

Estimation géométrique des tangentes à partir de coniques et algèbre géométrique. Exemples sous GAviewer

Jean-Paul Becar, Laurent Fuchs, Lionel Garnier

► **To cite this version:**

Jean-Paul Becar, Laurent Fuchs, Lionel Garnier. Estimation géométrique des tangentes à partir de coniques et algèbre géométrique. Exemples sous GAviewer. GTMG 2018, 2018, AIX EN PROVENCE, France. hal-02508953

HAL Id: hal-02508953

<https://hal-uphf.archives-ouvertes.fr/hal-02508953>

Submitted on 16 Mar 2020

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.

Estimation géométrique des tangentes à partir de coniques et algèbre géométrique. Exemples sous GViewer

Jean-Paul BECAR¹, Laurent FUCHS² et Lionel GARNIER³

¹Université de Valenciennes et

²Université de Poitiers

³Université de Bourgogne-Franche Comté

Résumé

Les nombres complexes sont fortement liés à la géométrie plane. Si les rotations, symétries et similitudes planes parmi d'autres s'expriment aisément à l'aide des complexes, le mérite de ce mode de représentation prend tout son sens dans toutes les compositions de ces transformations. Cette algébrisation des problèmes géométriques est le fil directeur de l'article. Elle fournit ainsi des formules de calcul aisément exploitables sur ordinateur. A titre d'exemple, l'article propose un algorithme géométrique de calcul de tangentes à une conique, son adaptation au contexte de l'algèbre géométrique et son implémentation au moyen d'un logiciel dédié. L'algorithme repose sur le théorème de Pascal sur les coniques connu sous le nom d'hexagramme mystique. La traduction de l'algorithme dans le formalisme de l'algèbre géométrique laisse envisager de nouvelles applications dans d'autres dimensions comme le calcul des plans tangents à une surface quadrique.

Mots-clés : coniques, algèbre géométrique, GViewer

1. Introduction

Les nombres complexes sont fortement liés à la géométrie plane. Si les rotations, symétries et similitudes planes parmi d'autres s'expriment aisément à l'aide des complexes, le mérite de ce mode de représentation prend tout son sens dans toutes les compositions de ces transformations. Cette algébrisation des problèmes géométriques est le fil directeur de l'article.

En géométrie de la CAO et en particulier issu de la numérisation 3D, le modéleur géométrique dispose de points situés dans un plan. Les points constituent la seule information disponible. Elle peut se compléter par une estimation des tangentes en chacun des points. Des méthodes existantes d'estimation de tangentes sont améliorées par celle qui repose sur l'utilisation de coniques. Les différents résultats sont exposés dans la section 2. Celle présentée ici concerne la détermination d'une tangente à une conique définie par cinq points. Elle repose sur le théorème de Pascal sur les coniques connu sous le nom d'hexagramme mystique. L'algorithme de calcul repose sur un mode opératoire proche du produit vectoriel usuel. Il devient ainsi proche des préoccupations traitées par l'algèbre géométrique. est revisité puis

illustré du point de vue calcul à l'aide d'un outil logiciel dédié aux calculs dans les algèbres géométriques. La section 3 rappelle les éléments de base de l'algèbre géométrique qui donne dès lors une nouvelle approche de l'algorithme mentionné. A titre d'illustration, le logiciel GViewer dédié aux calculs dans les algèbres géométriques est détaillé en section 4. La traduction de l'algorithme dans le formalisme de l'algèbre géométrique laisse envisager de nouvelles applications dans d'autres dimensions comme le calcul des plans tangents à une surface quadrique. Ces travaux ont été rendus possibles par l'usage de la plateforme Renater d'une part et le partage de savoir et savoir faire lors des journées de réunion du groupe de travail en modélisation géométrique qui renforcent ce type de collaboration.

2. Les coniques d'approximation

Les courbes planes d'interpolation de points sont couramment employées en géométrie de la CAO. Cependant, les tangentes ou les courbures qui fournissent des détails supplémentaires ne figurent pas dans les jeux de données.

De nombreux auteurs ont ainsi développé des méthodes d'estimation de tangentes à partir de ces données ponctuelles [Aki70, Far01]. Les méthodes connues considèrent

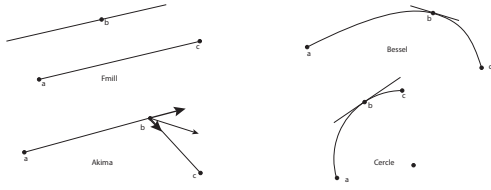


Figure 1: Méthodes d'estimation de tangentes

trois points comme celles développées par Fmill, Bessel et Akima ainsi que celle qui fait appel au cercle défini par l'ensemble des trois points (voir figure 1).

La méthode de Fmill, considère trois points quelconques et la droite passant par le point central et parallèle à corde déterminée par les deux points extrêmes fournit une estimation de la tangente. La méthode de Bessel consiste à tracer l'unique parabole passant par les trois points et prendre la tangente à la parabole au point central comme tangente d'estimation. La méthode d' Akima repose sur une estimation du vecteur tangent au point central comme une combinaison linéaire de 2 vecteurs formés à partir des autres points. Les coefficients de combinaison linéaire sont établis en considérant que les segments de droite sont parcourus à la même vitesse. Enfin, le calcul de l'unique cercle passant par trois points fournit également une technique de détermination de la tangente estimée.

Cependant, une autre méthode existe et a été développée par [ABFH05]. Elle fait appel au théorème de Pascal sur les coniques appelé hexagramme de Pascal ou encore hexagramme mystique (Figure 2 à gauche). A partir de six points quelconques sur une conique les intersections de paires de cotés opposés sont alignés et définissent la droite de Pascal. Lorsque deux points consécutifs de la conique deviennent confondus, la droite de Pascal sert de support à la tangente à la conique (Figure 2 à droite). L'algorithme est

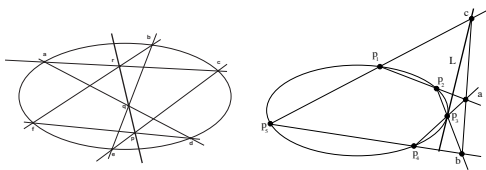


Figure 2: L'hexagramme de Pascal et construction d'une tangente

alors élégant et efficace. Élégant car il est très de proche de l'algèbre géométrique et efficace car il est en $O(h^4)$ où h est le pas de mesure. Le détail des démonstrations se trouve dans [ABFH08]. L'algorithme est précisé dans la suite.

Un point $\tilde{p}(p_0, p_1, p_2)$ dans l'espace projectif P^2 est noté

en minuscule et son correspondant affine dans E^2 est noté $p(p_1/p_0, p_2/p_0)$. Une droite dans l'espace projectif est représentée par une lettre en capitales L ses coordonnées $L = [L_0, L_1, L_2]$ sont issues de l'équation $L_0 + L_1.x + L_2.y = 0$. Les points p, q forment la droite $L = \tilde{p} \times \tilde{q}$ où \times est un opérateur calqué sur le produit vectoriel soit

$$L = [p_1q_2 - p_2q_1, -p_0q_2 + q_0p_2, p_0q_1 - q_0p_1]$$

Les droites $L = [L_0, L_1, L_2]$ et $M = [M_0, M_1, M_2]$ se coupent en le point $L \times M$ où

$$L \times M = [L_1M_2 - L_2M_1, -L_0M_2 + M_0L_2, L_0M_1 - M_0L_1]$$

L'algorithme d'estimation d'une tangente nécessite 5 points du plan. Les points p_1, p_2, p_3, p_4, p_5 pris dans cet ordre déterminent une seule conique et fournissent la droite de Pascal par le jeu des paires de points considérés. La tangente s'appuie alors sur un des points de la droite. Cette situation est illustrée dans la figure 3.

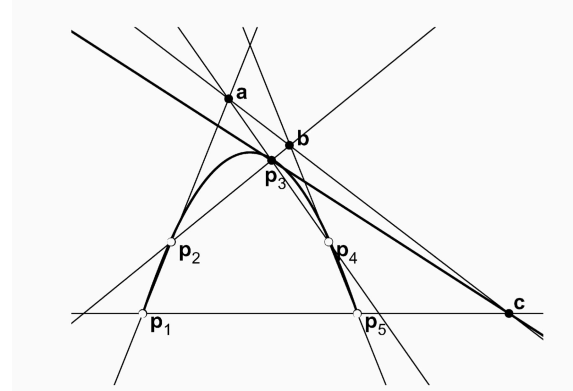


Figure 3: Approximation d'une tangente à partir du théorème de Pascal.

L'algorithme 1 construit une tangente au point p_3 à la conique définie par les points p_1, p_2, p_3, p_4, p_5 en exploitant la situation illustrée sur la figure 3.

Algorithme :

Si l'algorithme s'exprime sur le produit vectoriel, il en découle une autre représentation grâce à l'algèbre géométrique laissant envisager d'autres applications. La section suivante rappelle les fondements de l'algèbre géométrique et ses principales propriétés. L'objectif est de fournir tous les outils pour que l'algorithme puisse être exploité dans le monde informatique de l'algèbre géométrique. Le logiciel choisi pour la mise en oeuvre des applications est GViewer développé dans [DFM17].

3. Algèbre géométrique

L'algèbre géométrique [DFM17] est issue des travaux et résultats successifs d'Argand et de Gauss sur l'usage des

Algorithme 1 : Estimation de tangente par la droite de Pascal

```

Entrée : Cinq points  $p_1, p_2, p_3, p_4, p_5$ 
Sortie : La tangente  $T$ 

/* Calcul de deux points de la droite
   de Pascal */
 $a := L_{12} \times L_{34}$ 
 $b := L_{54} \times L_{32}$ 

/* Calcul de la droite de Pascal */
 $L := a \times b$ 

/* Calcul du troisième point sur la
   droite de Pascal */
 $c := L_{15} \times L$ 

/* Calcul de la tangente en  $p_3$  */
 $T := p_3 \times c$ 
    
```

complexes en géométrie, d'Hamilton avec les quaternions, de Grassmann avec la théorie des espaces vectoriels et enfin de Clifford et son algèbre associée aux formes quadratiques. C'est David Hestenes en 1960 qui a associé les travaux de Dirac et Pauli à l'algèbre géométrique en unifiant les manipulations symboliques de la physique. Elle trouve de nos jours de nouvelles applications dans des domaines variés comme l'astronomie, la vision, le traitement d'images et la robotique. Dans la suite sont rappelées la définition du produit vectoriel et ses propriétés retrouvées dans le monde de l'algèbre géométrique.

3.1. Le produit vectoriel et ses propriétés

Dans l'espace vectoriel R^3 muni de la base orthonormée $\vec{e}_1, \vec{e}_2, \vec{e}_3$ le produit vectoriel de $a(a_1, a_2, a_3)$ et $b(b_1, b_2, b_3)$ est le vecteur $a \times b$ s'exprimant à l'aide du déterminant

$$a \times b = \begin{vmatrix} \vec{e}_1 & \vec{e}_2 & \vec{e}_3 \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{vmatrix}$$

dont les principales propriétés figurent dans le tableau :

| | |
|---|-------------------|
| $a \times b = -b \times a$ | anticommutativité |
| $(a + b) \times c = a \times c + b \times c$ | linéarité |
| $(\lambda \cdot a) \times b = \lambda \cdot (a \times b)$ | linéarité |

3.2. Algèbre géométrique

Un espace vectoriel est un espace sur lequel sont définies une addition et une multiplication par un scalaire. Le vecteur λa admet pour module $|\lambda|$ fois le module du vecteur a . L'idée algébrique du module ou de la norme d'un vecteur se trouve dans le produit intérieur ou produit scalaire

$$a \cdot b = |a| \cdot |b| \cdot \cos \theta$$

où θ est l'angle entre les vecteurs a et b . Ainsi $a \times b$ est un vecteur de module $= |a| \cdot |b| \cdot \sin \theta$ et perpendiculaire à a et b ce qui pose problème en dimension 2 et ≥ 4 . Il est alors remplacé par l'opérateur \wedge ou produit extérieur ou produit progressif. Le module de $a \wedge b$ est $|a| \cdot |b| \cdot \sin \theta$. L'objet $a \wedge b$ est une aire orientée dans le plan contenant les vecteurs a et b et s'appelle un bivecteur ou vecteur de degré 2. Par exemple, dans R^2 muni de la base orthonormée \vec{e}_1, \vec{e}_2 en dimension 2, soit $\vec{u}_1(x_1; y_1)$ et $\vec{u}_2(x_2; y_2)$. Alors :

$$\begin{aligned} \vec{u}_1 \wedge \vec{u}_2 &= (x_1 \vec{e}_1 + y_1 \vec{e}_2) \wedge (x_2 \vec{e}_1 + y_2 \vec{e}_2) \\ &= x_1 y_2 \vec{e}_1 \wedge \vec{e}_2 + y_1 x_2 \vec{e}_2 \wedge \vec{e}_1 \\ &= (x_1 y_2 - y_1 x_2) \vec{e}_1 \wedge \vec{e}_2 \\ &= \begin{vmatrix} x_1 & x_2 \\ y_1 & y_2 \end{vmatrix} \vec{e}_1 \wedge \vec{e}_2 \end{aligned}$$

d'où :

$$\begin{aligned} \vec{u}_1 \wedge \vec{u}_2 &= \det(\vec{u}_1; \vec{u}_2) \cdot \vec{e}_1 \wedge \vec{e}_2 \\ &= \|\vec{u}_1\| \|\vec{u}_2\| \sin(\angle(\vec{u}_1; \vec{u}_2)) \cdot \vec{e}_1 \wedge \vec{e}_2 \end{aligned}$$

Produit vectoriel et dualité de Hodge : En dimension 3 l'opérateur \wedge et l'opérateur dual de Hodge noté $*$ établissent le lien avec le produit vectoriel usuel.

Soit $\vec{e}_1, \vec{e}_2, \vec{e}_3$ une base orthonormée de R^3 et $\vec{u}_1(x_1; y_1; z_1)$ et $\vec{u}_2(x_2; y_2; z_2)$ 2 vecteurs quelconques exprimés dans cette base. Il vient d'une part

$$\vec{u}_1 \times \vec{u}_2 = \begin{vmatrix} \vec{e}_1 & x_1 & x_2 \\ \vec{e}_2 & y_1 & y_2 \\ \vec{e}_3 & z_1 & z_2 \end{vmatrix}$$

$$= \begin{vmatrix} y_1 & y_2 \\ z_1 & z_2 \end{vmatrix} \vec{e}_1 - \begin{vmatrix} x_1 & x_2 \\ z_1 & z_2 \end{vmatrix} \vec{e}_2 + \begin{vmatrix} x_1 & x_2 \\ y_1 & y_2 \end{vmatrix} \vec{e}_3$$

d'autre part

$$\begin{aligned} \vec{u}_1 \wedge \vec{u}_2 &= (x_1 \vec{e}_1 + y_1 \vec{e}_2 + z_1 \vec{e}_3) \wedge (x_2 \vec{e}_1 + y_2 \vec{e}_2 + z_2 \vec{e}_3) \\ &= x_1 y_2 \vec{e}_1 \wedge \vec{e}_2 + x_1 z_2 \vec{e}_1 \wedge \vec{e}_3 + y_1 x_2 \vec{e}_2 \wedge \vec{e}_1 \\ &\quad + y_1 z_2 \vec{e}_2 \wedge \vec{e}_3 + z_1 x_2 \vec{e}_3 \wedge \vec{e}_1 + z_1 y_2 \vec{e}_3 \wedge \vec{e}_2 \\ &= (x_1 y_2 - y_1 x_2) \vec{e}_1 \wedge \vec{e}_2 + (x_1 z_2 - z_1 x_2) \vec{e}_1 \wedge \vec{e}_3 \\ &\quad + (y_1 z_2 - z_1 y_2) \vec{e}_2 \wedge \vec{e}_3 \\ &= \begin{vmatrix} x_1 & x_2 \\ y_1 & y_2 \end{vmatrix} (\vec{e}_1 \wedge \vec{e}_2) + \begin{vmatrix} x_1 & x_2 \\ z_1 & z_2 \end{vmatrix} \vec{e}_1 \wedge \vec{e}_3 + \\ &\quad \begin{vmatrix} y_1 & y_2 \\ z_1 & z_2 \end{vmatrix} \vec{e}_2 \wedge \vec{e}_3 \end{aligned}$$

Le lien entre $\vec{u}_1 \times \vec{u}_2$ et $\vec{u}_1 \wedge \vec{u}_2$ est établi par :

$$\begin{cases} (\vec{e}_1 \wedge \vec{e}_2)^* &= \vec{e}_3 \\ (\vec{e}_1 \wedge \vec{e}_3)^* &= -\vec{e}_2 \\ (\vec{e}_2 \wedge \vec{e}_3)^* &= \vec{e}_1 \end{cases}$$

ainsi,

$$(\vec{u}_1 \wedge \vec{u}_2)^* = \begin{vmatrix} x_1 & x_2 \\ y_1 & y_2 \end{vmatrix} (\vec{e}_1 \wedge \vec{e}_2)^* + \begin{vmatrix} x_1 & x_2 \\ z_1 & z_2 \end{vmatrix} (\vec{e}_1 \wedge \vec{e}_3)^* \\ + \begin{vmatrix} y_1 & y_2 \\ z_1 & z_2 \end{vmatrix} (\vec{e}_2 \wedge \vec{e}_3)^*$$

ce qui donne

$$\vec{u}_1 \times \vec{u}_2 = (\vec{u}_1 \wedge \vec{u}_2)^*$$

Le lien entre le produit vectoriel dans R^3 et le produit extérieur fournit une autre approche de l'algorithme de détermination de la tangente. L'algèbre conforme propose un mode représentation des droites évoqué dans la suite. Cette représentation est implémentée sous GAViewer.

4. Espace affine dans l'algèbre conforme $G_{3,1}$ ou $G_{4,1}$

Dans une algèbre géométrique, une variété affine V (droite, plan) de dimension k est représentée par un point P de V , son espace vectoriel attaché \vec{V} muni d'une base $(\vec{u}_i)_{i \in [1:k]}$ est caractérisé par :

$$\vec{p} \wedge_{i=1}^k \vec{p}_i \wedge \vec{e}_\infty$$

où \vec{p} est la représentation du point P . On pose $\vec{A}_k = \wedge_{i=1}^k \vec{p}_i$. La variété affine est :

$$\left\{ \vec{m} \mid \vec{m} \wedge \vec{p} \wedge_{i=1}^k \vec{p}_i \wedge \vec{e}_\infty = \vec{0} \right\}$$

où \vec{m} est la représentation du point M dans l'algèbre conforme. Notons que :

$$\vec{e}_\infty \wedge \vec{p} \wedge_{i=1}^k \vec{p}_i \wedge \vec{e}_\infty = \vec{0}$$

c'est-à-dire que toute variété affine passe par le point à l'infini.

Par cohérence avec ce qui précède, lorsque $V = \{P\}$, nous avons $\vec{V} = \{\vec{0}\}$ et nous posons par convention :

$$\vec{p} \wedge \vec{0} \wedge \vec{e}_\infty = \vec{p} \wedge \vec{e}_\infty$$

et l'équation :

$$\vec{m} \wedge \vec{p} \wedge \vec{e}_\infty = \vec{0}$$

n'admet qu'une solution affine \vec{p} . Evidemment, \vec{e}_∞ est solution c'est-à-dire que la variété affine $\{P\}$ passe par le point à l'infini.

5. Algorithmes sous GAViewer

5.1. Le logiciel GAViewer

GAViewer [FDBM10] est un logiciel d'apprentissage et d'expérimentation pour l'algèbre géométrique. Il a été développé par D. Fontijne, L. Dorst, T. Bouma et S. Mann des universités d'Amsterdam et de Waterloo.

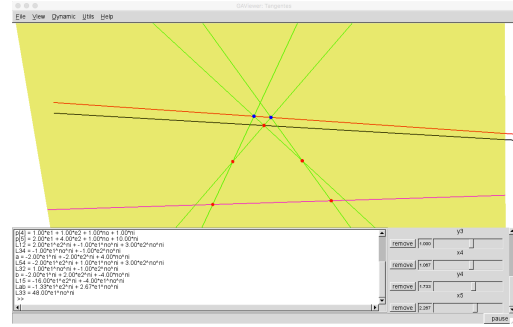


Figure 4: La fenêtre d'interaction de GAViewer

Il s'agit d'un logiciel interactif où l'on peut entrer interactivement des formules de l'algèbre géométrique à l'aide d'un langage de programmation et observer le résultat dans une fenêtre de visualisation, la figure 4 montre une vue de GAViewer où notre algorithme est présenté. La fenêtre est divisée en trois parties, la partie supérieure visualise les objets géométriques. La partie inférieure gauche permet d'entrer les formules écrites dans le langage de programmation et la partie inférieure droite affiche les curseurs qui permettent de définir les valeurs des différentes variables utilisées. La figure 5 montre la situation générique vue à la section 2

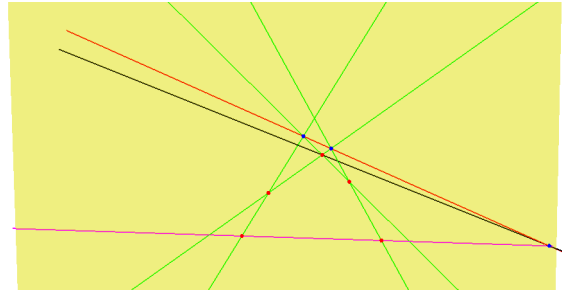


Figure 5: La situation de la figure 3 dans GAViewer

5.2. Traduction de l'algorithme

Le langage de programmation propose toute les structures de contrôle usuelles (boucles, conditionnelles), permet de gérer des variables et des structures de données élémentaires comme des tableaux. Il y a aussi la possibilité de charger des programmes à partir de fichiers.

Tous les opérateurs de l'algèbre géométrique sont disponibles dans le langage. Il est ainsi possible de traduire directement l'algorithme de la section 2 en un programme GAViewer. La figure 6 montre le code que l'on obtient avec la particularité de ramener les calculs en dimension trois de GAViewer dans un plan P défini par les trois p_0 , p_a et p_b . Cela se fait simplement en divisant par le plan P .

L'opération `op` correspond au produit extérieur \wedge . L'opération `join` est l'opération qui correspond au produit extérieur en tenant compte que l'on manipule une variété affine, il faut donc faire attention à ne pas introduire deux fois le point à l'infini représenté par \vec{e}_∞ (sinon on obtient $\vec{0}$ car toute variété affine passe par le point à l'infini).

```

pa = c3ga_point(1, 0, 0);
pb = c3ga_point(0, 1, 0);
po = c3ga_point(0, 0, 0);

L1 = op(op(po, pa), ni);
L2 = op(op(po, pb), ni);

P = join(L1, L2),

dynamic{point:
  p[1] = c3ga_point(x1, y1, 0),
  p[2] = c3ga_point(x2, y2, 0),
  p[3] = c3ga_point(x3, y3, 0),
  p[4] = c3ga_point(x4, y4, 0),
  p[5] = c3ga_point(x5, y5, 0),
};

dynamic{algotgt:
  L12 = op(op(p[1], p[2]), ni),
  L34 = op(op(p[3], p[4]), ni),

  a = op(L12/P, L34/P)/P,

  L54 = op(op(p[5], p[4]), ni),
  L32 = op(op(p[3], p[2]), ni),

  b = op(L54/P, L32/P)/P,

  L15 = magenta(
    op(op(p[1], p[5]), ni)),

  Lab = red(join(a, b)),

  c = op(L15/P, Lab/P)/P,

  L33 = black(op(p[3], c)),
};

```

Figure 6: Algorithme d'approximation des tangentes dans le langage de GAViewer.

L'intérêt d'utiliser les opération de l'algèbre géométrique est qu'ils évitent de devoir se préoccuper des cas particuliers. Ainsi, les cas particuliers (quand la conique dégénère en droites [ABFH05]) de l'algorithme 1 sont inclus dans le code présenté comme le montre la figure 7. Dans ces figures, on voit que des éléments particuliers de l'algèbre apparaissent, deux vecteurs qui ne sont pas attachés à une position définie et un scalaire. L'interprétation de ces éléments n'a pas encore été étudiée.

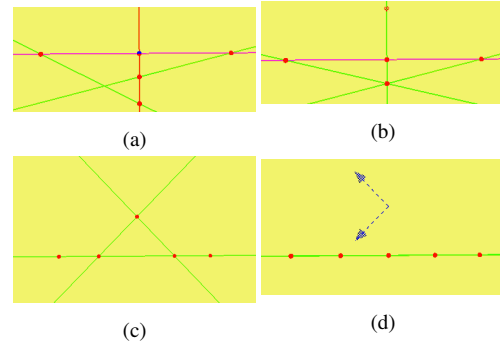


Figure 7: Les différents cas particuliers de l'algorithme quand la conique dégénère en droites. Dans les cas (b) et (d) des éléments particuliers de l'algèbre apparaissent.

6. Conclusion et perspectives

L'article a rapproché trois points de vue différents (géométrie, algorithmique géométrique et algèbre géométrique) pour formaliser et étudier un algorithme d'estimation de tangentes à une courbe. Cet algorithme a été directement mis en œuvre dans le logiciel GAViewer sans se préoccuper du codage des cas particuliers.

La traduction du formalisme classique (produit vectoriel et variétés affines) dans le formalisme de l'algèbre géométrique se fait aisément et peut être étendue à d'autres algorithmes.

Une fois l'algorithme traduit et codé dans le formalisme de l'algèbre géométrique, les cas particuliers de celui-ci ont pu être étudiés et de nouveaux éléments sont apparus. Il s'agit maintenant de pouvoir les interpréter.

Une autre voie qui est aussi ouverte est l'extension de l'algorithme en dimension supérieure pour estimer des plans tangents à des surfaces. En effet, les opérations de l'algèbre géométrique sont valables en toute dimension et ainsi les algorithmes écrits pour une dimension peuvent être étendus aux autres dimensions.

Remerciements

Les auteurs remercient chaleureusement leurs universités et laboratoires respectifs sans lesquels un tel travail n'aurait pu se réaliser. Appuyés par l'incontournable réseau Renater, les auteurs ont ainsi partagé leurs savoir et savoir faire.

Références

- [ABFH05] ALBRECHT G., BÉCAR J.-P., FARIN G., HANSFORD D. : Détermination de tangentes par l'emploi de coniques d'approximation. *Journal of Association for Computing Machinery. Vol. 1*, Num. 1 (2005).
- [ABFH08] ALBRECHT G., BÉCAR J.-P., FARIN G., HANSFORD D. : On the approximation order of tangent

- estimators. *Computer Aided Geometric Design*. Vol. 25 (2008).
- [Aki70] AKIMA H. : A new method of interpolation and smooth curve fitting based on local procedure. *Journal of Association for Computing Machinery*. Vol. 17, Num. 4 (1970).
- [DFM17] DORST L., FONTIJNE D., MANN S. : *Geometric Algebra For Computer Science, An Object Oriented Approach to Geometry*. Morgan Kaufmann Publishers, 2017.
- [Far01] FARIN G. : *Curves and Surfaces for CAGD*. Morgan Kaufmann, 2001.
- [FDBM10] FONTIJNE D., DORST L., BOUMA T., MANN S. : GViewer, interactive visualization software for geometric algebra. Downloadable at <http://www.geometricalgebra.net/downloads.html>, 2010.