $\mathbf{d}_{k+1} \cdot \mathbf{m}_{k+2} > 0$ *and* $\mathbf{d}_k \cdot \mathbf{m}_{k+2} < 0$.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés



(A) (B) (C) (D)

FIGURE 4.7: Illustration of Lemma 4.10 in (a), Lemma 4.11 in (b) and Lemma 4.12 in (c) and (d).

Proof. We use Lemma 4.9, with $\mathbf{d}, \mathbf{d}', \mathbf{m}$ respectively set to $(-\mathbf{d}_k)$, $\mathbf{d}_{k+1}$ and $\mathbf{m}_{k+2}$. Note that the normal $\mathbf{d}_k \times \mathbf{d}_{k+1}$ is by definition equal to $\hat{\mathbf{N}}(\mathbf{T})$. Note also that Lemma 4.8 implies

$$\begin{aligned}(\hat{\mathbf{N}}(\mathbf{T}) \times (-\mathbf{d}_k)) \cdot \mathbf{m}_{k+2} =& (\hat{\mathbf{N}}(\mathbf{T}) \times (\mathbf{m}_{k+2} - \mathbf{m}_{k+1})) \cdot \mathbf{m}_{k+2} \\ =& -(\mathbf{m}_{k+1} \times \mathbf{m}_{k+2}) \cdot \hat{\mathbf{N}}(\mathbf{T}) \\ =& -\hat{\mathbf{N}}_k \cdot \hat{\mathbf{N}}(\mathbf{T}) \\ <& 0.\end{aligned} \tag{4.11}$$

And,

$$\begin{aligned}(\hat{\mathbf{N}}(\mathbf{T}) \times (\mathbf{d}_{k+1})) \cdot \mathbf{m}_{k+2} =& (\hat{\mathbf{N}}(\mathbf{T}) \times (\mathbf{m}_{k+2} - \mathbf{m}_k)) \cdot \mathbf{m}_{k+2} \\ =& -(\mathbf{m}_k \times \mathbf{m}_{k+2}) \cdot \hat{\mathbf{N}}(\mathbf{T}) \\ =& \hat{\mathbf{N}}_{k+1} \cdot \hat{\mathbf{N}}(\mathbf{T}) \\ >& 0.\end{aligned} \tag{4.12}$$

These are the projection criterions of Lemma 4.9.

Since we assume in addition that $(-\mathbf{d}_k) \cdot \mathbf{d}_{k+1} \geq 0$, we conclude by Lemma 4.9 that $(-\mathbf{d}_k) \cdot \mathbf{m}_{k+2} > 0$ and $\mathbf{d}_{k+1} \cdot \mathbf{m}_{k+2} > 0$. □

Likewise,

Lemma 4.11. *For all k, if $\mathbf{d}_k \cdot \mathbf{d}_{k+1} \leq 0$, then $\mathbf{d}_k \cdot (\mathbf{d}_k + \mathbf{m}_k) > 0$ and $\mathbf{d}_{k+1} \cdot (\mathbf{d}_k + \mathbf{m}_k) < 0$.*

Proof. We use Lemma 4.9, with $\mathbf{d}, \mathbf{d}', \mathbf{m}$ respectively set to $(-\mathbf{d}_k)$, $\mathbf{d}_{k+1}$ and $-(\mathbf{d}_k + \mathbf{m}_k)$. Note that the normal is equal to $\hat{\mathbf{N}}(\mathbf{T})$ and the projection criterion is implied

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

by Lemma 4.8

$$\begin{aligned}(\hat{\mathbf{N}}(\mathbf{T}) \times (-\mathbf{d}_k)) \cdot (\mathbf{d}_k + \mathbf{m}_k) =& ((-\mathbf{d}_k) \times (\mathbf{d}_k + \mathbf{m}_k)) \cdot \hat{\mathbf{N}}(\mathbf{T}) \\ =& ((-\mathbf{d}_k) \times \mathbf{m}_k) \cdot \hat{\mathbf{N}}(\mathbf{T}) \\ =& ((-\mathbf{m}_{k+1} + \mathbf{m}_{k+2}) \times \mathbf{m}_k) \cdot \hat{\mathbf{N}}(\mathbf{T}) \\ =& (\hat{\mathbf{N}}_{k+2} + \hat{\mathbf{N}}_{k+1}) \cdot \hat{\mathbf{N}}(\mathbf{T}) \\ >& 0.\end{aligned} \tag{4.13}$$

And,

$$\begin{aligned}(\hat{\mathbf{N}}(\mathbf{T}) \times \mathbf{d}_{k+1}) \cdot (\mathbf{d}_k + \mathbf{m}_k) =& (\hat{\mathbf{N}}(\mathbf{T}) \times \mathbf{d}_{k+1}) \cdot (-\mathbf{d}_{k+1} + \mathbf{m}_{k+1}) \\ =& (\hat{\mathbf{N}}(\mathbf{T}) \times \mathbf{d}_{k+1}) \cdot (\mathbf{m}_{k+1}) \\ =& (\mathbf{d}_{k+1} \times (\mathbf{m}_{k+1})) \cdot \hat{\mathbf{N}}(\mathbf{T}) \\ =& (((\mathbf{m}_{k+2} - \mathbf{m}_k) \times (\mathbf{m}_{k+1})) \cdot \hat{\mathbf{N}}(\mathbf{T}) \\ =& (-\hat{\mathbf{N}}_{k+2} - \hat{\mathbf{N}}_k)) \cdot \hat{\mathbf{N}}(\mathbf{T}) \\ <& 0.\end{aligned} \tag{4.14}$$

From Lemma 4.9, we thus have $\mathbf{d}_k \cdot (\mathbf{d}_k + \mathbf{m}_k) > 0$ and $\mathbf{d}_{k+1} \cdot (\mathbf{d}_k + \mathbf{m}_k) > 0$. □

Finally,

Lemma 4.12. *For all k, if $\mathbf{m}_k \cdot (\mathbf{d}_k + \mathbf{m}_k) \geq 0$, then $\mathbf{d}_{k+1} \cdot \mathbf{m}_k < 0$ and $\mathbf{d}_{k+1} \cdot (\mathbf{d}_k + \mathbf{m}_k) < 0$. Similarly, if $\mathbf{m}_{k+1} \cdot (\mathbf{d}_k + \mathbf{m}_k) \geq 0$, then $\mathbf{d}_k \cdot \mathbf{m}_{k+1} > 0$ and $\mathbf{d}_k \cdot (\mathbf{d}_k + \mathbf{m}_k) > 0$.*

Proof. We focus on the first part, because the proof of the second part is quite similar.

We use Lemma 4.9, with $\mathbf{d}, \mathbf{d}', \mathbf{m}$ respectively set to $(\mathbf{d}_k + \mathbf{m}_k)$, $\mathbf{m}_k$ and $(-\mathbf{d}_{k+1})$. Note that the normal is equal to $\mathbf{m}_k \times (\mathbf{d}_k + \mathbf{m}_k) = \hat{\mathbf{N}}_{k+1} + \hat{\mathbf{N}}_{k+2}$ and the projection criterion is implied by Lemma 4.8

$$\begin{aligned}((\hat{\mathbf{N}}_{k+1} + \hat{\mathbf{N}}_{k+2}) \times (\mathbf{d}_k + \mathbf{m}_k)) \cdot (-\mathbf{d}_{k+1}) =& ((\hat{\mathbf{N}}_{k+1} + \hat{\mathbf{N}}_{k+2}) \times (\mathbf{m}_{k+1} - \mathbf{d}_{k+1})) \cdot (-\mathbf{d}_{k+1}) \\ =& ((\hat{\mathbf{N}}_{k+1} + \hat{\mathbf{N}}_{k+2}) \times (\mathbf{m}_{k+1})) \cdot (-\mathbf{d}_{k+1}) \\ =& ((\mathbf{m}_{k+1}) \times (-\mathbf{d}_{k+1})) \cdot (\hat{\mathbf{N}}_{k+1} + \hat{\mathbf{N}}_{k+2}) \\ =& ((\mathbf{m}_{k+1}) \times (-\mathbf{m}_{k+2} + \mathbf{m}_k)) \cdot (\hat{\mathbf{N}}_{k+1} + \hat{\mathbf{N}}_{k+2}) \\ =& (-\hat{\mathbf{N}}_k - \hat{\mathbf{N}}_{k+2}) \cdot (\hat{\mathbf{N}}_{k+1} + \hat{\mathbf{N}}_{k+2}) \\ =& -(\hat{\mathbf{N}}_k \cdot \hat{\mathbf{N}}_{k+1} + \hat{\mathbf{N}}_k \cdot \hat{\mathbf{N}}_{k+2} + \hat{\mathbf{N}}_{k+2} \cdot \hat{\mathbf{N}}_{k+1} + \|\hat{\mathbf{N}}_{k+2}\|^2) \\ <& 0.\end{aligned} \tag{4.15}$$

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

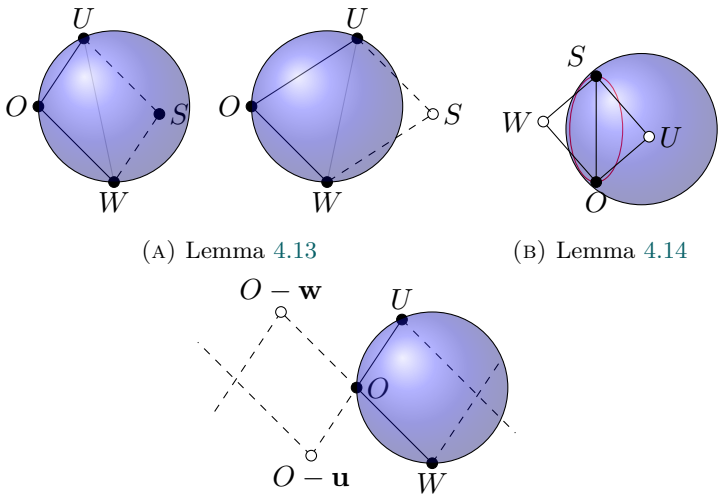
And,

$$\begin{aligned}((\hat{\mathbf{N}}_{k+1}+\hat{\mathbf{N}}_{k+2})\times\mathbf{m}_k)\cdot(-\mathbf{d}_{k+1}) =&(\mathbf{m}_k\times(-\mathbf{d}_{k+1}))\cdot(\hat{\mathbf{N}}_{k+1}+\hat{\mathbf{N}}_{k+2})\\ =&(\mathbf{m}_k\times(-\mathbf{m}_{k+2}+\mathbf{m}_k))\cdot(\hat{\mathbf{N}}_{k+1}+\hat{\mathbf{N}}_{k+2})\\ =&\hat{\mathbf{N}}_{k+1}\cdot(\hat{\mathbf{N}}_{k+1}+\hat{\mathbf{N}}_{k+2})\\ =&(\|\hat{\mathbf{N}}_{k+1}\|^2+\hat{\mathbf{N}}_{k+1}\cdot\hat{\mathbf{N}}_{k+2})\\ >&0.\end{aligned} \tag{4.16}$$

From Lemma 4.9, we thus have $\mathbf{d}_{k+1}\cdot\mathbf{m}_k<0$ and $\mathbf{d}_{k+1}\cdot(\mathbf{d}_k+\mathbf{m}_k)<0$, which concludes. □

4.2.2 Circumsphere-based results

In this section, we show several general and purely geometrical circumsphere-based results. They are the cornerstone of many proof in this thesis. Lemma 4.13 is the most often used, notably in some of the key results, such as Lemma 4.5 and Lemma 4.7, as well as in Corollary 5.1. Lemma 4.14 is invoked in Lemma 4.5, whereas Lemma 3.2 and Lemma 4.15 are crucial in sec. 4.2.3.



(A) Lemma 4.13 (B) Lemma 4.14

(C) Lemma 4.15

FIGURE 4.8: Illustrations for circumsphere-based lemmas of sec. 4.2.2. A point is depicted as a black disk if it is inside the ball of interest, and as a hollow disk if it is not in the closed ball.

Lemma 4.13. *Let* $\mathbf{u},\mathbf{w}$ *be two non-zero vectors of* $\mathbb{R}^3$*. Let* B *be a closed ball whose border passes through the origin* O *as well as through* $U:=O+\mathbf{u}$ *and* $W:=O+\mathbf{w}$*. The point* $S:=O+\mathbf{u}+\mathbf{w}$ *belongs to the interior of the ball* B *if and only if* $\mathbf{u}\cdot\mathbf{w}<0$*.*

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Proof. Consider a ball B with centre c and radius r. First $O \in \partial B$ is equivalent to $r^2 = c^2$. Using this relation it follows:

$$U \in \partial B \Leftrightarrow (O + \mathbf{u} - c)^2 = c^2 \Leftrightarrow 2c \cdot \mathbf{u} = \mathbf{u}^2, \tag{4.17}$$

$$W \in \partial B \Leftrightarrow (O + \mathbf{w} - c)^2 = c^2 \Leftrightarrow 2c \cdot \mathbf{w} = \mathbf{w}^2, \tag{4.18}$$

$$S \in \mathring{B} \Leftrightarrow (O + \mathbf{u} + \mathbf{w} - c)^2 < c^2 \Leftrightarrow (\mathbf{u} + \mathbf{w})^2 < 2(\mathbf{u} + \mathbf{w}) \cdot c. \tag{4.19}$$

Developing $(\mathbf{u} + \mathbf{w})^2$ in (4.19), we get $\mathbf{u}^2 + \mathbf{w}^2 + 2\mathbf{u} \cdot \mathbf{w} < 2(\mathbf{u} + \mathbf{w}) \cdot c$, which is equivalent to $\mathbf{u}^2 + \mathbf{w}^2 + 2\mathbf{u} \cdot \mathbf{w} < \mathbf{u}^2 + \mathbf{w}^2$ by (4.17) and (4.18). Subtracting $\mathbf{u}^2 + \mathbf{w}^2$ from both sides gives the equivalent formulation $\mathbf{u} \cdot \mathbf{w} < 0$. □

Lemma 4.14. *Let $\mathbf{u}, \mathbf{w}$ be two non-zero vectors of $\mathbb{R}^3$. Let B be a closed ball whose border passes through the origin O and the point $S := O + \mathbf{u} + \mathbf{w}$. If $U := O + \mathbf{u}$ and $W := O + \mathbf{w}$ do not lie in the ball B, then $\mathbf{u} \cdot \mathbf{w} < 0$.*

Proof. This lemma is the reverse version of Lemma 3.1. □

Lemma 4.15. *Let $\mathbf{u}, \mathbf{w}$ be two non-zero vectors of $\mathbb{R}^3$. Let B be a closed ball whose border passes through the origin O and the two points $U := O + \mathbf{u}$ and $W := O + \mathbf{w}$. No point of the set $\Lambda := \{O - a\mathbf{u} - b\mathbf{w} \mid (a, b) \in \mathbb{N}^2\}$ lies in the interior $\mathring{B}$ of the ball B.*

Proof. Consider a ball B with centre c and radius r, and recalling that $O \in \partial B$ is equivalent to $r^2 = c^2$, we get

$$U \in \partial B \Leftrightarrow (O + \mathbf{u} - c)^2 = c^2 \Leftrightarrow 2c \cdot \mathbf{u} = \mathbf{u}^2, \tag{4.20}$$

$$W \in \partial B \Leftrightarrow (O + \mathbf{w} - c)^2 = c^2 \Leftrightarrow 2c \cdot \mathbf{w} = \mathbf{w}^2. \tag{4.21}$$

We then compute for arbitrary non-negative integers a and b:

$$\begin{aligned}(O - a\mathbf{u} - b\mathbf{w} - c)^2 &= c^2 + (a\mathbf{u} + b\mathbf{w})^2 + 2c \cdot (a\mathbf{u} + b\mathbf{w}) \\ &= c^2 + (a\mathbf{u} + b\mathbf{w})^2 + a\mathbf{u}^2 + b\mathbf{w}^2 \qquad \text{(using (4.20) and (4.21))} \\ &\geq r^2. \qquad \text{(since } a \geq 0,\ b \geq 0 \text{ and } c^2 = r^2\text{)}\end{aligned}$$

We conclude since $(O - a\mathbf{u} - b\mathbf{w} - c)^2 \geq r^2$ is equivalent to $O - a\mathbf{u} - b\mathbf{w} \notin \mathring{B}$. □

4.2.3 Proximity results

In this section, we demonstrate some technical lemmas that give the order relations induced by the spheres circumscribing the current triangle. They are used in the proofs of Lemma 4.5 and Lemma 4.7 to establish the Delaunay property.

Most proofs in this section uses (3.17).

Lemma 4.16. *Let $\mathbf{u} := -\mathbf{m}_0 + \mathbf{m}_1 + \mathbf{m}_2$. If $\mathbf{d}_1 \cdot \mathbf{u} \geq 0$ (resp. $(-\mathbf{d}_2) \cdot \mathbf{u} \geq 0$), then $\delta^0_{\mathbf{T}}(\mathbf{m}_1, a\mathbf{u}) \geq 0$ (resp. $\delta^0_{\mathbf{T}}(\mathbf{m}_2, a\mathbf{u}) \geq 0$) for all $a \in \mathbb{N}$.*

Proof. The lemma is trivially true for $a = 0$ and we can safely assume that $a \geq 1$.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Base case Using Lemma 4.13 with the vectors $\mathbf{d}_1$, $\mathbf{u}$ and the origin set to $\mathbf{v}_2$, $\mathbf{d}_1 \cdot \mathbf{u} \geq 0$ implies that the ball whose border passes through $\mathbf{T}$ and $\mathbf{v}_2 + \mathbf{u} = \mathbf{v}_0 + \mathbf{m}_1$ does not include $\mathbf{v}_2 + \mathbf{d}_1 + \mathbf{u} = \mathbf{v}_0 + \mathbf{u}$ in its interior. That means that $\delta^0_{\mathbf{T}}(\mathbf{m}_1, \mathbf{u}) \geq 0$ and we can similarly show that $\delta^0_{\mathbf{T}}(\mathbf{m}_2, \mathbf{u}) \geq 0$ if $(-\mathbf{d}_2) \cdot \mathbf{u} \geq 0$.

Induction step Let $\mathbf{m}$ be either $\mathbf{m}_1$ or $\mathbf{m}_2$. We now assume that for some $a \in \mathbb{N}$, $\delta^0_{\mathbf{T}}(\mathbf{m}, a\mathbf{u}) \geq 0$ and we want to show that $\delta^0_{\mathbf{T}}(\mathbf{m}, (a+1)\mathbf{u}) \geq 0$.

By (3.17), we have

$$\begin{aligned}\delta^0_{\mathbf{T}}(\mathbf{m}, (a+1)\mathbf{u}) = \delta^0_{\mathbf{T}}(\mathbf{m}, a\mathbf{u}) + \delta^0_{\mathbf{T}}(\mathbf{m}, \mathbf{u}) \\ + 2a(\mathbf{u} \cdot \mathbf{u}) \det[\mathbf{d}_2, -\mathbf{d}_1, \mathbf{m}].\end{aligned}$$

Since $\det[\mathbf{d}_2, -\mathbf{d}_1, \mathbf{m}] = \det[\mathbf{m}_0, \mathbf{m}_1, \mathbf{m}_2]$, which is equal to 1 by Theorem 2.2, the whole sum is strictly positive due to the induction hypothesis and the base case. □

Lemma 4.17, Lemma 4.18, Lemma 4.19 are respectively used in the cases (i), (ii) and (iii) of the proof of Lemma 4.5.

Lemma 4.17. *Let* $\mathbf{u} := -\mathbf{m}_0 + \mathbf{m}_1 + \mathbf{m}_2$ *and* $\mathbf{w} := a(\mathbf{d}_1) + b(-\mathbf{d}_2) + c(\mathbf{u})$*, with* $a, b, c \geq 0$*. If* $\mathbf{d}_1 \cdot \mathbf{u} \geq 0$ *and* $(-\mathbf{d}_2) \cdot \mathbf{u} \geq 0$*, then* $\delta^0_{\mathbf{T}}(\mathbf{m}_2, \mathbf{w}) \geq 0$.

Proof. By (3.17), we have

$$\begin{aligned}\delta^0_{\mathbf{T}}(\mathbf{m}_2, \mathbf{w}) = \delta^0_{\mathbf{T}}(\mathbf{m}_2, a\mathbf{d}_1 + b(-\mathbf{d}_2)) + \delta^0_{\mathbf{T}}(\mathbf{m}_2, c\mathbf{u}) \\ + 2\Big((a\mathbf{d}_1 + b(-\mathbf{d}_2)) \cdot c\mathbf{u}\Big) \det[\mathbf{d}_2, -\mathbf{d}_1, \mathbf{m}_2].\end{aligned}$$

We show below that the three terms are positive, so is the whole sum.

- For the first term, we apply Lemma 4.15 with the vectors $-\mathbf{d}_1$, $\mathbf{d}_2$ and the origin set to $\mathbf{v}_0$ to deduce that the point $\mathbf{v}_0 + a\mathbf{d}_1 + b(-\mathbf{d}_2)$ is not in the interior of the ball passing through $\mathbf{T}$ and $\mathbf{v}_0 + \mathbf{m}_2$. Thus, $\delta^0_{\mathbf{T}}(\mathbf{m}_2, a\mathbf{d}_1 + b(-\mathbf{d}_2)) \geq 0$.
- Since $(-\mathbf{d}_2) \cdot \mathbf{u} \geq 0$, $\delta^0_{\mathbf{T}}(\mathbf{m}_2, c\mathbf{u}) \geq 0$ by Lemma 4.16.
- Finally, $(a\mathbf{d}_1 + b(-\mathbf{d}_2)) \cdot c\mathbf{u} \geq 0$ because $a, b, c, \mathbf{d}_1 \cdot \mathbf{u}$ and $(-\mathbf{d}_2) \cdot \mathbf{u}$ are assumed to be positive and, using Theorem 2.2, one can easily check that $\det[\mathbf{d}_2, -\mathbf{d}_1, \mathbf{m}_2] = \det[\mathbf{m}_0, \mathbf{m}_1, \mathbf{m}_2] = 1$.

□

Lemma 4.18. *Let* $\mathbf{u} := -\mathbf{m}_0 + \mathbf{m}_1 + \mathbf{m}_2$ *and* $\mathbf{w} := a(\mathbf{d}_1) + b(\mathbf{m}_2) + c(\mathbf{u})$*, with* $a, b, c \geq 0$*. If* $\mathbf{m}_2 \cdot \mathbf{u} \geq 0, \mathbf{d}_1 \cdot \mathbf{u} \geq 0, (-\mathbf{d}_2) \cdot \mathbf{u} \geq 0$ *and* $\mathbf{d}_1 \cdot \mathbf{m}_2 \geq 0$*, then* $\delta^0_{\mathbf{T}}(\mathbf{m}_2, \mathbf{w}) \geq 0$.

Proof. By (3.17), we have

$$\begin{aligned}\delta^0_{\mathbf{T}}(\mathbf{m}_2, \mathbf{w}) = \delta^0_{\mathbf{T}}(\mathbf{m}_2, a\mathbf{d}_1 + b\mathbf{m}_2) + \delta^0_{\mathbf{T}}(\mathbf{m}_2, c\mathbf{u}) \\ + 2\Big((a\mathbf{d}_1 + b\mathbf{m}_2) \cdot c\mathbf{u}\Big) \det[\mathbf{d}_2, -\mathbf{d}_1, \mathbf{m}_2].\end{aligned}$$

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

One can easily check that $\det[\mathbf{d}_2, -\mathbf{d}_1, \mathbf{m}_2] = \det[\mathbf{m}_0, \mathbf{m}_1, \mathbf{m}_2]$, which is equal to 1 by Theorem 2.2.

In addition, we use (3.17) again to decompose the first term and finally get

$$\begin{aligned}\delta^0_{\mathbf{T}}(\mathbf{m}_2, \mathbf{w}) = \delta^0_{\mathbf{T}}(\mathbf{m}_2, a\mathbf{d}_1) + \delta^0_{\mathbf{T}}(\mathbf{m}_2, b\mathbf{m}_2) + \delta^0_{\mathbf{T}}(\mathbf{m}_2, c\mathbf{u}) \\ + 2ab(\mathbf{d}_1 \cdot \mathbf{m}_2) + 2(a\mathbf{d}_1 + b\mathbf{m}_2) \cdot c\mathbf{u}.\end{aligned}$$

We can now prove that each term of the sum is positive:

- For the first two terms, we consider the ball whose border passes through $\mathbf{T}$ and $\mathbf{v}_0+\mathbf{m}_2$. If $a = 0$ (resp. $b \in \{0, 1\}$), the point $\mathbf{v}_0+a\mathbf{d}_1$ (resp. $\mathbf{v}_0+b\mathbf{m}_2$) trivially belongs to the boundary of the ball, which implies a null term. If $a \geq 1$ (resp. $b \geq 2$), we consider the ray from $\mathbf{v}_2$ in direction $\mathbf{d}_1$ (resp. from $\mathbf{v}_0$ in direction $\mathbf{m}_2$) and we use Lemma 3.2 to show that the point $\mathbf{v}_0 + a\mathbf{d}_1$ (resp. $\mathbf{v}_0 + b\mathbf{m}_2$) does not belong to the interior of the ball, which means that $\delta^0_{\mathbf{T}}(\mathbf{m}_2, a\mathbf{d}_1) \geq 0$ (resp. $\delta^0_{\mathbf{T}}(\mathbf{m}_2, b\mathbf{m}_2) \geq 0$).
- Since $(-\mathbf{d}_2) \cdot \mathbf{u} \geq 0$, $\delta^0_{\mathbf{T}}(\mathbf{m}_2, c\mathbf{u}) \geq 0$ by Lemma 4.16.
- All scalar products of the last two terms are positive or null due to the hypotheses.

□

Lemma 4.19. *Let* $\mathbf{u} := -\mathbf{m}_0+\mathbf{m}_1+\mathbf{m}_2$, $\mathbf{w} := a(\mathbf{m}_1)+b(\mathbf{m}_2)+c(\mathbf{u})$, *with* $a, b, c \geq 0$. *Let* Λ *be the set* $\{\alpha\mathbf{m}_1 + \beta\mathbf{m}_2 \mid \alpha, \beta \in \mathbb{N} \setminus (0, 0), \alpha \leq a, \beta \leq b\}$ *and* $\mathbf{w}' \in \Lambda$ *be such that* $\forall \mathbf{w}'' \in \Lambda, \delta^0_{\mathbf{T}}(\mathbf{w}', \mathbf{w}'') \geq 0$. *If* $\mathbf{m}_1 \cdot \mathbf{u} \geq 0$ *and* $\mathbf{m}_2 \cdot \mathbf{u} \geq 0$, *then* $\delta^0_{\mathbf{T}}(\mathbf{w}', \mathbf{w}) \geq 0$.

Proof. By (3.17), we have

$$\begin{aligned}\delta^0_{\mathbf{T}}(\mathbf{w}', \mathbf{w}) = \delta^0_{\mathbf{T}}(\mathbf{w}', (a\mathbf{m}_1 + b\mathbf{m}_2)) + \delta^0_{\mathbf{T}}(\mathbf{w}', c\mathbf{u}) \\ + 2\Big((a\mathbf{m}_1 + b\mathbf{m}_2)) \cdot c\mathbf{u}\Big) \det[\mathbf{d}_2, -\mathbf{d}_1, \mathbf{w}'].\end{aligned}$$

- The first term $\delta^0_{\mathbf{T}}(\mathbf{w}', (a\mathbf{m}_1 + b\mathbf{m}_2))$ is positive by definition of $\mathbf{w}'$.
- $2\Big((a\mathbf{m}_1+b\mathbf{m}_2)) \cdot c\mathbf{u}\Big)$ is positive because we assume that $a, b, c, \mathbf{m}_1 \cdot \mathbf{u}$ and $\mathbf{m}_2 \cdot \mathbf{u}$ are positive. Moreover, setting $\mathbf{w}' := a'\mathbf{m}_1 + b'\mathbf{m}_2$ and using Theorem 2.2, one can easily check that

  $$\det[\mathbf{d}_2, -\mathbf{d}_1, \mathbf{w}'] = (a' + b') \det[\mathbf{m}_0, \mathbf{m}_1, \mathbf{m}_2] = (a' + b') \geq 1.$$

  As a consequence, the third term of the sum is positive.
- It remains to show that the second term is also positive.

  By Lemma 4.12, $\mathbf{m}_2 \cdot \mathbf{u} \geq 0 \Rightarrow \mathbf{d}_1 \cdot \mathbf{u} > 0$. From the last inequality, we have by Lemma 4.16, $\delta^0_{\mathbf{T}}(\mathbf{m}_1, c\mathbf{u}) \geq 0$, which means that $\mathbf{v}_0 + \mathbf{m}_1 \leq_{\mathbf{T}} \mathbf{v}_0 + c\mathbf{u}$. However, since $\mathbf{v}_0 + \mathbf{w}' \leq_{\mathbf{T}} \mathbf{v}_0 + \mathbf{m}_1$ by definition of $\mathbf{w}'$, we have by transitivity $\mathbf{v}_0 + \mathbf{w}' \leq_{\mathbf{T}} \mathbf{v}_0 + c\mathbf{u}$, i.e., $\delta^0_{\mathbf{T}}(\mathbf{w}', c\mathbf{u}) \geq 0$.

□

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Lemma 4.20 is used in the the proof of Lemma 4.7.

Lemma 4.20. *Let* $\mathbf{w} := a(-\mathbf{d}_0) + b(\mathbf{d}_1) + c(\mathbf{m}_2)$*, with* $a, b, c \geq 0$*. If* $(-\mathbf{d}_0) \cdot \mathbf{d}_1 \geq 0$*, then* $\delta^0_{\mathbf{T}}(\mathbf{m}_2, \mathbf{w}) \geq 0$*.*

Proof. By (3.17), we have

$$\begin{aligned}\delta^0_{\mathbf{T}}(\mathbf{m}_2, \mathbf{w}) = \delta^0_{\mathbf{T}}(\mathbf{m}_2, a(-\mathbf{d}_0) + b(\mathbf{d}_1)) + \delta^0_{\mathbf{T}}(\mathbf{m}_2, c\mathbf{m}_2) \\ + 2\Big((a(-\mathbf{d}_0) + b\mathbf{d}_1) \cdot c\mathbf{m}_2\Big) \det[\mathbf{d}_2, -\mathbf{d}_1, \mathbf{m}_2].\end{aligned}$$

One can easily check that $\det[\mathbf{d}_2, -\mathbf{d}_1, \mathbf{m}_2] = \det[\mathbf{m}_0, \mathbf{m}_1, \mathbf{m}_2]$, which is equal to 1 by Theorem 2.2.

In addition, we use (3.17) again to decompose the first term and finally get

$$\begin{aligned}\delta^0_{\mathbf{T}}(\mathbf{m}_2, \mathbf{w}) = \delta^0_{\mathbf{T}}(\mathbf{m}_2, a(-\mathbf{d}_0)) + \delta^0_{\mathbf{T}}(\mathbf{m}_2, b(\mathbf{d}_1)) + \delta^0_{\mathbf{T}}(\mathbf{m}_2, c\mathbf{m}_2) \\ + 2ab((-\mathbf{d}_0) \cdot \mathbf{d}_1) + 2(a(-\mathbf{d}_0) + b\mathbf{d}_1) \cdot c\mathbf{m}_2,\end{aligned}$$

We can now prove that each term of the sum is positive:

- For the first term, we consider the ball whose border passes through $\mathbf{T}$ and $\mathbf{v}_0 + \mathbf{u}$. If $a = 0$, the point $\mathbf{v}_0 + a(-\mathbf{d}_0) = \mathbf{v}_0$ trivially belongs to the boundary of the ball, which implies a null term. If $a = 1$, we apply Lemma 4.13 with the vectors $(-\mathbf{d}_0)$, $\mathbf{d}_1$ and the origin set to $\mathbf{v}_2$ to deduce from $(-\mathbf{d}_0) \cdot \mathbf{d}_1 \geq 0$ that the point $\mathbf{v}_2 + \mathbf{d}_1 + a(-\mathbf{d}_0) = \mathbf{v}_0 + a(-\mathbf{d}_0)$ does not belong to the interior of the ball, which means that $\delta^0_{\mathbf{T}}(\mathbf{m}_2, a(-\mathbf{d}_0)) \geq 0$. For $a \geq 2$, we consider the ray from $\mathbf{v}_0$ in direction $-\mathbf{d}_0$ to show that we have the same result in that case too.
- The two next terms are also positive or null and we can verify this using Lemma 3.2 as in the proof of Lemma 4.18 (first item).
- The fourth term is positive because a, b and $((-\mathbf{d}_0) \cdot \mathbf{d}_1)$ are assumed to be positive.
- For the sign of the last term, it is enough to note that $((-\mathbf{d}_0) \cdot \mathbf{d}_1) \geq 0$ also implies $\mathbf{m}_2 \cdot (-\mathbf{d}_0) > 0$ and $\mathbf{m}_2 \cdot \mathbf{d}_1 > 0$ by Lemma 4.10. As a consequence, the term $(a(-\mathbf{d}_0) + b\mathbf{d}_1) \cdot c\mathbf{m}_2$ develops into two positive scalar products and is therefore positive.

□

4.3 Increasing radius of ball

The Delaunay property allows us to prove this geometrical property, that is only true for the L-algorithm.

Lemma 4.21. *The sequence of radii of* $\{\mathcal{B}^{(i)}\}_{0 \leq i \leq n}$ *is non-decreasing.*

Proof. We conveniently denote by $\mathcal{H}^{(n)}$ the upper leaning plane $\{\mathbf{x} \in \mathbb{R}^3 \mid \mathbf{x} \cdot \mathbf{N} = \|\mathbf{N}\|_1 - 1\}$, and by $\mathcal{H}^{(n)}_+$ the open half-space lying above it, i.e., $\{\mathbf{x} \in \mathbb{R}^3 \mid \mathbf{x} \cdot \mathbf{N} >$

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

$\|\mathbf{N}\|_1 - 1\}$. We show below how the claim of the lemma is a by-product of the following property:

Property 4.2. *For all $i \in \{1, \dots, n\}$, the center of $\mathcal{B}^{(i)}$, denoted by $\mathcal{C}^{(i)}$, belongs to both $\mathcal{H}_+^{(i)}$ and $\mathcal{H}_+^{(n)}$.*

Note that $\mathcal{C}^{(0)}$ is neither in $\mathcal{H}_+^{(0)}$ nor in $\mathcal{H}_+^{(n)}$. The base case of the induction is $\mathcal{C}^{(1)}$.

Base case: If we translate everything so that $\mathbf{q} = (0,0,0)$, $\mathbf{v}_0^{(0)} = (-1,0,0)$, $\mathbf{v}_1^{(0)} = (0,-1,0)$ and $\mathbf{v}_2^{(0)} = (0,0,-1)$, then $\mathcal{C}^{(1)} = (0.5, 0.5, 0.5)$, no matter which point is chosen for $\mathbf{T}^{(1)}$ in the H-neighborhood $\mathcal{N}_{S_H}^{(0)}$. [1] We can indeed check that every point of $\mathbf{T}^{(0)}$ and $\mathcal{N}_{S_H}^{(0)}$ (and therefore $\mathbf{T}^{(1)}$) verify the following equation:

$$(x - 0.5)^2 + (y - 0.5)^2 + (z - 0.5)^2 = 2.75 \tag{4.22}$$

Moreover, if we take the case where $\mathbf{v}_2^{(1)} = (1, 0, -1)$ as example, and compute the following determinant:

$$\begin{vmatrix} 1 & 1 & 1 & 1 \\ -1 & 0 & 1 & 0.5 \\ 0 & -1 & 0 & 0.5 \\ 0 & 0 & -1 & 0.5 \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 2 & 1.5 \\ 0 & -1 & 0 & 0.5 \\ 0 & 0 & -1 & 0.5 \end{vmatrix} = \begin{vmatrix} 1 & 2 & 1.5 \\ 0 & 2 & 2 \\ 0 & -1 & 0.5 \end{vmatrix} = 1 + 2 = 3 > 0, \tag{4.23}$$

we found that $\mathcal{C}^{(1)} \in \mathcal{H}_+^{(1)}$. The other cases, when a different point of $\mathcal{N}_{S_H}^{(1)}$ is picked, are similar by symmetry or rotation. In addition, we also have trivially $\mathcal{C}^{(1)} \in \mathcal{H}_+^{(n)}$.

For the following, we use Fig. 4.9 as visual aid.

Induction step: Suppose that $\mathcal{C}^{(i)}$ belongs to both $\mathcal{H}_+^{(i)}$ and $\mathcal{H}_+^{(n)}$. We want to show that $\mathcal{C}^{(i+1)}$ belongs to both $\mathcal{H}_+^{(i+1)}$ and $\mathcal{H}_+^{(n)}$.

Let $\mathcal{H}^{(i)}$ be the plane that coincide with $\mathbf{T}^{(i)}$ and $c^{(i)}$ be the circumcenter of $\mathbf{T}^{(i)}$. Let $\mathcal{L}^{(i)}$ be the straight line that is perpendicular to $\mathcal{H}^{(i)}$ and passes through $c^{(i)}$.

We say that a point x is higher that a point y on $\mathcal{L}^{(i)}$ if and only if $\overrightarrow{c^{(i)}x} \cdot \hat{\mathbf{N}}(\mathbf{T}^{(i)}) > \overrightarrow{c^{(i)}y} \cdot \hat{\mathbf{N}}(\mathbf{T}^{(i)})$, where $\hat{\mathbf{N}}(\mathbf{T}^{(i)})$ is the normal of triangle $\mathbf{T}^{(i)}$.

If the two balls $\mathcal{B}^{(i)}$ and $\mathcal{B}^{(i+1)}$ are identical, there is nothing to say. Otherwise, by the Delaunay property, we know that one of the vertices of $\mathbf{T}^{(i+1)}$ belongs to $\mathcal{H}_+^{(i)}$, but does not belong to $\mathcal{B}^{(i)}$. This implies that $\mathcal{C}^{(i+1)}$ is higher than $\mathcal{C}^{(i)}$ on $\mathcal{L}^{(i)}$. Since $\mathcal{C}^{(i)}$ belongs to both $\mathcal{H}_+^{(i)}$ and $\mathcal{H}_+^{(n)}$ by hypothesis, $\mathcal{C}^{(i+1)}$ also belongs to both $\mathcal{H}_+^{(i)}$ and $\mathcal{H}_+^{(n)}$. It remains to show that $\mathcal{C}^{(i+1)} \in \mathcal{H}_+^{(i+1)}$.

Let $p^{(i)}$ be the intersection between $\mathcal{L}^{(i)}$ and $\mathcal{H}^{(i+1)}$. Since $p^{(i)}$ clearly does not belong to $\mathcal{H}_+^{(n)}$, $\mathcal{C}^{(i+1)}$ is further than $p^{(i)}$ on $\mathcal{L}^{(i)}$ and as a consequence, $\mathcal{C}^{(i+1)} \in \mathcal{H}_+^{(i+1)}$.

The position of $\mathcal{C}^{(i)}$ and $\mathcal{C}^{(i+1)}$ on the line $\mathcal{L}^{(i)}$ clearly implies that the radius of $\mathcal{C}^{(i+1)}$ is larger than the radius of $\mathcal{C}^{(i)}$. Therefore, we can conclude. □

[1] The angle between the rays emitted from the vertices of $\mathbf{T}^{(0)}$ is always right. No matter which neighborhood we adopt, $\mathcal{B}^{(1)}$ will surely pass one of the point of the H-neighborhood.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

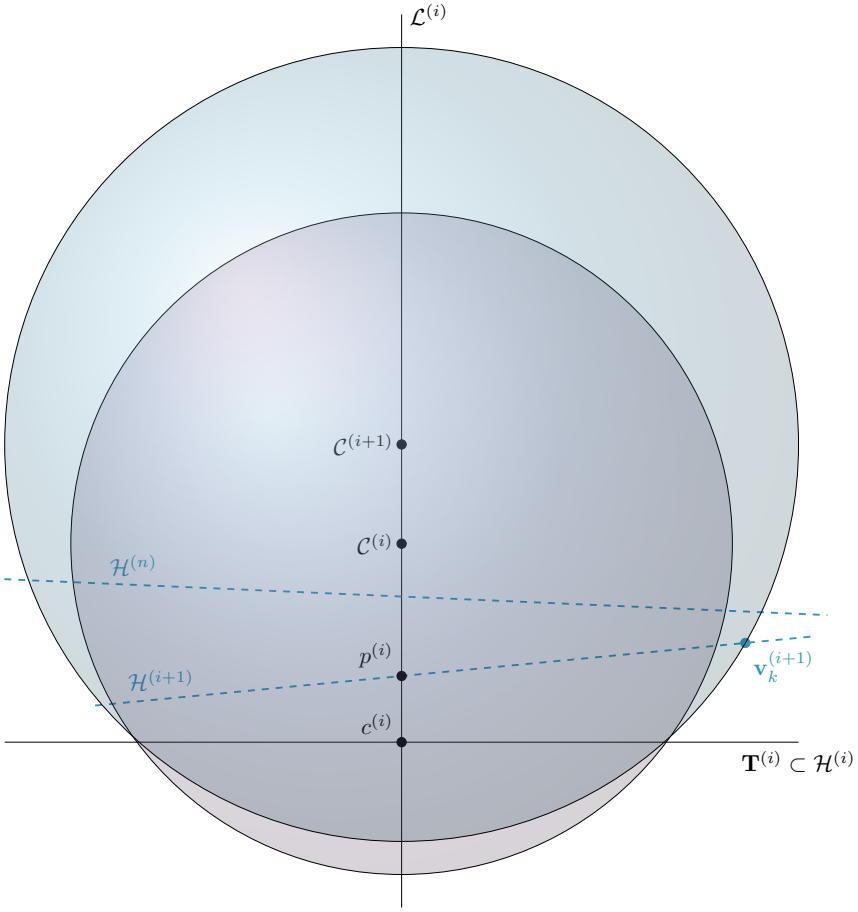


FIGURE 4.9: Illustration for the proof of lemma 4.21.

4.4 Conclusion

In this chapter, we introduce the Delaunay property as an invariant of the L-algorithm. We discussed that proving this property involves a lengthy process with many technical details, particularly concerning the connections between spheres, angles, and other projection-related results. In addition, we show how the property is useful in proving some interesting geometric results, such as the radius of the circumscribing balls is non-decreasing.

In the next chapter, we will apply the Delaunay property to study how the L-algorithm behaves locally.

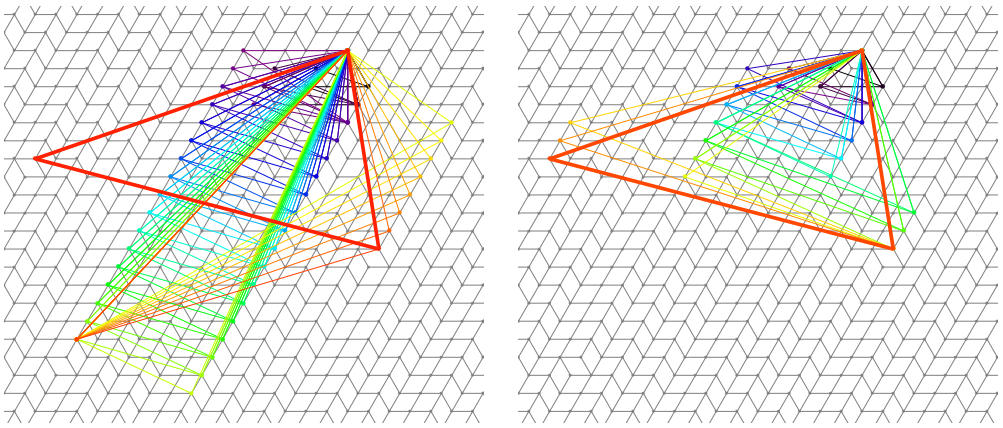
Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

CHAPTER 5

Study of locality

In sec. 2.5, we demonstrate that on digital planes, the plane-probing algorithms can accurately compute the normal vector of the digital plane. Despite the fact that the digital plane is infinite and it has unlimited amount of points at the algorithm's disposition, the algorithm only probes a finite number of points that are by construction *local* around the initial point $\mathbf{p}$. We would like to measure the space that encompasses the points visited by the algorithm in order to understand more about its behavior.

Comparison among different *tetrahedron-based* algorithms on a digital plane are illustrated in Fig. 5.1, where only the triangles corresponding to the bases are drawn. Both figures are presented in the projection of direction $\mathbf{p} - \mathbf{q}$. $\mathbf{p}$ and $\mathbf{q}$ are superimposed on a point that belong inside all triangles. We observe that the L-algorithm probes points more locally than the H-algorithm.



(A) (B)

FIGURE 5.1: The evolution for normal $(1, 73, 100)$ with H-algorithm (a) and R-algorithm (b). Here, L-algorithm has the same output as the R-algorithm. Every triangle of the evolution is superimposed. The initial triangle is blue. The last one is red.

We collect in this chapter three theoretical results about locality of plane-probing algorithms. In sec. 5.1, we show that the point $\mathbf{q}$ also projects into the triangles

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

along the estimated normal at each step. Then, we show in sec. 5.2 that for L-algorithm, the last triangle always has acute or right angles. Based on this result, we obtain a tight bound for the distance between the vertices of the triangle $\mathbf{T}^{(n)}$ and the point $\mathbf{q}$ in sec. 5.3, which will be useful in sec. 5.4 for an estimation of the minimum required space that encompasses every triangle.

5.1 Projection of q onto the triangles

For *tetrahedron-based* plane-probing algorithms, the apex of the tetrahedron does not move, stays right above the starting point and always projects into the opposite face, i.e., the base, in the direction of the starting point [LPR17, Lemma 4]. In other words, if we draw a line that passes by the points $\mathbf{p}$ and $\mathbf{q}$, it will always pass through the interior of each triangle.

Let us recall the notations introduced in sec. 4.2 in equation (4.7) (see Fig. 4.3):

$$\forall i \in \{0,\dots,n\},\ \begin{cases} \forall k \in \mathbb{Z}/3\mathbb{Z},\ \hat{\mathbf{N}}_k^{(i)} := \mathbf{m}_{k+1}^{(i)} \times \mathbf{m}_{k+2}^{(i)}, \\ \sum_{k\in\{0,1,2\}} \hat{\mathbf{N}}_k^{(i)} =: \hat{\mathbf{N}}(\mathbf{T}^{(i)}). \end{cases}$$

We also recall the following equality that describes the normal to the current triangle:

$$\forall i \in \{0,\dots,n\},\ \forall k \in \mathbb{Z}/3\mathbb{Z},\ \hat{\mathbf{N}}(\mathbf{T}^{(i)}) = \mathbf{d}_k^{(i)} \times \mathbf{d}_{k+1}^{(i)}.$$

Lemma 5.1. *For any step $\forall i \in \{0,\dots,n\}$, let $p^{(i)}(\mathbf{q})$ be the projection of the point $\mathbf{q}$ onto the plane that encompasses the current triangle $\mathbf{T}^{(i)}$ in the direction of the normal $\hat{\mathbf{N}}(\mathbf{T}^{(i)})$. The projection $p^{(i)}(\mathbf{q})$ lies inside the triangle.*

Proof. We can prove by contradiction. If the point is on or outside of the current triangle, there exists one $k' \in \mathbb{Z}/3\mathbb{Z}$ such that we can observe the following: The plane, in which the triangle lies, is divided into two half-planes by the straight line that passes the vertices $\mathbf{v}_{k'+1}^{(i)}$ and $\mathbf{v}_{k'+2}^{(i)}$. The point $\mathbf{v}_{k'}^{(i)}$ and the projection $p^{(i)}(\mathbf{q})$ belong to distinct half-plane, so does the projection of the triangle defined by $\mathbf{v}_{k'+1}^{(i)}$, $\mathbf{v}_{k'+2}^{(i)}$ and $p^{(i)}(\mathbf{q})$, and the triangle $\mathbf{T}^{(i)}$. Thus,

$$\hat{\mathbf{N}}_{k'}^{(i)} \cdot \hat{\mathbf{N}}(\mathbf{T}^{(i)}) \leq 0.$$

This contradicts the invariant property that we proved in lemma 4.8 in sec. 4.2, which is valid for all *tetrahedron-based* plane-probing algorithms. □

5.2 Acute triangle and minimal basis

On a digital plane, if $\mathbf{x}$ is one of the highest point, we know that $\mathbf{x} \cdot \mathbf{N} = \|\mathbf{N}\|_1 - 1$. We call them *upper leaning points*. These points forms a 2D lattice and any two edges of the final triangle $\mathbf{T}^{(n)}$ can be a basis of this lattice (See Fig. 5.2.) The form of the last triangle is thus closely related to the basis of the lattice of those *upper leaning points*.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

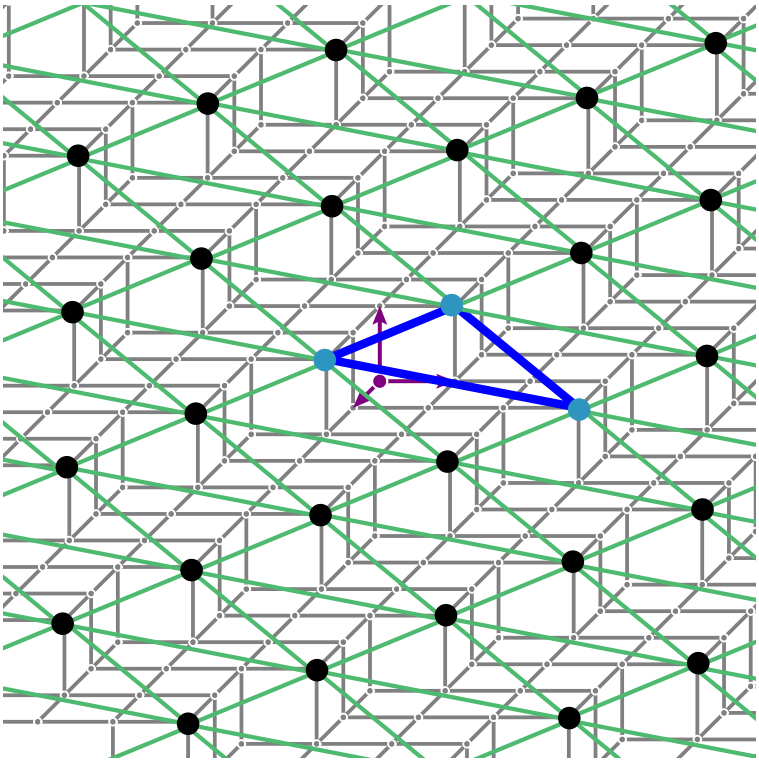


FIGURE 5.2: Illustration of upper leaning points (marked with dots) and the last triangle $\mathbf{T}^{(n)}$ (marked in blue).

One major consequence of the Delaunay property (Theorem 4.1) is the following result:

Corollary 5.1. *The final triangle* $\mathbf{T}^{(n)}$ *has acute or right angles.*

Proof. By Theorem 4.1, the circumsphere $\mathcal{B}^{(n)}$ does not contain any point of $\mathbf{P}$ in its interior. In particular, the circumcircle passing by $\mathbf{T}^{(n)}$ does not strictly contain the points $\mathbf{v}_n^{(k)} + (\mathbf{v}_n^{(k+1)} - \mathbf{v}_n^{(k)}) + (\mathbf{v}_n^{(k+2)} - \mathbf{v}_n^{(k)})$, for all $k \in \mathbb{Z}/3\mathbb{Z}$. By Lemma 4.13, this implies that the final triangle has three acute or right angles. □

This geometrical result has another consequence that requires the following definition:

Definition 5.1. *Let L be a rank-two integral lattice. A basis $(\mathbf{x}, \mathbf{y})$ of L is minimal if and only if $\|\mathbf{x}\|_2, \|\mathbf{y}\|_2 \leq \|\mathbf{x} - \mathbf{y}\|_2 \leq \|\mathbf{x} + \mathbf{y}\|_2$, where $\|\cdot\|_2$ denotes the Euclidean norm.*

Such a basis is said *minimal* because this definition matches with the well-known Minkowski's minima [Ngu09, Theorem 7]. We can retrieve a minimal basis of the underlying lattice in a digital plane from the last triangle returned by the L-algorithm.

Corollary 5.2. *The two shortest edges of the final triangle* $\mathbf{T}^{(n)}$ *form a minimal basis of the lattice of* upper leaning points.

Proof. We know by Theorem 2.2 that any two edges of the final triangle form a basis of the lattice of upper leaning points. We show below that the fact that all angles are acute or right (Corollary 5.1) implies that the two shortest edges form a *minimal* basis.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Let $\mathbf{x}, \mathbf{y}, \mathbf{z}$ be respectively equal to $(\mathbf{v}_1^{(n)} - \mathbf{v}_0^{(n)})$, $(\mathbf{v}_2^{(n)} - \mathbf{v}_0^{(n)})$ and $(\mathbf{v}_2^{(n)} - \mathbf{v}_1^{(n)})$ and assume w.l.o.g. that $\mathbf{x}$ and $\mathbf{y}$ are the two shortest vectors, i.e., $\|\mathbf{x}\|_2, \|\mathbf{y}\|_2 \leq \|\mathbf{z}\|_2$. On one hand, since $-\mathbf{z} = \mathbf{x} - \mathbf{y}$, we have by definition

$$\|\mathbf{x}\|_2, \|\mathbf{y}\|_2 \leq \|\mathbf{x} - \mathbf{y}\|_2.$$

On the other hand, since $\mathbf{x} \cdot \mathbf{y} \geq 0$ by Corollary 5.1, it is obvious that

$$\|\mathbf{x} - \mathbf{y}\|_2 \leq \|\mathbf{x} + \mathbf{y}\|_2.$$

Putting all together, we have $\|\mathbf{x}\|_2, \|\mathbf{y}\|_2 \leq \|\mathbf{x} - \mathbf{y}\|_2 \leq \|\mathbf{x} + \mathbf{y}\|_2$, which means by definition that the basis $(\mathbf{x}, \mathbf{y})$ is minimal. □

Thanks to the Delaunay property, we have been able to prove that the L-algorithm always terminates with an acute or right triangle. In addition, the two shortest edges of the final triangle form a minimal basis of the lattice of upper leaning points.

Experimentally, the H-algorithm does not always return an acute triangle (see Fig. 5.3). The R-algorithm seems to always return acute triangle as L-algorithm does, but we currently do not have theoretical proof for this observation.

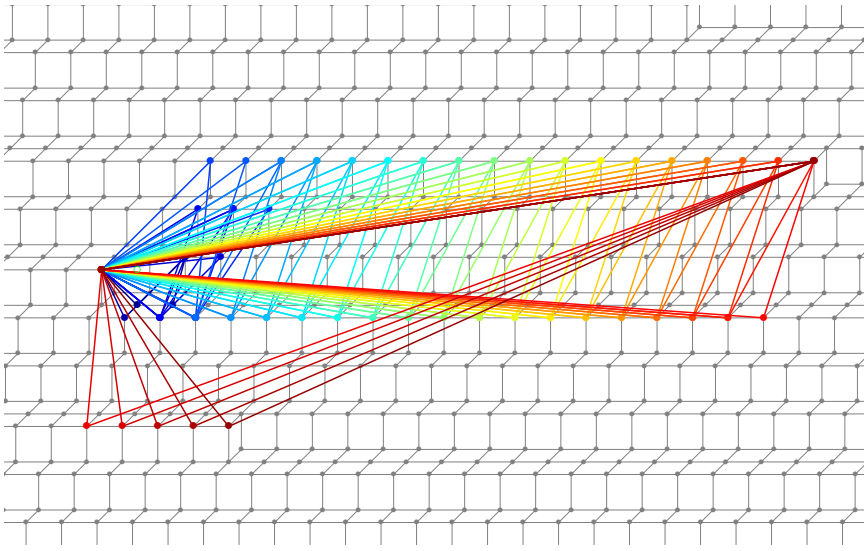


FIGURE 5.3: Example of a case where the H-algorithm does not return an acute triangle ($\mathbf{N} = (67, 1, 91)$). The ending triangle is depicted in red.

5.3 Maximal distance

In this section, we use the previous result to show that the last triangle cannot be too far away from the starting point. Fig. 5.4 displays an example of the maximal distance at each step. The figure shows a case where the H-algorithm ends with an triangle which has a further vertex with respect to the point $\mathbf{q}$ than the two other variants who have same behavior. In the following, we aim to find a bound for maximal distance on the last triangle $\mathbf{T}^{(n)}$.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

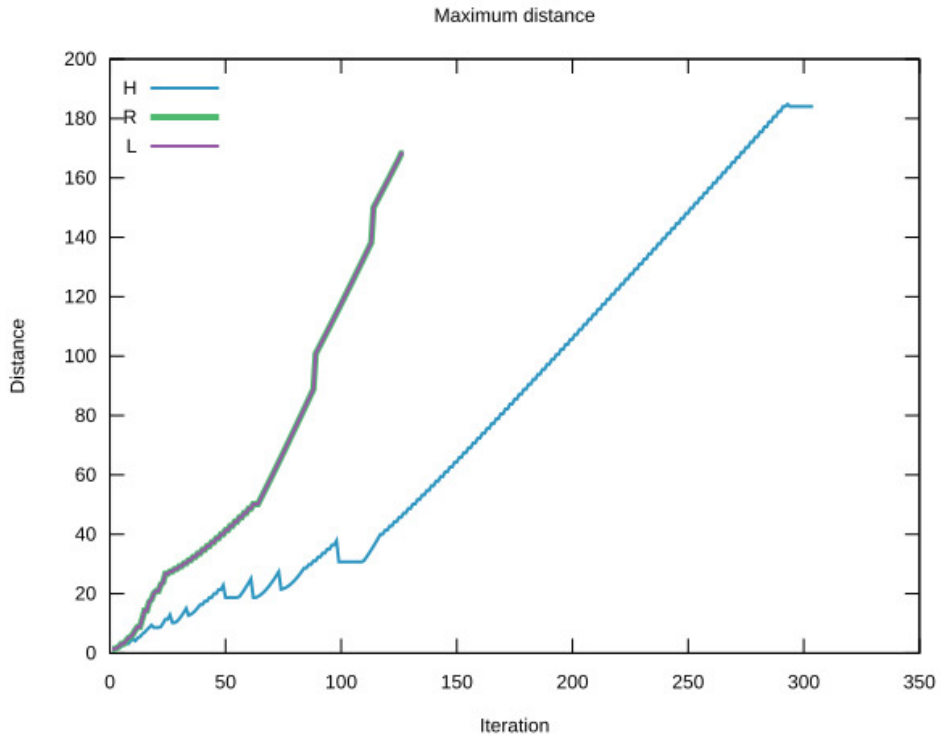


FIGURE 5.4: Maximal distance of the vertices of the last triangle $\mathbf{T}^{(n)}$ to $\mathbf{q}$ for $\mathbf{N} = (2071, 8513, 6444)$ per iteration during probings with the H, the R and the L-algorithm.

The goal of this section is to show an upper bound for the magnitude of the last three vectors $(\mathbf{m}_k^{(n)})_{k\in\mathbb{Z}/3\mathbb{Z}}$, i.e., for the distance of the three vertices $(\mathbf{v}_k^{(n)})_{k\in\mathbb{Z}/3\mathbb{Z}}$ from the fixed point $\mathbf{q}$, which is located very close to the starting point $\mathbf{p}$.

Let $\mathbf{x}, \mathbf{y}, \mathbf{z}$ be respectively equal to $(\mathbf{v}_1^{(n)} - \mathbf{v}_0^{(n)})$, $(\mathbf{v}_2^{(n)} - \mathbf{v}_0^{(n)})$ and $(\mathbf{v}_2^{(n)} - \mathbf{v}_1^{(n)})$ and assume w.l.o.g. that their magnitude are such that $\|\mathbf{x}\|_2 \leq \|\mathbf{y}\|_2 \leq \|\mathbf{z}\|_2$.

Let us focus on the rank-two lattice $L := \{\mathbf{x} \in \mathbb{Z}^3 \mid \mathbf{x} \cdot \mathbf{N} = \|\mathbf{N}\|_1 - 1\}$. Its volume, denoted by $vol(L)$, is defined as the square root of the Gram determinant of any basis $(\mathbf{b}_1, \mathbf{b}_2)$ of L [Ngu09, Definitions 3 and 7]. If we choose the basis returned by Algorithm 1 (see also Theorem 2.2), we can easily compute that $vol(L) = \|\mathbf{N}\|_2$.

By Corollary 5.2, $\|\mathbf{x}\|_2$ and $\|\mathbf{y}\|_2$ are respectively the shortest and second shortest non-zero vectors of L, i.e., the first and second Minkowski's minima of L. We can therefore relate them with $vol(L)$ and thus $\|\mathbf{N}\|_2$ using known results from lattice theory. Equation (5.1) involves Hermite's constant [Ngu09, Definition 14], whereas equation (5.2) involves Minkovski's Second Theorem [Ngu09, Theorem 5] and the trivial lower bound $\sqrt{2} \leq \|\mathbf{x}\|_2$:

$$\|\mathbf{x}\|_2^2 \leq \frac{2}{\sqrt{3}}\|\mathbf{N}\|_2, \tag{5.1}$$

$$\|\mathbf{y}\|_2 \leq \sqrt{\frac{2}{3}}\|\mathbf{N}\|_2. \tag{5.2}$$

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Furthermore, since the last triangle $\mathbf{T}^{(n)}$ has three acute or right angles by Corollary 5.1, we can use the law of cosines to bound from equations (5.1) and (5.2), the length of the longest side:

$$\|\mathbf{z}\|_2 \leq \sqrt{\frac{2}{\sqrt{3}}\|\mathbf{N}\|_2 + \frac{2}{3}\|\mathbf{N}\|_2^2}. \tag{5.3}$$

Now, for all $k \in \mathbb{Z}/3\mathbb{Z}$, let us consider the orthographic projection of $\mathbf{m}_k^{(n)}$ in direction $\mathbf{N}$ defined as:

$$p_{\mathbf{N}}(\mathbf{m}_k^{(n)}) := \mathbf{m}_k^{(n)} - \left(\mathbf{m}_k^{(n)} \cdot \frac{\mathbf{N}}{\|\mathbf{N}\|_2}\right)\frac{\mathbf{N}}{\|\mathbf{N}\|_2}.$$

Since $p_{\mathbf{N}}(\mathbf{m}_k^{(n)})$ is trivially bounded by $\|\mathbf{z}\|_2$, we can derive from (5.3) an upper bound for $\|\mathbf{m}_k^{(n)}\|_2$ as follows:

$$\begin{aligned}
\|p_{\mathbf{N}}(\mathbf{m}_n^{(i)})\|_2^2 &= \|\mathbf{m}_k^{(n)}\|_2^2 - 2\left(\mathbf{m}_k^{(i)} \cdot \frac{\mathbf{N}}{\|\mathbf{N}\|_2}\right)^2 + \left(\mathbf{m}_k^{(n)} \cdot \frac{\mathbf{N}}{\|\mathbf{N}\|_2}\right)^2 \left(\frac{\mathbf{N}}{\|\mathbf{N}\|_2}\right)^2 \\
&= \|\mathbf{m}_k^{(n)}\|_2^2 - \left(\mathbf{m}_k^{(i)} \cdot \frac{\mathbf{N}}{\|\mathbf{N}\|_2}\right)^2 \\
&\leq \frac{2}{\sqrt{3}}\|\mathbf{N}\|_2 + \frac{2}{3}\|\mathbf{N}\|_2^2 \\
\Rightarrow \|\mathbf{m}_k^{(n)}\|_2^2 &\leq \left(\mathbf{m}_k^{(n)} \cdot \frac{\mathbf{N}}{\|\mathbf{N}\|_2}\right)^2 + \frac{2}{\sqrt{3}}\|\mathbf{N}\|_2 + \frac{2}{3}\|\mathbf{N}\|_2^2
\end{aligned}$$

Since a direct consequence of Theorem 2.2 is $\mathbf{m}_k^{(n)} \cdot \mathbf{N} = 1$, we finally obtain

$$\forall k \in \mathbb{Z}/3\mathbb{Z},\ \|\mathbf{m}_k^{(n)}\|_2^2 \leq \frac{2}{3}\|\mathbf{N}\|_2^2 + \frac{2}{\sqrt{3}}\|\mathbf{N}\|_2 + \frac{1}{\|\mathbf{N}\|_2^2}.$$

Hence,

$$\max_k\{\|\mathbf{m}_k^{(n)}\|_2\} \leq \sqrt{\frac{2}{3}\|\mathbf{N}\|_2^2 + \frac{2}{\sqrt{3}}\|\mathbf{N}\|_2 + \frac{1}{\|\mathbf{N}\|_2^2}}. \tag{5.4}$$

This result shows that the last triangle has vertices not too far away from $\mathbf{q}$ and thus, from the starting point $\mathbf{p}$. More precisely, their distance to $\mathbf{q}$ is comparable to the magnitude of the normal vector of the digital plane. This provides some evidence that the L-algorithm *locally* probes the digital plane to determine its normal vector. This property is quite important for the analysis of digital surfaces with the help of a plane-probing algorithm.

Note, however, that this result is still partial, because our derivation only holds for the last triangle $\mathbf{T}^{(n)}$ and not for all previous triangles. Even if it is unlikely, points farther away might be probed in the course of the algorithm.

Experimental results We wish to estimate the proximity of the probes to the initial vertex $\mathbf{q}$ during the iterations. Experimentally, we want to know how tight the theoretical bound that we computed is. In Fig. 5.5, we measure the max distance of the last triangle computed by the L-algorithm for all normals whose l_2-norm is less

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

than 200 and compare them with the above theoretical bound. Both the theoretical bound and the bound given by experiments shows that the max distance is linear with respect to $\|\mathbf{N}\|_2$.

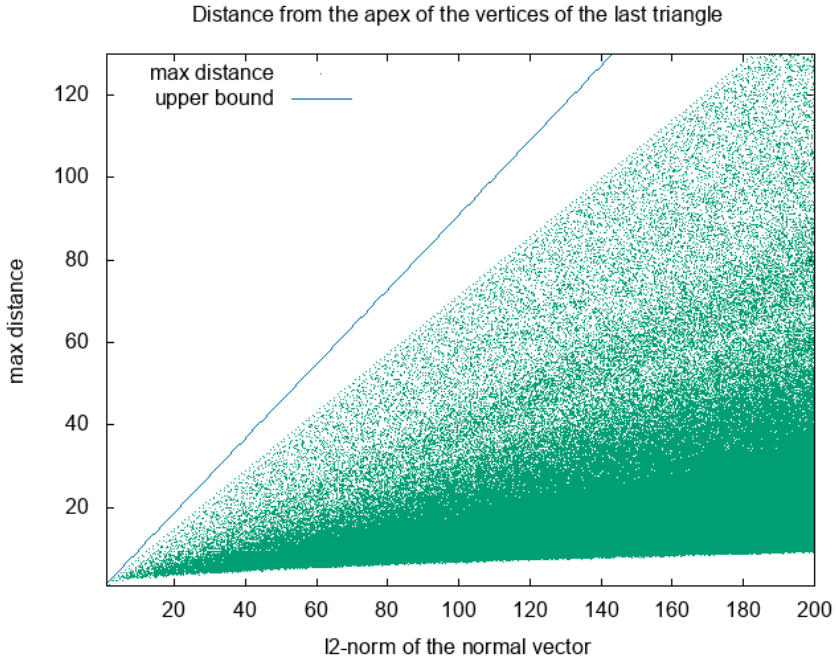


FIGURE 5.5: The relation between maximum distance and the l_2-norm of normal vectors. Each green dot corresponds to the output of the L-algorithm for a given normal vector in χ. The theoretical upper bound is in blue.

5.4 Probing space

In previous section, we obtain a upper bound for the max distance of the last triangle $\mathbf{T}^{(n)}$. Additionally, we wish to establish an area that encompasses every vertices of the triangles $\left\{\mathbf{T}^{(i)}\right\}_{\forall i\in\{0,\dots,n\}}$. We define the *probing space* as the convex hull $conv\Big(\left\{\mathbf{T}^{(i)}\right\}_{\forall i\in\{0,\dots,n\}}\Big)$. Here we show how the points are all included in a parallelepiped that contains that area.

To begin, we recall the update rule of the algorithm. For all $i \in \{0,\dots,n-1\}$, there exists a permutation $\sigma = \sigma^{(i)}$ from the group of permutation of three indices $\{0,1,2\}$, and two non-negative integers α and β such that we can rewrite the update rules:

$$\mathbf{m}^{(i+1)}_{\sigma(0)} = \mathbf{m}^{(i)}_{\sigma(0)} - \alpha\mathbf{m}^{(i)}_{\sigma(1)} - \beta\mathbf{m}^{(i)}_{\sigma(2)} \tag{5.5}$$

$$\mathbf{m}^{(i+1)}_{\sigma(1)} = \mathbf{m}^{(i)}_{\sigma(1)} \tag{5.6}$$

$$\mathbf{m}^{(i+1)}_{\sigma(2)} = \mathbf{m}^{(i)}_{\sigma(2)} \tag{5.7}$$

If we align the digital plane with its normal direction, we identify the point $\mathbf{p}$ as one of the lowest point and the point $\mathbf{q}$ as one of the highest. All the points that the algorithm visits are in between. We try to describe the set of points that encompasses

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

the probing space. We start by looking at the relation between the triangle $\mathbf{T}^{(i)}$ and the following sets:

$$\forall i \in \{0, \dots, n\}\,,\ Cone_{\mathbf{q}}{}^{(i)} := \left\{ \mathbf{q} - \sum_k c_k \mathbf{m}_k^{(i)}, c_k \in \mathbb{N}, k \in \{0,1,2\} \right\}, \tag{5.8}$$

and

$$\forall i \in \{0, \dots, n\}\,,\ Cone_{\mathbf{p}}{}^{(i)} := \left\{ \mathbf{p} + \sum_k d_k \mathbf{m}_k^{(i)}, d_k \in \mathbb{N}, k \in \{0,1,2\} \right\}. \tag{5.9}$$

For some given step $i \in \{0, \dots, n\}$, the following lemma states that the set $Cone_{\mathbf{q}}{}^{(i)}$ encompasses the vertices of all previous triangles.

Lemma 5.2. *For all $i \in \{1, \dots, n\}$, $j \in \{0, \dots, i\}$, $\mathbf{T}^{(j)} \subset Cone_{\mathbf{q}}{}^{(i)}$.*

Proof. In order to show that for all $k \in \{0,1,2\}$, $\mathbf{q} - \mathbf{m}_k^{(j)} \in Cone_{\mathbf{q}}{}^{(i)}$, we will prove by recurrence the property $P^{(i)}$: for all $k \in \{0,1,2\}$, for all $j \in \{0, \dots, i\}$, $\mathbf{m}_k^{(j)} = \sum_l c_l \mathbf{m}_l^{(i)}$ with $c_l \geq 0$ for all $l \in \{0,1,2\}$.

First, it is easy to prove $P^{(0)}$. We assume now $P^{(i)}$ for any step i in $\{1, \dots, n-1\}$ and will prove $P^{(i+1)}$.

For all $k \in \{0,1,2\}$, $\mathbf{m}_k^{(j)}$ can be seen as a linear combination of the three vectors $\mathbf{m}_0^{(i+1)}, \mathbf{m}_1^{(i+1)}, \mathbf{m}_2^{(i+1)}$, because they form a basis of $\mathbb{Z}^3$ due to Theorem 2.2. It follows:

$$\begin{aligned}
\mathbf{m}_k^{(j)} &= \sum_l c_l \mathbf{m}_l^{(i+1)} \\
&= \sum_l c_l \mathbf{m}_l^{(i)} - \alpha c_{\sigma(0)} \mathbf{m}_{\sigma(1)}^{(i)} - \beta c_{\sigma(0)} \mathbf{m}_{\sigma(2)}^{(i)} \\
&= c_{\sigma(0)} \mathbf{m}_{\sigma(0)}^{(i)} + (c_{\sigma(1)} - \alpha c_{\sigma(0)}) \mathbf{m}_{\sigma(1)}^{(i)} + (c_{\sigma(2)} - \beta c_{\sigma(0)}) \mathbf{m}_{\sigma(2)}^{(i)}.
\end{aligned}$$

Since we assume $P^{(i)}$, the coefficients are non-negative by identification, i.e.,

$$\begin{aligned}
c_{\sigma(0)} &\geq 0, \\
(c_{\sigma(1)} - \alpha c_{\sigma(0)}) &\geq 0, \\
(c_{\sigma(2)} - \beta c_{\sigma(0)}) &\geq 0.
\end{aligned}$$

As a consequence, $c_{\sigma(0)}, c_{\sigma(1)}, c_{\sigma(2)}$ are all non-negative, which proves $P^{(i+1)}$ and concludes.

□

Similarly, the two following results show that $Cone_{\mathbf{p}}{}^{(i)}$ encompasses the vertices of all previous triangles.

Lemma 5.3. *For all $i \in \{1, \dots, n-1\}$, $Cone_{\mathbf{p}}{}^{(i)} \subset Cone_{\mathbf{p}}{}^{(i+1)}$.*

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Proof. Let $x \in Cone_{\mathbf{p}}{}^{(i)}$. By definition there exist coefficients such that $x = \mathbf{p} + \sum_k c_k \mathbf{m}_k^{(i)}$. It follows:

$$\begin{aligned} x &= \mathbf{p} + \sum_k c_k \mathbf{m}_k^{(i)} \\ &= \mathbf{p} + \sum_k c_k \mathbf{m}_k^{(i+1)} + \alpha c_{\sigma(0)} \mathbf{m}_{\sigma(1)}^{(i+1)} + \beta c_{\sigma(0)} \mathbf{m}_{\sigma(2)}^{(i+1)} \\ &= \mathbf{p} + c_{\sigma(0)} \mathbf{m}_{\sigma(0)}^{(i+1)} + \underbrace{(c_{\sigma(1)} + \alpha c_{\sigma(0)})}_{\geq 0} \mathbf{m}_{\sigma(1)}^{(i+1)} + \underbrace{(c_{\sigma(2)} + \beta c_{\sigma(0)})}_{\geq 0} \mathbf{m}_{\sigma(2)}^{(i+1)} \in Cone_{\mathbf{p}}{}^{(i+1)}. \end{aligned}$$

□

Lemma 5.4. *For all $i \in \{1, \ldots, n\}$, $\mathbf{T}^{(i)} \subset Cone_{\mathbf{p}}{}^{(i)}$.*

Proof. It is trivial that $\mathbf{T}^{(0)} \subset \mathcal{D}^{(0)}$. We assume now that $\mathbf{T}^{(i)} \subset Cone_{\mathbf{p}}{}^{(i)}$ for $i \in \{0, \ldots, n-1\}$ and will prove that $\mathbf{T}^{(i+1)} \subset Cone_{\mathbf{p}}{}^{(i+1)}$. For $k \in \{\sigma(1), \sigma(2)\}$, we have $\mathbf{q} - \mathbf{m}_k^{(i+1)} = \mathbf{q} - \mathbf{m}_k^{(i)} \in Cone_{\mathbf{p}}{}^{(i)} \subset Cone_{\mathbf{p}}{}^{(i+1)}$ by Lemma 5.3. In addition, for $\sigma(0)$, we have

$$\mathbf{q} - \mathbf{m}_{\sigma(0)}^{(i+1)} = \underbrace{\mathbf{q} - \mathbf{m}_{\sigma(0)}^{(i)}}_{\in Cone_{\mathbf{p}}{}^{(i)} \text{ by hypothesis}} + \alpha \mathbf{m}_{\sigma(1)}^{(i)} + \beta \mathbf{m}_{\sigma(2)}^{(i)} \in Cone_{\mathbf{p}}{}^{(i)} \underbrace{\subset}_{\text{Lemma 5.3}} Cone_{\mathbf{p}}{}^{(i+1)}.$$

Therefore, $\mathbf{T}^{(i+1)} \subset Cone_{\mathbf{p}}{}^{(i+1)}$, which concludes. □

Combining all precedent lemmas, all the vertices of each triangle would belong to $Cone_{\mathbf{q}}{}^{(n)}$ and $Cone_{\mathbf{p}}{}^{(n)}$. That is to say, for all $i \in \{1, \ldots, n\}$ and $k \in \{0, 1, 2\}$, $\mathbf{q} - \mathbf{m}_k^{(i)} \in Cone_{\mathbf{q}}{}^{(n)}$ (by Lemma 5.2) and $\mathbf{q} - \mathbf{m}_k^{(i)} \in Cone_{\mathbf{p}}{}^{(n)}$ (by Lemma 5.4).

Now, we want to compute the intersection between $Cone_{\mathbf{q}}{}^{(n)}$ and $Cone_{\mathbf{p}}{}^{(n)}$, which lies in a parallelepiped that has edges that follow the direction $\mathbf{m}_k^{(n)}$. Our objective is to determine the integers t_0, t_1, and t_2 that satisfy

$$t_0 \mathbf{m}_0^{(n)} + t_1 \mathbf{m}_1^{(n)} + t_2 \mathbf{m}_2^{(n)} = \mathbf{q} - \mathbf{p} = (1, 1, 1). \tag{5.10}$$

The problem is actually equivalent to solving the following system:

$$\mathbf{M}^{(n)} \begin{bmatrix} t_0 \\ t_1 \\ t_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \tag{5.11}$$

where

$$\mathbf{M}^{(n)} = \begin{bmatrix} \mathbf{m}_0^{(n)}{}_x & \mathbf{m}_1^{(n)}{}_x & \mathbf{m}_2^{(n)}{}_x \\ \mathbf{m}_0^{(n)}{}_y & \mathbf{m}_1^{(n)}{}_y & \mathbf{m}_2^{(n)}{}_y \\ \mathbf{m}_0^{(n)}{}_z & \mathbf{m}_1^{(n)}{}_z & \mathbf{m}_2^{(n)}{}_z \end{bmatrix}.$$

Recall that $\det(\mathbf{M}^{(n)}) = 1$ (see theorem 2.2), then by Cramer's rule:

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

$$t_0 = \frac{\begin{vmatrix} 1 & \mathbf{m}_1^{(n)}{}_x & \mathbf{m}_2^{(n)}{}_x \\ 1 & \mathbf{m}_1^{(n)}{}_y & \mathbf{m}_2^{(n)}{}_y \\ 1 & \mathbf{m}_1^{(n)}{}_z & \mathbf{m}_2^{(n)}{}_z \end{vmatrix}}{\det(\mathbf{M}^{(n)})} = (\mathbf{m}_1^{(n)} \times \mathbf{m}_2^{(n)}) \cdot (1,1,1).$$

In the end, for all $k \in \{0,1,2\}$

$$t_k = (\mathbf{m}_{k+1}^{(n)} \times \mathbf{m}_{k+2}^{(n)}) \cdot (1,1,1) = \|\hat{\mathbf{N}}_k^{(i)}\|_1. \tag{5.12}$$

We conclude that the probing space is included in a parallelepiped that can be described as follow:

$$\left\{ \mathbf{q} - \sum_k u_k \mathbf{m}_k^{(n)} \mid 0 \leq u_k \leq \|\hat{\mathbf{N}}_k^{(i)}\|_1 \right\}. \tag{5.13}$$

In practical terms, the parallelepiped is currently larger than the precise probing space (see Fig. 5.6).

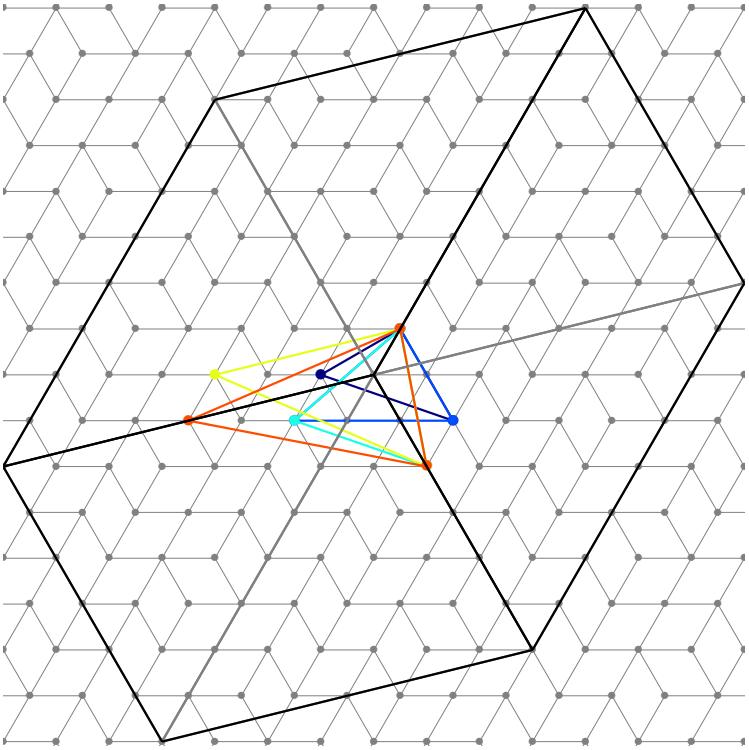


FIGURE 5.6: The evolution for normal $(1,5,7)$ with L-algorithm and the parallelpiped (black) that encompasses all vertices of triangles.

In fact, we know that the sequence of radii of the balls circumscribing consecutive triangles is non-decreasing (see Sec. 4.3). It is likely that the projections of vertices of all triangles stay in the projection of the last ball, which delimits a space that is much smaller than the parallelepiped. There is potential for improvement in the future.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

5.5 Conclusion

In this chapter, we have demonstrated that the point $\mathbf{q}$ consistently projects onto the triangles. This projection occurs in two directions: first, along the shift vector $\mathbf{q} - \mathbf{p}$, and second, along the normal vector of the current triangle. Additionally, we have delved into the connection between the algorithm's output and the minimal basis of the underlying lattice within the digital plane. Building on this, we have established a constraint for the distance between the final triangle $\mathbf{T}^{(n)}$ and the point $\mathbf{q}$. To conclude, we have provided a coarse estimation of the probing space.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

CHAPTER 6

Estimation

Within the context of plane-probing algorithms, one of the significant applications involves estimating normal fields. In previous chapters, we focused on *tetrahedron-based* algorithms on digital planes whose expected normal vector is identical everywhere. While these methods have interesting geometric properties, the true challenge lies in estimating the normal field on a digital surface for *tetrahedron-based* approaches due to their initialization conditions and stop criteria. In this chapter, we aim to overcome these limitations by introducing *parallelepiped-based* algorithms, as presented in [LMR20]. Alongside this approach, we explore existing digital methods for normal estimation and share our efforts to enhance the multigrid convergence of these algorithms on digital surfaces. We also draw comparisons between the convergence curves of our method and other established approaches, which are mostly implemented with the DGtal library.

6.1 Multigrid convergence

The topological boundary of the shape digitization is never the same as the boundary of an Euclidean object, on which the normal vector field is defined. Consequently, a direct comparison between the true normal field and the estimated one is not feasible. To address this issue, we use a geometric definition that asserts that any surface element in close proximity to the point of interest on the shape will have its estimated normal vector gradually approaching the expected one [Kle00; CLR12].

Definition 6.1. *Let $X \subset \mathbb{R}^3$ be a compact simply connected volume. The* Gauss digitization *of X with grid step $h > 0$, denoted by $\mathbf{DS}(X)_h$, is the set of all discrete points $p \in h(\mathbb{Z}^3) \equiv \{(ha, hb, hc) \,|\, (a, b, c) \in \mathbb{Z}^3\}$ lying in X. In other words,*

$$\mathbf{DS}(X)_h = X \cap h(\mathbb{Z}^3). \tag{6.1}$$

Let us denote by $\mathbb{X}$ a family of compact simply connected subsets of $\mathbb{R}^3$ and by $\mathbf{DS}(X)_h$ the Gauss digitization of $X \in \mathbb{X}$ with grid step h. For any x in the topological boundary ∂X of X, let $\mathbf{N}(X, x)$ be the unit vector normal to ∂X at x. A *discrete*

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

normal estimator is a family of mappings which associates to any digital surface $\mathbf{DS}(X)_h$ and a point $y \in \mathbf{DS}(X)_h$ some value of $\mathbb{R}^3$. Note that we usually have constant estimates per surface element.

Definition 6.2. *The estimator* $\hat{\mathbf{N}}$ *is* multigrid-convergent *for the family* $\mathbb{X}$ *if and only if, for any* $X \in \mathbb{X}$, $h > 0$, *for any* $x \in \partial X$,

$$\forall y \in \partial\mathbf{DS}(X)_h \text{ with } \|y - x\|_1 \leq h, \|\hat{\mathbf{N}}(\mathbf{DS}(X)_h, y) - \mathbf{N}(X, x)\| \leq \tau_{X,x}(h),$$

where $\tau_{X,x} : \mathbb{R}^{+*} \to \mathbb{R}^+$ *has null limit at* 0. *This function defines the speed of convergence of* $\hat{\mathbf{N}}$ *toward* $\mathbf{N}$ *at point* x *of* X. *The convergence is* uniform *for* X *when every* $\tau_{X,x}$ *is bounded from above by a function* τ_X *independent of* $x \in X$ *with null limit at* 0.

In practice, we compare the two corresponding normals by examining their angular difference rather than measuring the magnitude of their difference. Therefore, all the experimental results in this section will be shown in terms of angle errors (in radians). The method is currently implemented in the DGtal library, and Fig. 6.1 demonstrates its multigrid convergence. The mean value is converging when the grid step h decreases, but the algorithm failed to converge on some of the surfels according to the curve of maximum values. Our aim is to identify the cause of non-convergence some surfels and to improve the convergence of the method.

6.2 Normal estimation with plane-probing algorithm

In previous chapters, we show that a plane-probing algorithm calculates the normal vector of $\mathcal{P}$ by selectively testing integer points using the predicate "is x in $\mathcal{P}$?" (see Algorithm 1). While initially designed for digital planes, these algorithms can also be employed with arbitrary digital surfaces, as demonstrated in [LPR17] and [LMR20].

Definition 6.3. *We consider the digital surface as a boundary of a 3D volume. When we spot three surfels which are adjacent to each other on a digital surface, by adding an additional point* x *to the set of points of the surfels, we can construct a cube which has these points as vertices. If the point* x *lies outside of the volume, the corner created by the three adjacent surfels is a* reentrant corner.

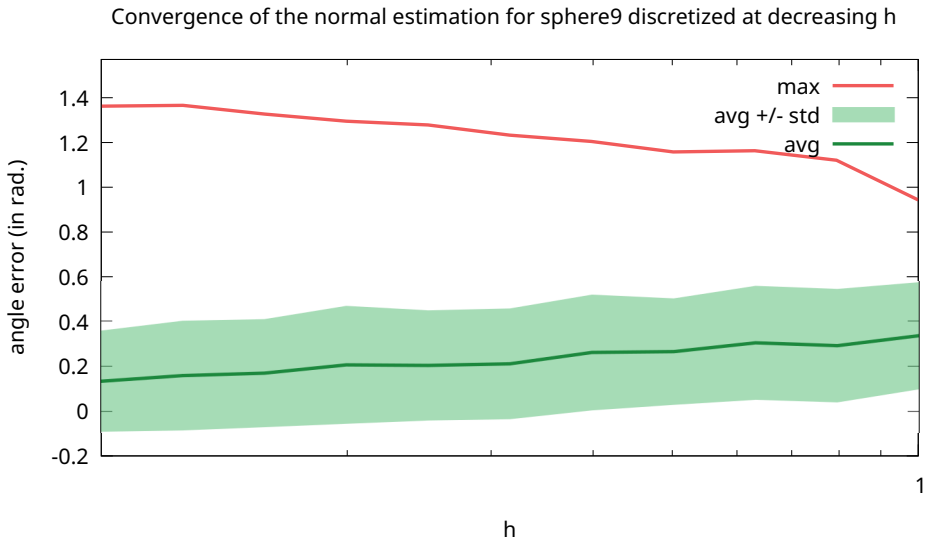
The initialization of *tetrahedron-based* plane-probing algorithms are limited to reentrant corners (see Tab. 6.1). This kind of configuration does not usually exist regularly on a digital surface(see Fig. 6.2).

Let us note **DS** as a digital surface. We can run *tetrahedron-based* plane-probing algorithms on reeentrant corners, which are defined by a point $\mathbf{p} \in \mathbf{DS}$ and an octant $\mathbf{s} \in \{\pm 1, \pm 1, \pm 1\}$ that points to a point that is outside of the digital surface **DS**. The algorithm is illustrated in Algorithm 4

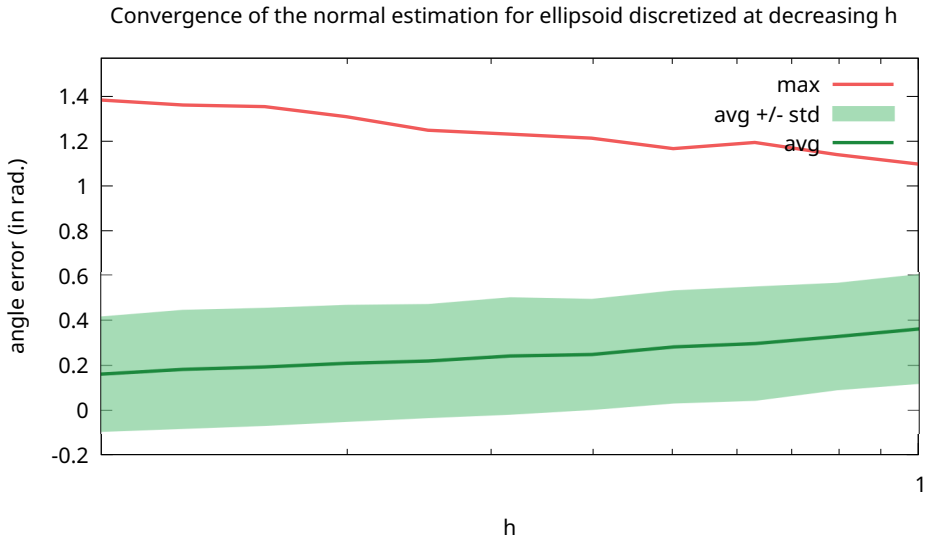
6.2.1 Parallelepiped-based plane-probing algorithm

[LMR20] introduced the *parallelepiped-based* plane-probing algorithm that is more generic to apply on digital planes. In this subsection, we compare the differences between *tetrahedron-based* and *parallelepiped-based* plane-probing algorithms.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés



(A)



(B)

FIGURE 6.1: Convergence graph of normal estimation of *parallelepiped-based* plane-probing algorithm implemented in the DGtal Library on various shapes. The parameters of the shapes are detailed in Tab. 6.1.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

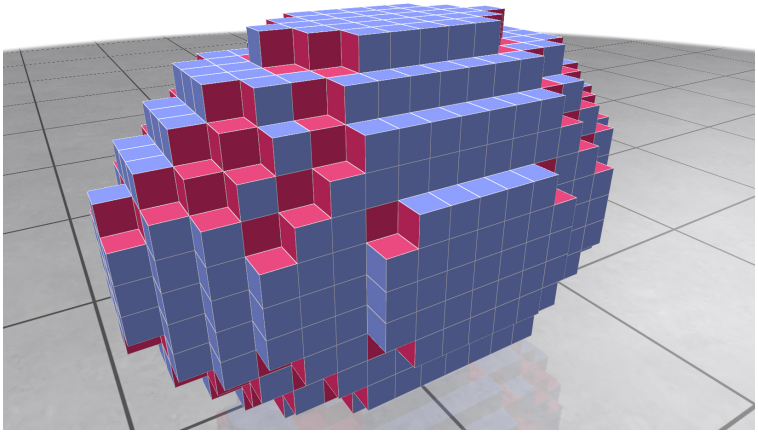


FIGURE 6.2: Illustration of the reentrant corners (red) on a digital ellipsoid.

Algorithm 4: Normal vector estimation with plane-probing algorithms

Input: The predicate InSurface := "Is a point $\mathbf{x} \in \mathbf{DS}$?", an octant $\mathbf{s} \in \{\pm 1, \pm 1, \pm 1\}$, a point $\mathbf{p} \in \mathbf{DS}$ the type of neighborhood $S \in \{S_H, S_R, S_L\}$ (see equations (2.7))

Output: A normal vector $\hat{\mathbf{N}}$

$\mathbf{q} \leftarrow \mathbf{p} + \mathbf{s}$; $(\mathbf{v}_k^{(0)})_{k\in\{0,1,2\}} \leftarrow (\mathbf{q} - \mathbf{e}_k)_{k\in\{0,1,2\}}$; // initialization

$i \leftarrow 0$;

while $\mathcal{N}_S^{(i)} \cap \{\mathbf{x} \mid \text{InSurface}(\mathbf{x})\} \neq \emptyset$ **do**

  Let (k, α, β) be such that, for all $\mathbf{y} \in \mathcal{N}_S^{(i)} \cap \{\mathbf{x} \mid \text{InSurface}(\mathbf{x})\}$,
  $\mathbf{v}_k^{(i)} + \alpha(\mathbf{q} - \mathbf{v}_{k+1}^{(i)}) + \beta(\mathbf{q} - \mathbf{v}_{k+2}^{(i)}) \leq_{\mathbf{T}^{(i)}} \mathbf{y}$; // equation (2.8)

  $\mathbf{v}_k^{(i+1)} \leftarrow \mathbf{v}_k^{(i)} + \alpha(\mathbf{q} - \mathbf{v}_{k+1}^{(i)}) + \beta(\mathbf{q} - \mathbf{v}_{k+2}^{(i)})$; // equation (2.9)

  $\forall l \in \{0,1,2\} \setminus k,\ \mathbf{v}_l^{(i+1)} \leftarrow \mathbf{v}_l^{(i)}$;

  $i \leftarrow i + 1$;

$B \leftarrow \{\mathbf{v}_0^{(i)} - \mathbf{v}_1^{(i)}, \mathbf{v}_1^{(i)} - \mathbf{v}_2^{(i)}, \mathbf{v}_2^{(i)} - \mathbf{v}_0^{(i)}\}$;

Let $\mathbf{b}_1$ and $\mathbf{b}_2$ be the shortest and second shortest vectors of B ;

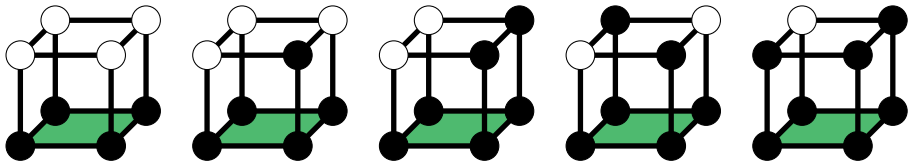
return $\mathbf{b}_1 \times \mathbf{b}_2$; // $\times$ denotes the cross product

Initialization on one surfel The *parallelepiped-based* algorithm necessitates a single surfel from the digital surface and an additional integer point situated outside the digital surface. The point, denoted as $\mathbf{p}$, along with the surfel, serves as the foundation for defining a cube, and more specifically, a parallelepiped. These initial configurations of the starting parallelepiped are limited to just five possibilities, as illustrated in Fig. 6.3, ignoring the cases that are identical by symmetry and rotation. It is worthy to note that among the five cases, there is only Fig. 6.3-(E) where we can start a *tetrahedron-based* plane-probing algorithm.

Translation of the point q The parallelepiped has an orientation, and this orientation can change when there are fewer than four vertices on the surface. In the *tetrahedron-based* case, the point $\mathbf{q}$ remains stationary, but in the *parallelepiped-based* algorithm, it may sometimes move. Thus, we note $\mathbf{q}^{(i)}$ instead of $\mathbf{q}$, where i is the iteration step. As a result, the final triangle representing the surface can end up relatively distant from the initially selected surfel.

In the previous chapter 5, we have conclusion admitting that $\mathbf{q}^{(i)} = \mathbf{q}^{(0)}$ for all steps i on a digital surface. The fact that the *parallelepiped-based* algorithm does not

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés



(A) (B) (C) (D) (E)

FIGURE 6.3: Illustration of all possible initialization configuration for *parallelepiped-based* plane-probing algorithm. The reference surfel is marked in green. A point is black if it is in the digital surface and white if it is outside.

necessarily satisfy this property might generate results that are less likely to stay true to local information.

Different predicate In *tetrahedron-based* algorithms, we check if a point x is part of **DS** by comparing its height within **DS** to the height of $\mathbf{q}^{(i)}$ in the normal direction. We do this without needing to know the specific normal direction we're looking for.

On the other hand, in *parallelepiped-based* algorithms, a broader criterion known as *NotAbove* is employed to evaluate the position of a point x, whether it belongs to **DS** or not. This evaluation is in relation to the position of $\mathbf{q}^{(i)}$ with respect to the border of the Gaussian discretization.

When dealing with digital surfaces, we utilize Algorithm 5. To select the suitable upper bound value denoted as L, careful consideration is essential. L should strike a balance, neither too small to ensure correctness nor excessively large for efficiency. In our experimental work, we set L equal to the length of the longest side of the bounding box encompassing the digital surface.

Algorithm 5: Implementation of predicate NotAbove for digital surfaces

Data: InSurface, $\mathbf{q}^{(i)}$ and an integer L
Input: A point $\mathbf{x} \in \mathbb{Z}^3$
Output: true iff $\mathbf{x}$ is less high than $\mathbf{q}^{(i)}$ in the normal direction

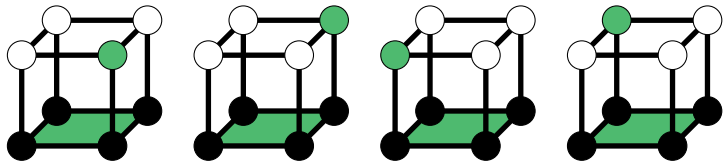
```
1 u ← x − q^(i) ; // direction
2 s ← q^(i) ; // starting point
3 l ← 1;
4 while l < L do
5 |   if InSurface(s + lu) then return True ;
6 |   l ← l + 1;
7 return False;
```

6.2.2 Pre-estimation

Sometimes, there is no unique way to initialize the algorithm. The initial parallelepiped could be visualized as a cube where the initial surfel is one of the face. If the other four points are all not in the digital surface, there are four possible ways to start the algorithm (see Fig. 6.4).

The impact of the pre-estimation remains an open problem. One could initialize the algorithm with a randomly given pre-estimation. In our estimaton, we use the result

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés



(A) (B) (C) (D)

FIGURE 6.4: The *parallelepiped-based* algorithm accept at most four different possible cases to start the algorithm. Take the case (A) in Fig. 6.3 as example, there are four points that are outside of the digital surface so there are four choices for the point $\mathbf{q}^{(0)}$ (marked in green).

of the estimation with the slice method as pre-estimaton. We devide the space above the target surfel into four octants from the middle point of the surfel in the direction of the two axes. The pre-estimated normal would point into one of the octant and we will choose the vertices that belong to the octant as point $\mathbf{q}^{(0)}$.

Algorithm 6 shows an example of *parallelepiped-based* algorithm using the NotAbove predicate.

Algorithm 6: Normal vector estimation (*parallelepiped-based*)

Input: The predicate NotAbove, an octant $\mathbf{s} \in \{\pm 1, \pm 1, \pm 1\}$, a point $\mathbf{p} \in \mathbf{DS}$ the type of neighborhood $S \in \{S_H, S_R, S_L\}$ (see equations (2.7))

Output: A normal vector $\hat{\mathbf{N}}$

$\mathbf{q} \leftarrow \mathbf{p} + \mathbf{s}$; $(\mathbf{v}_k^{(0)})_{k\in\{0,1,2\}} \leftarrow (\mathbf{q} - \mathbf{e}_k)_{k\in\{0,1,2\}}$; // initialization

$i \leftarrow 0$;

while $\mathcal{N}_S^{(i)} \cap \{\mathbf{x} \mid \text{NotAbove}(\mathbf{x})\} \neq \emptyset$ **do**

  Let (k, α, β) be such that, for all $\mathbf{y} \in \mathcal{N}_S^{(i)} \cap \{\mathbf{x} \mid \text{NotAbove}(\mathbf{x})\}$, $\mathbf{v}_k^{(i)} + \alpha(\mathbf{q} - \mathbf{v}_{k+1}^{(i)}) + \beta(\mathbf{q} - \mathbf{v}_{k+2}^{(i)}) \leq_{\mathbf{T}^{(i)}} \mathbf{y}$; // equation (2.8)

  $\mathbf{v}_k^{(i+1)} \leftarrow \mathbf{v}_k^{(i)} + \alpha(\mathbf{q} - \mathbf{v}_{k+1}^{(i)}) + \beta(\mathbf{q} - \mathbf{v}_{k+2}^{(i)})$; // equation (2.9)

  $\forall l \in \{0,1,2\} \setminus k,\ \mathbf{v}_l^{(i+1)} \leftarrow \mathbf{v}_l^{(i)}$; // compute $\mathbf{T}^{(i+1)}$, updated copy of $\mathbf{T}^{(i)}$

  if $\text{Card}(\{\mathbf{x} \in \mathbf{\Pi}^{(i)} \mid \text{NotAbove}(\mathbf{x})\}) < 4$ **then**

    compute $\mathbf{q}^{(i+1)}$, translated copy of $\mathbf{q}^{(i)}$; // $\mathbf{\Pi}^{(i)}$ is parallelepiped at step i.

  $i \leftarrow i + 1$;

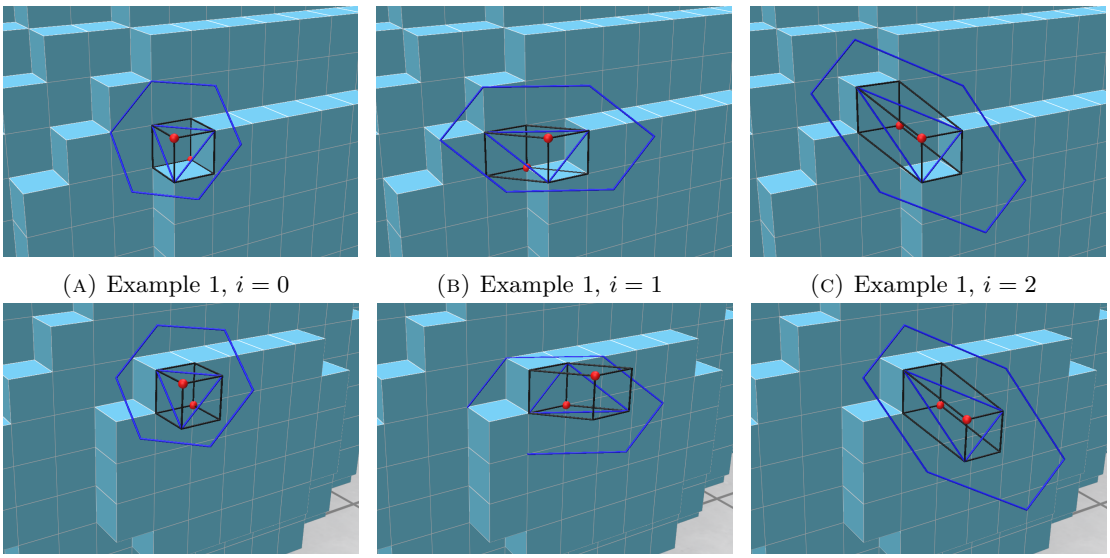
$B \leftarrow \{\mathbf{v}_0^{(i)} - \mathbf{v}_1^{(i)}, \mathbf{v}_1^{(i)} - \mathbf{v}_2^{(i)}, \mathbf{v}_2^{(i)} - \mathbf{v}_0^{(i)}\}$;

Let $\mathbf{b}_1$ and $\mathbf{b}_2$ be the shortest and second shortest vectors of B ;

return $\mathbf{b}_1 \times \mathbf{b}_2$; // $\times$ denotes the cross product

Finally, we include Tab. 6.1 summarizing the differences and similarities between the *tetrahedron-based* method and the parallelepiped method. There is only the *parallelepiped-based* version of the H-algorithm (PH) and R-algorithm (PR) in the DGtal library. We implemented the *parallelepiped-based* version of the L-algorithm (PL) and the convergence graph showed in this chapter are generated with this implementation. Two examples of the evolution of the *parallelepiped-based* algorithm are shown in Fig. 6.5. The first example shows the classic behavior like the *tetrahedron-based* algorithm, the point $\mathbf{q}$ does not change its position. The second example shows a case where the point $\mathbf{q}$ translates.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés



(A) Example 1, $i = 0$ (B) Example 1, $i = 1$ (C) Example 1, $i = 2$

(D) Example 2, $i = 0$ (E) Example 2, $i = 1$ (F) Example 2, $i = 2$

FIGURE 6.5: Illustration of the *parallelepiped-based* algorithm on an ellipsoid. The algorithm stops when the H-neighborhood does not intersect with the digital ellipsoid. The points $\mathbf{p}$ and $\mathbf{q}$ are depicted in red. We represent the parallelepiped in black and the H-neighborhood in blue.

	tetrahedron-based	parallelepiped-based
Initialization	Reentrant corner	A surfel and a point outside of the surface
$\det(\mathbf{m}_0^{(i)}, \mathbf{m}_1^{(i)}, \mathbf{m}_2^{(i)}) = 1, \forall i$	✓	✓
The point $\mathbf{q}$ always projects onto the triangle $\mathbf{T}^{(i)}$ in the direction of $\mathbf{N}$	✓	✗
The triangle $\mathbf{T}^{(i)}$ always have three vertices that belongs to the surface	✓	✗
Returns the exact normal on a digital plan	✓	✓
The point $\mathbf{q}$ is fixed	✓	✗

TABLE 6.1: Comparison between *tetrahedron-based* algorithm (H,R,L) and *parallelepiped-based* algorithm (PH,PR,PL).

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

6.3 Estimation on selected surfels

The current implementation of the plane-probing on digital surface is not doing well on some of the surfels. In Fig. 6.1, the overall average of the deviation of the estimated normal vectors to the expected ones are decreasing, but the maximum curve does not decrease. Here, we aim to identify the characteristics of the surfels, on which the plane-probing algorithm has good performance.

6.3.1 Framework of the experiment

Since the plane-probing algorithm returns the exact normal vector of a digital plane, we try to find the surfels that encourage the algorithm to performs similarly as on a digital plane. We look for similar initial condition as for *tetrahedron-based* algorithm. We also decide to add constraints on the position of the point $\mathbf{q}$. More precisely, we impose two following rules:

1. We can find a reentrant corner in the initial parallelepiped.
2. There is no translation throughout iterations.

In Fig. 6.6, we have graphically represented the angle error in relation to the grid step for the plane-probing method, applied to two different surfaces: a sphere with a radius of 9 and an ellipsoid, whose equations is given in Tab. 6.2. Each shape has undergone digitization across 20 grid steps, which were evenly distributed on a logarithmic scale ranging from $h = 1$ to h $= 0.02$. By selecting a subset of surfels carefully, the curve of maximum errors is also decreasing and we observe multigrid convergence. In the following, we would like to compare this experimental result with existing methods.

6.3.2 Comparison with existing methods for digital surface estimation

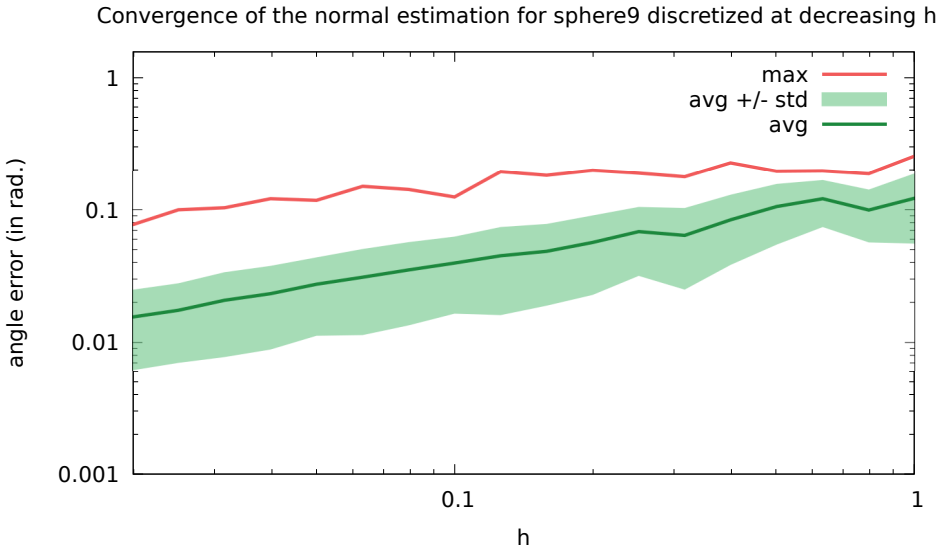
In this subsection, we present experimental results utilizing the DGtal Library. We conduct tests on various geometric convex closed shapes and generate statistical data regarding the divergence between an estimated vector and its corresponding true normal. To determine the true normal's associated point, we identify the embedded point of the surfel and seek the nearest point on the target continuous shape. We calculate the true normal as the normal of the associated point using the polynomial equation of the shape.

Shape $\mathcal{X}$	DGtal name	Equation
Sphere	sphere9	$x^2 + y^2 + z^2 - 9^2 = 0$
Ellipse	ellipse	$90 - 3 * x^2 - 2 * y^2 - z^2 = 0$

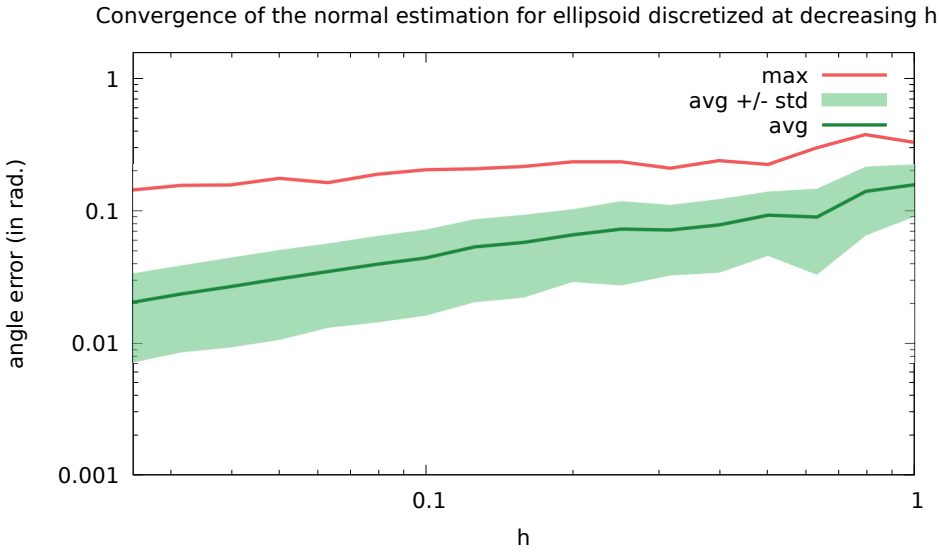
TABLE 6.2: Equations of Euclidean shapes for our experimental evaluation.

Slice In Fig. 6.7, we have conducted a comparative analysis between the method of maximum segments on 2D slices and our estimator, alongside the theoretical upper bound represented by $h^{1/3}$ for the slice approach. We show results on two discretized shape: the sphere of radius 9 and the ellipsoid described in Tab. 6.2. Each shape

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés



(A)



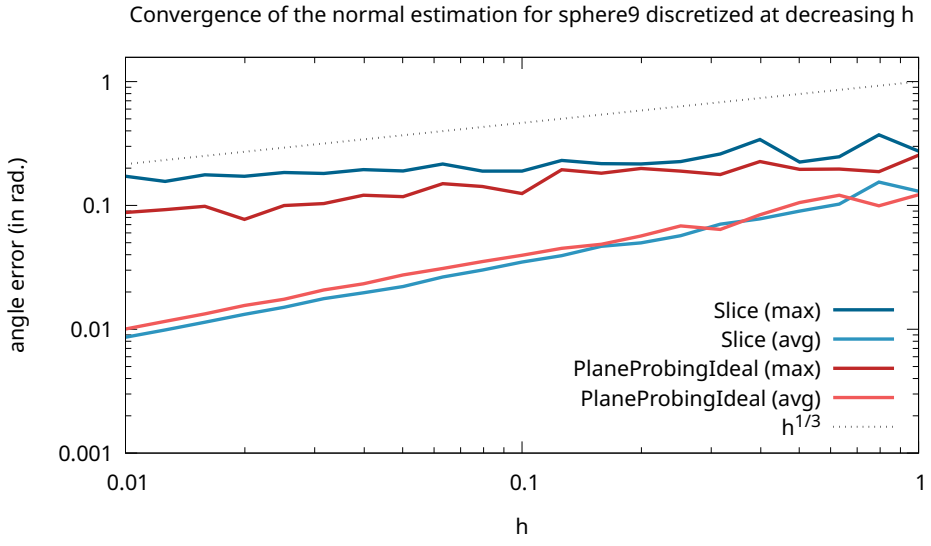
(B)

FIGURE 6.6: Convergence graph of normal estimation on selected corners only.

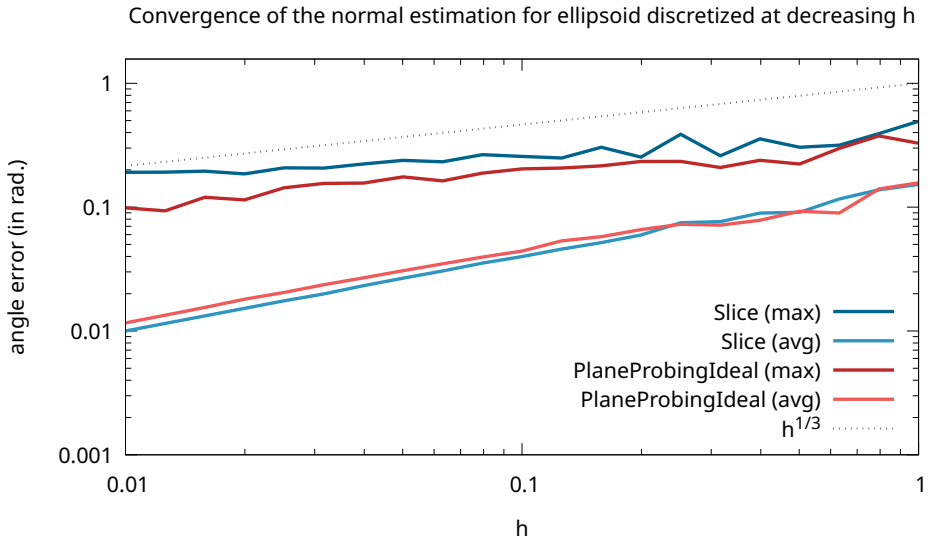
Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

has undergone digitization across 21 grid steps, which were evenly distributed on a logarithmic scale ranging from $h = 1$ to h = 0.01.

Both methods exhibit similar performance characteristics. It is noteworthy that both estimators appear to approach the ground-truth normals, on average, as the discretization step approaches zero. The performance of the two estimators on maximum error varies on different shapes, while the slice method performs slightly better on average.



(A)



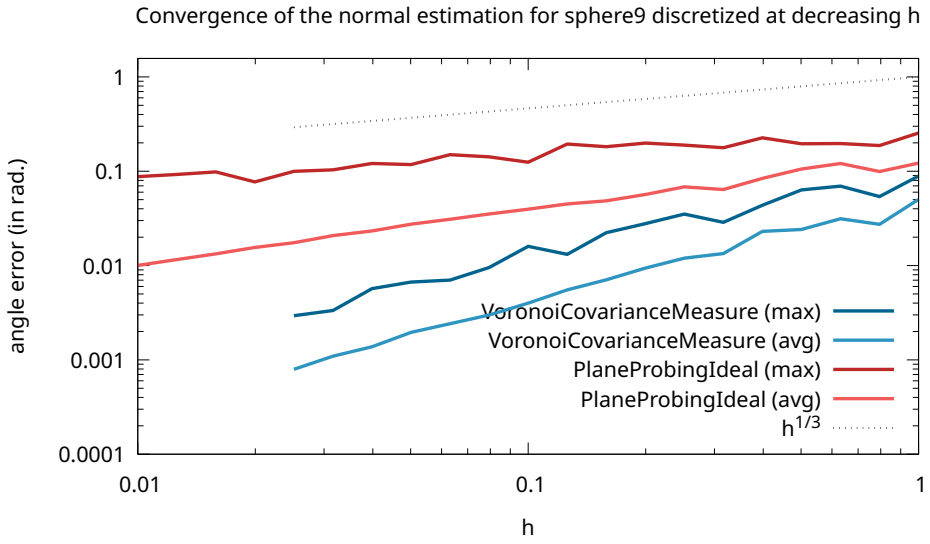
(B)

FIGURE 6.7: Convergence graph of normal estimation with max segment method.

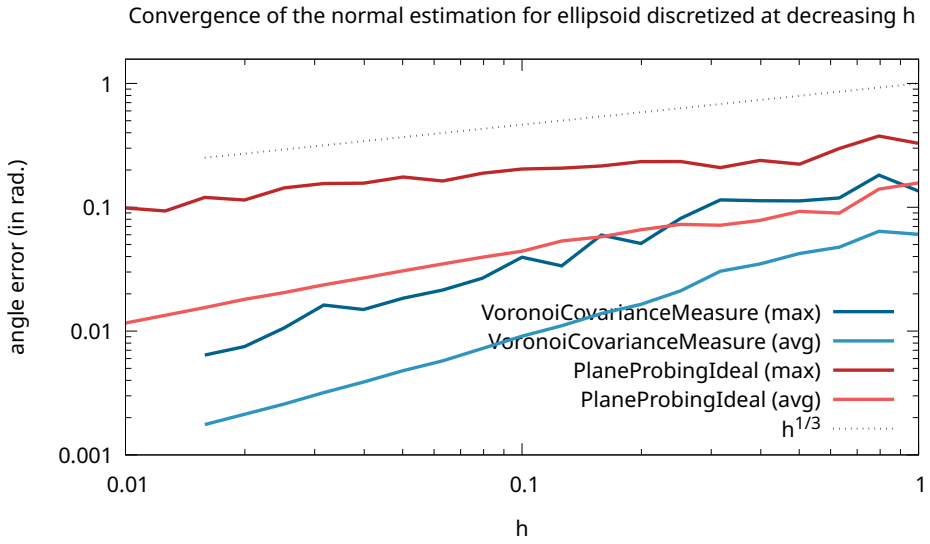
Voronoi Covariance Measure (VCM) We recall that the VCM computes a weighted average of covariance matrices from individual Voronoi cells. This method extracts local information of the surface geometry as it considers the intersection between the Voronoi diagram and a local region around a data point.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

In Fig. 6.8, we compare the performance of our estimator with a digital adaptation of this approach in the DGtal Library. We choose the same two discretized shapes: the sphere of radius 9 and the ellipsoid. Each shape has undergone digitization across 11 grid steps, which were evenly distributed on a logarithmic scale ranging from $h = 1$ to h = 0.025. The results shows that the VCM methods performs a better estimation than our method.



(A)



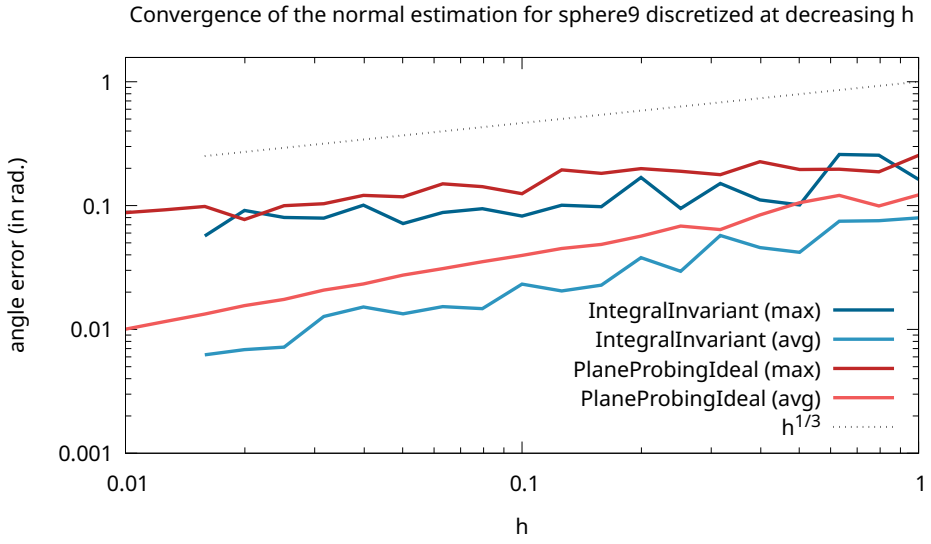
(B)

FIGURE 6.8: Convergence graph of normal estimation with voronoi covariance measure.

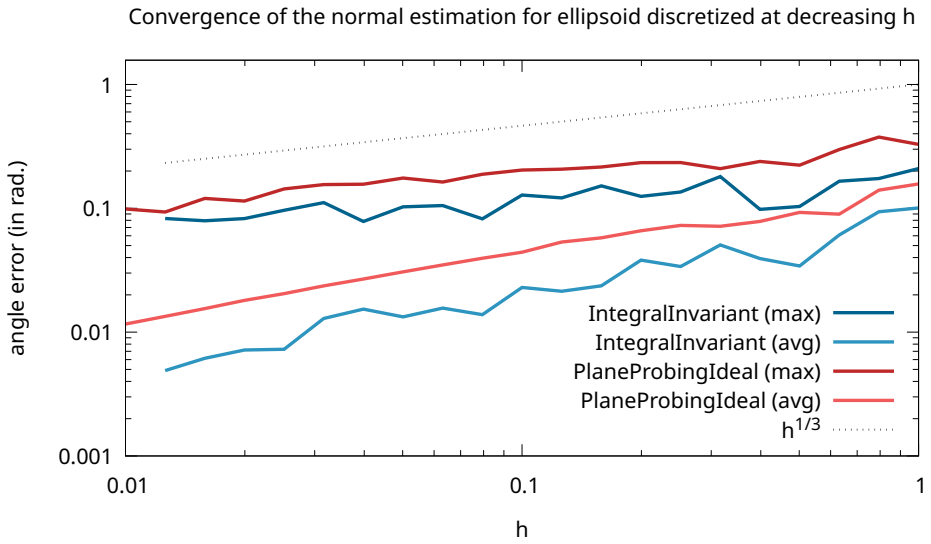
Integral Invariant The integral invariant method involves aggregating surface geometry within a spherical region, utilizing integrals over the intersection of the sphere and the mesh-bound volume to estimate principal curvatures and normals. Digital adaptations also exist in the DGtal Library.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

The choice of the ball radius is a user-defined parameter [CLL13]. We follow the implementation detail given in [LCL17] for our choice of the radius of the ball. The ball radius is expressed as $r = kh^{\frac{1}{3}}$. We choose the constant $k = 2$ such that at $h = 1$, it gives a reasonable value for the ball radius. Each shape has undergone digitization across 20 grid steps, which were evenly distributed on a logarithmic scale ranging from $h = 1$ to h = 0.019. Fig. 6.9 shows that the integral invariant method performs slightly better performance than our method.



(A)



(B)

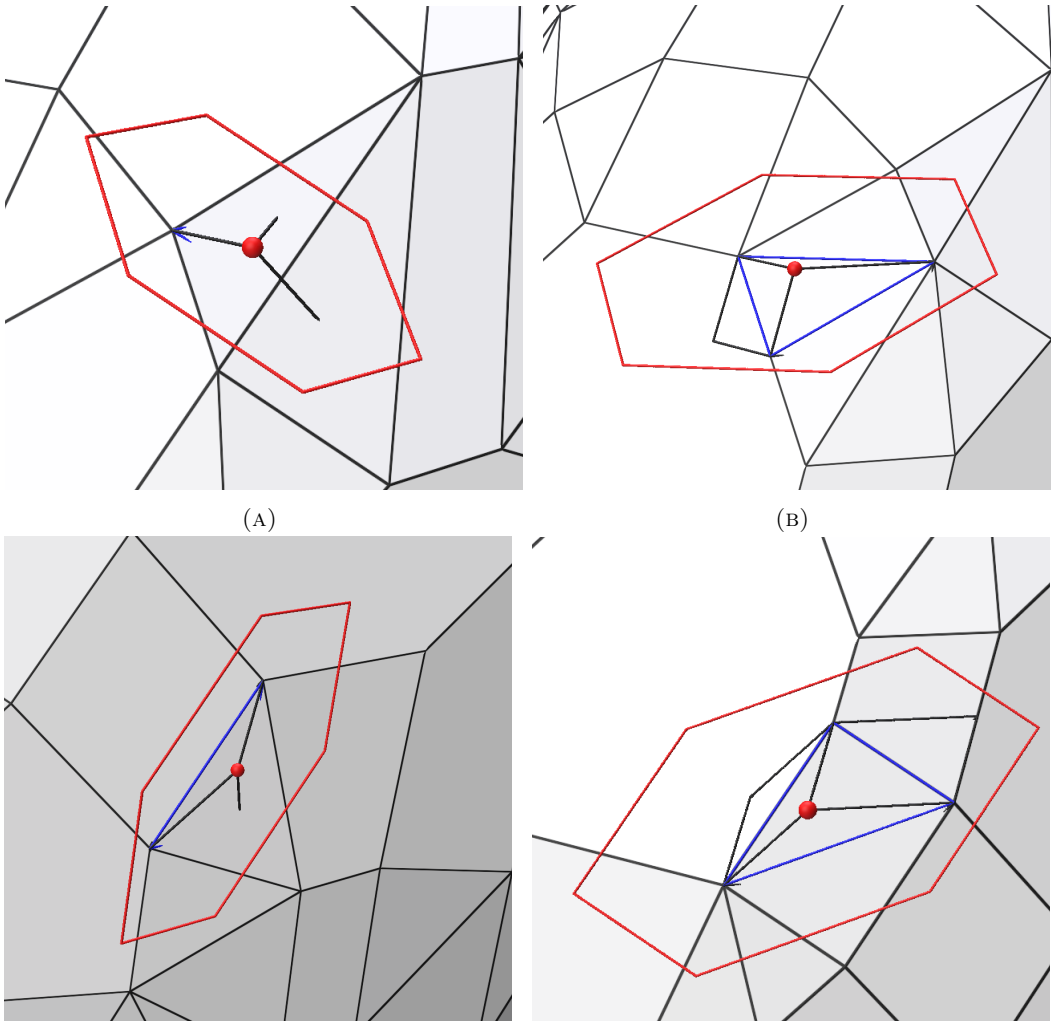
FIGURE 6.9: Convergence graph of normal estimation with integral invariant method (ball radius = 3).

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

6.3.3 Relation between estimated normals vectors and the ones on the convex hull

At a grid step of size h, we construct the convex hull of the digital volume. Our hypothesis is that the plane-probing algorithm's output closely resembles a tangent plane associated with one of the faces of the convex hull. Through experimentation, we have observed that the normal vectors associated with the faces of a digital volume's convex hull exhibit multigrid convergence toward the true normal field in its continuous counterpart.

In Fig. 6.10, we illustrate various final states resulting from the plane-probing algorithm. Fig. 6.10-(A) represents the case where the algorithm seems to stop prematurely. Fig. 6.10-(B) represents the case where the estimated normal is identical to the normal of the corresponding facet of the convex hull. Fig. 6.10-(C) represents the case where the estimated normal does not match the normal of the corresponding facet of the convex hull. Fig. 6.10-(D) represents the case where the final triangle is exactly a facet of the convex hull.



(A) (B)

(C) (D)

FIGURE 6.10: Illustration of the final triangle and the convex hull of an ellipsoid.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

6.4 Conclusion and perspectives

In this chapter, we have discussed various digital methods for estimating normals. Our exploration has included the suitability of the plane-probing algorithm for this purpose. Our efforts have been directed towards refining the existing implementation of the algorithm in the DGtal Library. In this chapter, we have discussed various digital methods for estimating normal vectors. We aim to refine the existing implementation of the algorithm in the DGtal Library.

Throughout our experiments, we made adjustments to elements like pre-estimation and ending configuration. Unfortunately, these changes did not result in significant improvements in the algorithm's performance. Subsequently, we returned to a more basic approach, limiting the algorithm's use to specific corners that closely resembled an ideal starting point for the *tetrahedron-based* plane-probing algorithm. Through this examination, we observed a pattern of multigrid convergence within this restricted context We aim to refine the existing implementation of the algorithm in the DGtal Library..

As we look forward to the future, our main goal is to broaden the scope of our normal estimation techniques to include all surfels. We also aim to enhance the algorithm's flexibility by making it applicable to non-convex shapes. These future developments hold potential for advancing the field of digital geometry and normal estimation.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

CHAPTER 7

Conclusion

In conclusion, digital geometry is intriguing due to its compatibility with octree-type spatial data structures. 3D volumes have inherent geometric limitations, but researchers have developed various methods to estimate geometric quantities. In this thesis, we focus on plane-probing algorithms that dynamically adapt and retrieve information on digital objects. These algorithms expand from a starting point, typically a triangle, and collect information by checking points that are selected based on specific criteria.

To begin, we explained the general framework of the *tetrahedron-based* plane-probing algorithm on a digital plane. We introduced the L-algorithm, a new and theoretically sound plane-probing algorithm that outperforms previous variants, namely the H-algorithm and the R-algorithm. It examines more potential points in each step, which reduces the number of steps required to achieve the correct normal vector. Both in prior studies and in our work, it has been demonstrated that the *tetrahedron-based* plane-probing algorithm precisely determines the normal vector of a digital plane.

Furthermore, we demonstrated how to probe a few points per step to ensure that the overall complexity is in $O(\|\mathbf{N}\|_1 \log(\|\mathbf{N}\|_1))$. We also explore some invariant properties of the L-algorithm. In particular, we discovered that the L-algorithm consistently produces triangles with better compactness properties, forming a local 3D Delaunay triangulation.

We then established the Delaunay property for plane-probing algorithms, and the L-algorithm adheres to it. The proof is lengthy and necessitates several technical details on the geometry of the sphere and projection. This property helps in estimating distances between points, which is valuable for determining the locality of the algorithm.

Later, we conducted a discussion on the locality of the algorithm. We studied the distance between each vertex of the triangle and the upper leaning point. As a result, we obtained a tight bound for the location of the last triangle and a broader bounding parallelepiped that encompasses all triangles' vertices.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

However, achieving accuracy in these algorithms is a challenging task on a digital surface. We attempted to enhance the multigrid convergence of the normal estimator. In this context, we identified the optimal conditions for the plane-probing algorithm to yield a reliable estimation, thereby limiting the scope of our experiments. We compared the performance of the plane-probing algorithm to other existing methods within this scope.

Looking forward, our research sets the stage for further exploration of the L-algorithm's performance, especially its locality aspect. We aim to optimize its performance and seek an alternative optimization for this variant, similar to what previous studies accomplished with the R-algorithm. In practice, we observed that the R-algorithm returns a non-obtuse triangle, much like the L-algorithm. We aim to prove this theoretically in the future.

Additionally, we aspire to establish tighter distance bounds for vertices in the tetrahedra or parallelepiped to measure how well these algorithms perform locally. This could be a crucial element in improving the normal estimator on digital surfaces. Our goals also involve extending our normal estimation techniques to cover all types of surfaces, including non-convex shapes. Another long-term objective is to develop other plane-probing-based estimators for various geometric differential quantities. These efforts will contribute to advancing the field of digital geometry and normal estimation.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

List of Figures

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

Bibliography

[AB99] N. Amenta and M. Bern. "Surface Reconstruction by Voronoi Filtering." In: *Discrete & Computational Geometry* 22.4 (Dec. 1999), pp. 481–504.

[All+07] P. Alliez et al. "Voronoi-based Variational Reconstruction of Unoriented Point Sets." In: *Proc. Eurographics Symposium on Geometry Processing*. 2007, pp. 39–48.

[Buz03] Lilian Buzer. "A linear incremental algorithm for naive and standard digital lines and planes recognition." In: *Graphical Models* 65.1 (2003). Special Issue: Discrete Topology and Geometry for Image and Object Representation, pp. 61–76. ISSN: 1524-0703. DOI: 10.1016/S1524-0703(03)00008-0.

[CK04] David Coeurjolly and Reinhard Klette. "A comparative evaluation of length estimators of digital curves." In: *IEEE Trans. Pattern Anal. Mach. Intell.* 26.2 (Feb. 2004), pp. 252–8. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2004.1262194. URL: http://www.ncbi.nlm.nih.gov/pubmed/15376899.

[CL11] E. Charrier and Jacques-Olivier Lachaud. "Maximal Planes and Multiscale Tangential Cover of 3D Digital Objects." In: *Proc. International Workshop on Combinatorial Image Analysis*. 2011, pp. 132–143.

[CLL13] David Coeurjolly, Jacques-Olivier Lachaud, and Jérémy Levallois. "Integral based Curvature Estimators in Digital Geometry." In: *17th International Conference on Discrete Geometry for Computer Imagery (DGCI 2013)*. Ed. by R. Gonzalez-Diaz, M-.J. Jimenez, and B. Medrano. 7749. Seville (Spain), Spain: Springer Verlag, Mar. 2013, pp. 215–227.

[CLR12] David Coeurjolly, Jacques-Olivier Lachaud, and Tristan Roussillon. "Multigrid Convergence of Discrete Geometric Estimators." In: *Digital Geometry Algorithms: Theoretical Foundations and Applications to Computational Imaging*. 2012, pp. 395–424.

[CP05] F. Cazals and M. Pouget. "Estimating differential quantities using polynomial fitting of osculating jets." In: *Computer Aided Geometric Design* 22.2 (2005), pp. 121–146. ISSN: 0167-8396. DOI: https://doi.org/10.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

1016/j.cagd.2004.09.004. URL: http://www.sciencedirect.com/science/article/pii/S016783960400113X.

[CP08] F. Cazals and M. Pouget. "Jet fitting 3: A Generic C++ Package for Estimating the Differential Properties on Sampled Surfaces via Polynomial Fitting." In: *ACM Trans. on Mathematical Software* 35.3 (2008).

[Cue+15] Louis Cuel et al. "Robust Geometry Estimation Using the Generalized Voronoi Covariance Measure." In: *SIAM Journal on Imaging Sciences* 8.2 (2015), pp. 1293–1314. DOI: 10.1137/140977552.

[Dar+09] Julien Dardenne et al. "Variational tetrahedral mesh generation from discrete volume data." In: *The Visual Computer* 25.5-7 (2009), pp. 401–410.

[DR94] I. Debled-Rennesson and J.P. Reveillès. "An incremental algorithm for digital plane recognition." In: *Proc. Discrete Geometry for Computer Imagery*. 1994, pp. 194–205.

[EM16] Henri-Alex Esbelin and Rémy Malgouyres. "Sparse convolution-based digital derivatives, fast estimation for noisy signals and approximation results." In: *Theor. Comput. Sci.* 624 (2016), pp. 2–24. ISSN: 0304-3975. DOI: https://doi.org/10.1016/j.tcs.2015.12.018. URL: http://www.sciencedirect.com/science/article/pii/S0304397515011792.

[EMC11] Henri-Alex Esbelin, Rémy Malgouyres, and Colin Cartade. "Convergence of binomial-based derivative estimation for C2 noisy discretized curves." In: *Theor. Comput. Sci.* 412.36 (Aug. 2011), pp. 4805–4813. ISSN: 03043975. DOI: 10.1016/j.tcs.2010.12.035. URL: http://linkinghub.elsevier.com/retrieve/pii/S030439751000736X.

[FDC03] Shachar Fleishman, Iddo Drori, and Daniel Cohen-Or. "Bilateral mesh denoising." In: *ACM transactions on graphics*. Vol. 22. 3. 2003, pp. 950–953.

[Fes05] F. Feschet. "Canonical representations of discrete curves." In: *Pattern Analysis & Applications* 8.1 (2005), pp. 84–94.

[Fli+05] F. Flin et al. "An adaptive filtering method to evaluate normal vectors and surface areas of 3D objects. Application to snow images from X-ray tomography." In: *IEEE Trans. Image Process.* 14.5 (2005), pp. 585–596.

[FM09] Sébastien Fourey and Rémy Malgouyres. "Normals estimation for digital surfaces based on convolutions." In: *Computers & Graphics* 33.1 (2009), pp. 2–10.

[FT99] F. Feschet and L. Tougne. "Optimal Time Computation of the Tangent of a Discrete Curve: Application to the Curvature." In: *Proc. Discrete Geometry for Computer Imagery*. 1999, pp. 31–40.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

[GDZ05] Y. Gérard, I. Debled-Rennesson, and P. Zimmermann. "An elementary digital plane recognition algorithm." In: *Discrete Applied Mathematics* 151.1 (2005), pp. 169–183.

[Hil+99] T. Hildebrand et al. "Direct three-dimensional morphometric analysis of human cancellous bone: microstructural data from spine, femur, iliac crest, and calcaneus." In: *Journal of bone and mineral research : the official journal of the American Society for Bone and Mineral Research* 14.7 (July 1999), pp. 1167–74. ISSN: 0884-0431.

[Hop+92] Hugues Hoppe et al. "Surface Reconstruction from Unorganized Points." In: *Proc. Conference on Computer Graphics and Interactive Techniques*. 1992, pp. 71–78. ISBN: 0-89791-479-1. DOI: 10.1145/133994.134011. URL: http://doi.acm.org/10.1145/133994.134011.

[Jon+10] Michael D Jones et al. "Directable weathering of concave rock using curvature estimation." In: *IEEE Trans. Vis. Comput. Graphics* 16.1 (2010), pp. 81–94.

[KL12] Bertrand Kerautret and Jacques-Olivier Lachaud. "Meaningful Scales Detection along Digital Contours for Unsupervised Local Noise Estimation." In: *IEEE Trans. Pattern Anal. Mach. Intell.* 34.12 (2012), pp. 2379–2392.

[Kle00] R Klette. "Multigrid Convergence of Geometric Features." In: *Digital and Image Geometry*. Ed. by G Bertrand, A Imiya, and R Klette. Vol. 2243. Lecture Notes in Computer Science. Springer, 2000, pp. 318–338. ISBN: 3-540-43079-2. DOI: 10.1007/3-540-45576-0_19. URL: http://www.informatik.uni-trier.de/~ley/db/conf/dagstuhl/dig2000.html.

[KR04] Reinhard Klette and Azriel Rosenfeld. *Digital Geometry: Geometric Methods for Digital Picture Analysis*. Ed. by Reinhard Klette and Azriel Rosenfeld. The Morgan Kaufmann Series in Computer Graphics. San Francisco: Morgan Kaufmann, 2004, pp. 231–268. ISBN: 978-1-55860-861-0.

[KS01] Reinhard Klette and Hao Jie Sun. "Digital Planar Segment Based Polyhedrization for Surface Area Estimation." In: *Visual Form*. 2001, pp. 356–366. ISBN: 978-3-540-45129-7.

[KS91] Chul E Kim and Ivan Stojmenovi. "On the recognition of digital planes in three-dimensional space." In: *Pattern Recognition Letters* 12.11 (1991), pp. 665–669. ISSN: 0167-8655. DOI: https://doi.org/10.1016/0167-8655(91)90003-5. URL: https://www.sciencedirect.com/science/article/pii/0167865591900035.

[KSA13] Viktor Kämpe, Erik Sintorn, and Ulf Assarsson. "High Resolution Sparse Voxel DAGs." In: *ACM Transactions on Graphics* 32.4 (2013).

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

[LCL17] Jacques-Olivier Lachaud, David Coeurjolly, and Jérémy Levallois. "Robust and Convergent Curvature and Normal Estimators with Digital Integral Invariants." In: *Modern Approaches to Discrete Curvature*. 2017. URL: https://hal.archives-ouvertes.fr/hal-01576020.

[LMR20] Jacques-Olivier Lachaud, Jocelyn Meyron, and Tristan Roussillon. "An Optimized Framework for Plane-Probing Algorithms." In: *Journal of Mathematical Imaging and Vision* 62 (2020), pp. 718–736.

[LPR16a] Jacques-Olivier Lachaud, X. Provençal, and Tristan Roussillon. "An output-sensitive algorithm to compute the normal vector of a digital plane." In: *Theor. Comput. Sci.* 624 (2016).

[LPR16b] Jacques-Olivier Lachaud, Xavier Provençal, and Tristan Roussillon. "Computation of the normal vector to a digital plane by sampling signicant points." In: *19th IAPR International Conference on Discrete Geometry for Computer Imagery*. Nantes, France, Apr. 2016.

[LPR17] Jacques-Olivier Lachaud, X. Provençal, and Tristan Roussillon. "Two Plane-Probing Algorithms for the Computation of the Normal Vector to a Digital Plane." In: *J. Math. Imaging Vis.* 59.1 (2017), pp. 23–39.

[LRC22] Jui-Ting Lu, Tristan Roussillon, and David Coeurjolly. "A New Lattice-Based Plane-Probing Algorithm." In: *Discrete Geometry and Mathematical Morphology*. Ed. by Étienne Baudrier et al. Cham: Springer International Publishing, 2022, pp. 366–381. ISBN: 978-3-031-19897-7.

[Lu+23] Jui-Ting Lu et al. "Delaunay property and proximity results of the L-algorithm for digital plane probing." working paper or preprint. May 2023. URL: https://hal.science/hal-04094380.

[LV03] Jacques-Olivier Lachaud and A. Vialard. "Geometric Measures on Arbitrary Dimensional Digital Surfaces." In: *Proc. Discrete Geometry for Computer Imagery*. 2003, pp. 434–443.

[LVV07] Jacques-Olivier Lachaud, Anne Vialard, and François de Vieilleville. "Fast, Accurate and Convergent Tangent Estimation on Digital Contours." In: *Image and Vision Computing* 25.10 (2007), pp. 1572–1587.

[Mar+10] Nicolas Maréchal et al. "Heat transfer simulation for modeling realistic winter sceneries." In: *Comput. Graph. Forum*. Vol. 29. 2. 2010, pp. 449–458.

[MOG11] Q. Merigot, M. Ovsjanikov, and L. J. Guibas. "Voronoi-Based Curvature and Feature Estimation from Point Clouds." In: *IEEE Trans. Vis. Comput. Graphics* 17.6 (June 2011), pp. 743–756. ISSN: 1077-2626. DOI: 10.1109/TVCG.2010.261.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

[Ngu09] P. Q. Nguyen. In: *The LLL Algorithm: Survey and Applications*. Ed. by P. Q. Nguyen and B. Vallée. Information Security and Cryptography. Springer, 2009. Chap. Hermite's Constant and Lattice Algorithms.

[Pot+09] H. Pottmann et al. "Integral Invariants for Robust Geometry Processing." In: *Computer Aided Geometric Design* 26.1 (2009), pp. 37–60.

[Rev91] J-P. Reveillès. "Géométrie Discrète, calculs en nombres entiers et algorithmique." Thèse d'Etat. Université Louis Pasteur, 1991.

[RKW91] Azriel Rosenfeld, T.Yung Kong, and Angela Y Wu. "Digital surfaces." In: *CVGIP: Graphical Models and Image Processing* 53.4 (1991), pp. 305–312. ISSN: 1049-9652. DOI: https://doi.org/10.1016/1049-9652(91)90034-H. URL: https://www.sciencedirect.com/science/article/pii/104996529190034H.

[RL19] Tristan Roussillon and Jacques-Olivier Lachaud. "Digital Plane Recognition with Fewer Probes." In: *21st IAPR International Conference on Discrete Geometry for Computer Imagery*. Vol. 11414. Lecture Notes in Computer Science. Couprie M. and Cousty J. and Kenmochi Y. and Mustafa N. Marne-la-Vallée, France: Springer, Cham, Mar. 2019, pp. 380–393.

[RS11] Tristan Roussillon and Isabelle Sivignon. "Faithful polygonal representation of the convex and concave parts of a digital curve." In: *Pattern Recognition* 44.10-11 (Oct. 2011), pp. 2693–2700. ISSN: 00313203. DOI: 10.1016/j.patcog.2011.03.018. URL: http://linkinghub.elsevier.com/retrieve/pii/S0031320311001221.

[SDC04] I. Sivignon, F. Dupont, and J.-M. Chassery. "Decomposition of a three-dimensional Discrete Object Surface into Discrete Plane Pieces." In: *Algorithmica* 38.1 (2004), pp. 25–43.

[TD99] Pierre Tellier and Isabelle Debled-Rennesson. "3D Discrete Normal Vectors." In: *Proc. Discrete Geometry for Computer Imagery*. 1999, pp. 447–458. ISBN: 978-3-540-49126-2.

[The10] The DGtal Project. *DGtal: Digital Geometry tools and algorithms library*. Ed. by David Coeurjolly and Jacques-Olivier Lachaud. 2010. URL: https://dgtal.org.

[Vee12] Peter Veelaert. "Fast Combinatorial Algorithm for Tightly Separating Hyperplanes." In: *Proc. International Workshop on Combinatorial Image Analysis*. 2012, pp. 31–44.

[VMG17] Alberto Jaspe Villanueva, Fabio Marton, and Enrico Gobetti. "Symmetry-aware Sparse Voxel DAGs (SSVDAGs) for compression-domain tracing of high-resolution geometric scenes." In: *Journal of Computer Graphics Techniques* 6.2 (May 2017), pp. 1–30. ISSN: 2331-7418.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

[Zhe+11] Youyi Zheng et al. "Bilateral normal filtering for mesh denoising." In: *IEEE Trans. Vis. Comput. Graphics* 17.10 (2011), pp. 1521–1530.

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés

FOLIO ADMINISTRATIF

THÈSE DE L'UNIVERSITÉ DE LYON OPÉRÉE AU SEIN DE L'INSA LYON

NOM : Lu **DATE DE SOUTENANCE :** 21/12/2023

PRÉNOM : Jui-Ting

TITRE : Parameter-free analysis of digital surfaces with plane probing algorithms

NATURE : Doctorat **NUMERO D'ORDRE :** 2023ISAL0123

ÉCOLE DOCTORALE : École Doctorale InfoMaths (ED512)

SPÉCIALITÉ : Informatique

RÉSUMÉ :

Les volumes 3D discrets proviennent de diverses sources, notamment la segmentation d'images, la simulation numérique, et les éditeurs basés sur les voxels. Notre intérêt réside dans le traitement de la géométrie des sur-faces discrètes entourant ces volumes, permettant la reconnaissance de structures locales telles que des segments de plans discrets. Cependant, les surfaces discrètes ont une géométrie pauvre, composée de surfels carrés parallèles aux axes. Pour analyser ces surfaces, des algorithmes de type plane-probing adaptent le voisinage autour d'un point en développant itérativement une approximation de plan, souvent sous forme de triangles, en fonction des informations locales. Notre objectif est d'analyser ces surfaces discrètes en utilisant les méthodes de type plane-probing. Nous introduisons les algorithmes de type plane-probing existants dans un cadre général. De plus, nous proposons une nouvelle variante de l'algorithme de type plane-probing qui prend en compte un voisinage plus étendu que ceux des algorithmes existants. Nous proposons également une implémentation efficace de cette nouvelle variante. Une découverte importante est que la suite de tétraèdres formée à partir de deux triangles consécutifs crée une triangulation de Delaunay dans une partie du plan discret. Cette propriété est vérifiée pour la nouvelle variante introduite. En conséquence, le triangle final retourné par l'algorithme a trois angles aigus ou droits. Ce résultat nous permet de déterminer l'étendue du voisinage considéré au cours des calculs.

Enfin, nous proposons quelques ajustements afin d'adapter ce type d'algorithme à des surfaces discrètes, permettant ainsi de déduire un estimateur de vecteurs normaux. Nous nous concentrons notamment sur la convergence multi-grille de cet estimateur, qui a été observée expérimentalement pour des positions bien identifiées sur des surfaces discrètes convexes.

MOTS-CLEFS : Géométrie discrète, Plane-probing, Estimation des normales

LABORATOIRE DE RECHERCHE : Laboratoire d'Informatique en Images et Systèmes d'Information(LIRIS)

DIRECTEUR DE THÈSE : David COEURJOLLY

PRÉSIDENTE DE JURY : Mme. Isabelle DEBLED-RENNESSON

COMPOSITION DU JURY : M. Fabien FESCHET, Mme. Yukiko KENMOCHI, Mme. Isabelle DEBLED-RENNESSON, Mme. Isabelle SIVIGNON, M. David COEURJOLLY, M. Tristan ROUSSILLON, M. Jacques-Olivier LACHAUD, M. Yan GERARD

Cette thèse est accessible à l'adresse : https://theses.insa-lyon.fr/publication/2023ISAL0123/these.pdf
© [J-T. Lu], [2023], INSA Lyon, tous droits réservés