Topology-Oriented Convex Hull Algorithm for Objects Bounded by Bézier Curves

Manuela Neagu, Bernard Lacolle

LMC-IMAG Tour IRMA
BP 53 38041 GRENOBLE Cédex 09
Manuela.Neagu@imag.fr, Bernard.Lacolle@imag.fr

Résumé

Il existe de nombreux algorithmes de calcul pour l'enveloppe convexe d'un polygone simple. Il existe aussi des propositions d'algorithmes calculant, à l'aide de quelques primitives assez complexes, l'enveloppe convexe d'objets bornés non pas par des segments de droite, mais par des courbes planes suffisament lisses et convexes par morceaux. Les résultats fournis par les algorithmes actuels sont des objets approchant l'enveloppe convexe de l'objet courbe donné, la qualité de l'approximation dependant de la précision avec laquelle les primitives peuvent être résolues. Résoudre ces primitives demande la résolution d'équations algébriques de degré élevé.

Ce rapport de recherche présente une approche differente du problème de l'enveloppe convexe d'objets non-linéaires, placée dans un contexte "calcul géométrique". Ainsi, toute la discussion est menée sous l'hypothèse que les équations algébriques ne peuvent pas être résolues de manière exacte (Chazelle, 1996). Pour concrétiser cette approche, nous proposons un algorithme calculant l'enveloppe convexe d'une *poly-courbe*, qui est union de courbes paramétriques contrôlées par des points ayant un polygone de contrôle simple et convexe. Ces courbes doivent satisfaire des propriétés assez générales ; pour fixer les idées, nous avons choisi les courbes de Bézier pour la présentation de nos résultats.

La méthode que nous proposons met l'accent sur les aspects topologiques. La notion d'enveloppe convexe topologique est introduite et nous montrons qu'elle peut être obtenue par des discrétisations convenables de l'objet courbe. La discrétisation semble une approche naïve et coûteuse, mais nous prouvons, premièrement, que l'objet linéaire obtenu capture tous les caractéristiques topologiques de celui non-linéaire, et deuxièmement, que la taille de ce polygone reste raisonable. Du point de vue pratique, nous donnons une méthode de construction de l'objet linéaire mentionné qui ne demande pas de résolution d'équations algébriques.

Mots-clef : enveloppe convexe, courbe de Bézier, subdivision, polygone de contrôle, topologie

Abstract

There exist many algorithms computing the convex hull of a simple polygon. There also exist some proposals of algorithms computing, using a few complex "oracles", the convex hull of objects bounded not by straight line segments, but by smooth piecewise convex planar curves. The results provided by the existing algorithms are objects approaching the convex hull of the given curved object, the quality of the approximation depending on the accuracy the oracles can be solved with. Solving these oracles implies high-degree algebraic equations resolution.

This paper presents a different approach for the convex hull problem for non-linear objects, placed in a "geometric computation" framework. So, all our discussion is made under the assumption that algebraic equations can not be accurately solved (Chazelle, 1996). To make this approach concrete, we propose an algorithm computing the convex hull of a *polycurve*, an union of point-controled parametric curves with simple and convex control polygon. The properties that these parametric curves must fulfill are general; to fix the ideas, we choose the Bézier curves for the presentation of our results.

The method we propose is topology-oriented. The notion of topological convex hull is introduced and we prove that it can be obtained via suitable discretisations of the curved object. Discretisation seems a naïve and expensive approach, but we show that, firstly, the obtained linear object captures all the topological features of the curved one, and secondly, the size of this polygon is reasonable. From a practical point of view, we give a construction method for the mentioned linear object that does not require the resolution of algebraic equations.

Keywords: convex hull, Bézier curve, subdivision, control polygon, topology

1. Introduction

The problem of computing a convex hull is a main topic in computational geometry. Preparata and Shamos (1985) give interesting applications of the convex hull of a set of points. For more complex objects, the convex hull is effectively making easier certain geometric decisions. In the intersection decision problem, testing the intersection of the objects' convex hulls is an efficient first test. Motion planning of convex objects among convex obstacles is known to be easier than general motion planning and thus the convex hull is useful in robotics.

Computing the convex hull of what we will call *linear* objects (i.e. sets of points, segments, or polygons) is a widely studied problem. For a set of n points, algorithms running in $\mathcal{O}(n \log n)$ are optimal (Graham, 1972), meanwhile if a structure of simple polygon is given linear-time algorithms can be conceived (Graham and Yao, 1983).

But linear objects do not suffice in real-world modeling; this rises the problem of treating 'curved' objects. Using a small set of *oracles*, a few authors gave algorithms for computing the convex hull of objects with curved boundary. By modifying the vertex-based algorithm of (Graham and Yao, 1983) in an edge-based one, (Schäffer and Van Wyk, 1987) give an algorithm for objects bounded by piecewise-smooth Jordan curves. (Dobkin and Souvaine, 1990) suggest an algorithm for a large class of objects with curved boundary, called *splinegons*. This algorithm uses an implicit *bounding polygon* for the given object. Both these algorithms are proved linear, considering that the necessary oracles (for example, computing the intersection of a straight line and a curve, or the - at most four - common supporting lines of two curves) are exactly and in constant time feasible. Going one step farther, (Bajaj and Kim, 1991) propose an algorithm for objects bounded by piecewise-algebraic curves whose complexity, $\mathcal{O}(nd^{\mathcal{O}(1)})$, takes into account the oracles' complexity, $\mathcal{O}(d^{\mathcal{O}(1)})$.

If we study the mentioned oracles, we remark that they imply the resolution of high-degree algebraic equations. Even if efficient numerical methods can be employed for this, we have decided to adopt in this paper the "geometric computation" approach (Chazelle, 1996). In our framework, it is considered that tools for algebraic equations resolution are not available; thus, the oracles can not be employed.

In this paper, we will firstly interest in the *topological structure* of the convex hull. As far as we know, only (Cho, Maekawa and Patrikalakis, 1996) adopted a similar approach, but for a different problem, the linear approximation of composite Bézier curves. Our method claims a complete explicit definition of the desired result, as well as one of the object to be treated. We chose dealing with objects whose boundary is an union of *completely convex* point-controled parametric curves (parametric curves with simple and convex control polygon). The most important geometrical properties required for these curves are the variation diminishing and the inclusion of the curve in its control polygon's convex hull, and the totality of the necessary properties is satisfied by numerous types of parametric curves: Bézier and rational Bézier curves, B-splines and NURBS, etc. These properties allowed us giving a correct description of the topology of the object's convex hull, in a sense which will be defined. To fix the ideas, we have chosen among these parametric curves the Bézier curves, choice motivated by the wide utilization of the Bézier curves in geometric modeling and design (e.g. Farin (1997), Hoschek and Lasser, (1993)).

The main feature of our algorithm is that it never deals directly with the curved boundary of the object, but only with polygonal approximations of this one. These polygonal approximations are naturally supplied by the control polygons of the Bézier curves, via the de Casteljau subdivision. Concretely, we establish the conditions under which the topological structure of the convex hull's boundary is completely defined and we subdivide a Bézier curve of the boundary every time it does not fulfill these conditions.

The mentioned subdivision of the curves bounding the given object seems implying a complete discretisation of the boundary, which would be too expensive in time and memory place. Yet, the experimental results show that the number of vertices of the polygon approaching the boundary grows, meanly, about 3.6 times, even if the average number of iterations applied is 6.1. This means that the subdivisions are to be done very locally (6 iterations generate, if the subdivisions are uniformly performed on the boundary, a growth of 2^6 times of the number of vertices).

Let us now discuss the choice we have made concerning the de Casteljau subdivision. Indeed, we have chosen the fixed subdivision parameter $1/2$, which might seem an artificial and limitative decision, compared to the use of adaptative subdivision parameter. But the computation of a really efficient adaptative subdivision parameter obviously requires algebraic equations resolution. Even if in this case we can tolerate some imprecision, we have for instant decided to completely banish the algebraic equations from our framework. Of course, the study of the choice of the subdivision parameter can be a natural improvement of our method.

This paper is structured as follows: Section 2 presents the motivations and the interest of a new method of computation for the convex hull and introduces the concept of *topological convex hull*. Section 3 defines the polycurve and its control polygon and gives some of their main properties. In Section 4, we formalise the notion of topological convex hull and prove that it can be obtained from the expression of the control polygon's convex hull. Section 5, we formulate linear-type criteria for deciding whether the control polygon provides the topological convex hull of the polycurve or not, and we study the degeneracies that may occur. In Section 6, we describe and discuss the algorithm we propose. Finally, Section 7 presents some examples and numerical results and Section 8 concludes.

2. The convex hull problem

2.1. Background

The convex hull of a polygon is bounded by a convex polygon, so computing it amounts to giving the (ordered) list of vertices of this convex polygon. All these vertices are vertices of the polygon we compute the convex hull for. In the case of curve objects, the convex hull's boundary is more complex and composed of non-trivial pieces of the object's boundary and line segments connecting them.

The first question one asks is how the pieces of the object's boundary featuring in the convex hull's boundary are defined. Of course, this depends on the definition of the object's boundary itself. The most convenient and intuitive way of defining such a piece is as the part of the object's boundary between two points A and B, considered in a given direction along the boundary. We must then focus on the definition of the points A and B.

If we see the convex hull's boundary as an elastic band around the object, the points

A and B are the points where the elastic loses (or gets in) contact with the object's boundary. Mathematically, these points are roots of (systems of) equations which, if the object's boundary is a (piecewise) algebraic curve, are algebraic; in this case, the equations are briefly discussed in (Bajaj and Kim, 1991).

In the papers treating the convex hull problem for curve objects that we have mentioned in the introduction, (Bajaj and Kim, 1991), (Dobkin and Souvaine, 1990) and (Schäffer and Van Wyk, 1987), it is asked the object's boundary to be (at least) piecewise convex. This property is retained in our framework by the use of completely convex Bézier curves (Bézier curves with convex control polygon). It allows a separation of the pairs of points A and B: a convex piece of the object's boundary can contain at most one such pair. This is due to the fact that a convex part of the object's boundary cannot contain two disconnected parts of the convex hull's boundary, property that will be essential in our proofs. Under this assumption, finding the object's convex hull amounts to

Phase 1. finding which of the convex curves of the object's boundary contain pieces of the convex hull's boundary

and

Phase 2. computing the A-and-B – type points for each of these convex curves.

Another main hypothesis made by the mentioned authors is that the basic geometric operations can be accurately performed for the considered convex curves that compose the object's boundary. These basic geometric operations are the computation of the intersection of such a curve and a line and tangents computation (from a point to a curve and common tangents of two curves). Supposing that the curves are algebraic, the above operations require (high degree) algebraic equations resolution; let us give an example illustrating the topological problems that can be thus risen.

We study what may happen when we want to compute the upper convex hull of the three curves, $\mathcal{C}_1$, $\mathcal{C}_2$ and $\mathcal{C}_3$, presented Figure 1. We suppose that $\mathcal{C}_2$ passes slightly above the common tangent of $\mathcal{C}_1$ and $\mathcal{C}_3$. Then the convex hull's boundary should contain a segment of the common tangent of $\mathcal{C}_1$ and $\mathcal{C}_2$, let us call it $\mathcal{S}_1$, a piece of $\mathcal{C}_2$ and a segment of the common tangent of $\mathcal{C}_2$ and $\mathcal{C}_3$, let us call it $\mathcal{S}_2$. Let A be the extremity of $\mathcal{S}_1$ lying on $\mathcal{C}_2$ and B the extremity of $\mathcal{S}_2$ lying on $\mathcal{C}_2$. The numerical method used to compute the points A and B will not give exactly these two points, but some approximations of them, let us call them A′ and B′. If A and B are very close one to another, the order of the points A′ and B′ on $\mathcal{C}_2$ might not be the same as the one of A and B, as shown in the figure.

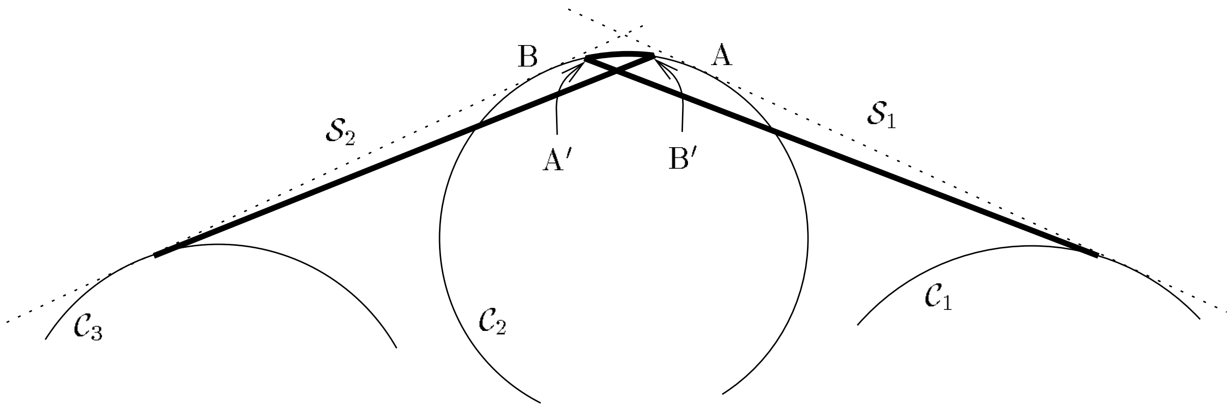


Figure 1. *Incoherence of the convex hull's boundary generated by inaccurate numerical solving of algebraic equations*

If we already know that the convex hull's boundary must contain a piece of $\mathcal{C}_2$, we can detect the incoherent relative position of the points A′ and B′, because the bold line in the figure cannot be a part of this convex hull's boundary. Otherwise, we will deduce that $\mathcal{C}_2$ does not overlap the convex hull's boundary, conclusion leading to a wrong convex hull of the curves.

So, the imprecisions inherent to numerical resolution of algebraic equations can bring inaccurate results in both Phase 1 and Phase 2 above. The errors occurring in Phase 2 are predictable and not very upsetting, but those appearing in Phase 1 completely change the description of the convex hull and therefore are critical. We can also see that, generally, degeneracy cases (when three or more of the convex curves composing the object's boundary have a common tangent) can be detected only to a given precision.

As a conclusion, we can say that computation based on numerical solving of algebraic equations provides a very good approximation of the convex hull, in terms of distance, but on the other hand this solution might not be satisfactory enough from a topological point of view. That is the reason why we propose in this paper a different approach, concentrated on the study of the Phase 1 described above and banishing the algebraic equations solving. We also remark that if the correct solution of the Phase 1 is known, the object's convex hull is completely and correctly defined, even if this definition remains implicit.

2.2. Topological convex hull

Let us start this section with a few notation and a definition that will be used all this paper long.

If A, B are two points, [AB] denotes the closed segment defined by A and B, and if A, B are two points of a simple non-closed curve c, $c\big|_{(\mathrm{A},\mathrm{B})}$ denotes the part of c delimited by A and B. If $\mathcal{A}$ is a set of points, $\mathcal{H}(\mathcal{A})$ will denote the convex hull of $\mathcal{A}$; $\mathrm{Int}(\mathcal{A})$ denotes the interior of the set $\mathcal{A}$ and $\overline{\mathcal{A}}$ denotes the closure of this set.

If Δ is a line and $\mathcal{A}$ is a set such that $\mathrm{Int}(\mathcal{H}(\mathcal{A})) \neq \emptyset$ and $\Delta \cap \mathrm{Int}(\mathcal{H}(\mathcal{A})) = \emptyset$, $\mathrm{DP}(\Delta, \mathcal{A})$ will denote the opened half-plane delimited by Δ and containing $\mathrm{Int}(\mathcal{H}(\mathcal{A}))$.

Definition 2.1. – *We say that a (Jordan) planar curve $\mathcal{C}$ is convex if it completely lies on the boundary of its convex hull.*

With the remark that, in the following, indices will always be considered modulo the size of the subscripts set they belong to, we can now define the topological convex hull of a curve-bounded object.

Let $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_n$ be convex planar (Jordan) curves such that $\mathcal{C} = \cup_{i=1}^{n} \mathcal{C}_i$ is a simple closed curve. We suppose that $\mathcal{C}_i,\ i = 1, \ldots, n$ are ordered counterclockwise on $\mathcal{C}$. For $i \neq j \in \{1, \ldots, n\}$, let Δ_{ij} denote the supporting line of the curves $\mathcal{C}_i$ and $\mathcal{C}_j$ which leaves both this curves to the left (i.e., such that the left half-plane defined by the line Δ_{ij} oriented from $\Delta_{ij} \cap \mathcal{C}_i$ to $\Delta_{ij} \cap \mathcal{C}_j$ includes the two curves).

Definition 2.2. – *We will say that a set $\mathcal{I} = \{i_1, \ldots, i_p\} \subset \{1, \ldots, n\}$ gives (one of) the topological structure(s) of the convex hull of the curve $\mathcal{C}$ iff the boundary $\mathcal{H}$ of $\mathcal{H}(\mathcal{C})$ can be written as a sequence of pieces of the curves $\mathcal{C}_{i_k}$, for $i_k \in \mathcal{I}$, and of segments relying them:*

$$\mathcal{H} = \bigcup_{k=1}^{p} \left(\mathcal{C}_{i_k}\big|_{(\mathrm{R}_{i_k}, \mathrm{R}'_{i_k})} \cup [\mathrm{R}'_{i_k} \mathrm{R}_{i_{k+1}}] \right)$$

where $\mathrm{R}_{i_k} \in \Delta_{i_{k-1}i_k} \cap \mathcal{C}_{i_k}$, $\mathrm{R}'_{i_k} \in \Delta_{i_k i_{k+1}} \cap \mathcal{C}_{i_k}$.

If the set $\mathcal{I}$ is considered such that $i_1 = \min_{k=1}^{p} i_k$, the simplicity of the curve $\mathcal{C}$ will imply $i_1 < i_2 < \ldots < i_p$. A set $\mathcal{I}$ with the given properties does obviously always exist, but in some cases it is not unique (so there is not a unique topological structure of the convex hull). For the curve $\mathcal{C}$ presented Figure 2, there are four admissible sets $\mathcal{I}$: $\mathcal{I}_1 = \{1,2,3,4,5\}$, $\mathcal{I}_2 = \{1,2,4,5\}$, $\mathcal{I}_3 = \{1,3,4,5\}$ and $\mathcal{I}_4 = \{1,4,5\}$.

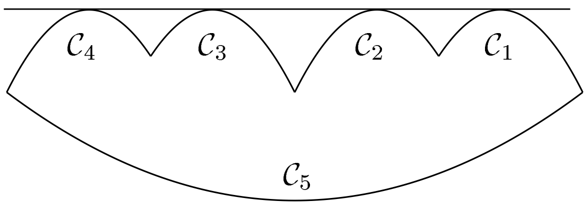


Figure 2. *In this case, the topological structure of the convex hull is not unique*

A simple composite curve $\mathcal{C}$ satisfying the properties above fulfills the conditions asked by (Dobkin and Souvaine, 1990) and (Schäffer and Van Wyk, 1987); if the curves $\mathcal{C}_i$ are algebraic, $\mathcal{C}$ is the object studied by (Bajaj and Kim, 1991). Finding the set $\mathcal{I}$ giving the topological structure of its convex hull solves the Phase 1 given in the previous subsection. As we mentioned, this gives a complete description of the convex hull of $\mathcal{C}$. This description gives less informations on the curve $\mathcal{H}$ bounding $\mathcal{H}(\mathcal{C})$ then the ones proposed in the cited papers, because the points R_{i_k} and R'_{i_k}, for $k \in \{1,\ldots,p\}$, are implicit. On the other hand, the method we propose avoids algebraic equations. After determining the topological structure of the convex hull, the computation of the important points (Phase 2 in Subsection 2.1) can be done "securely" from a topological point of view.

3. Definition and properties of the studied object

The convex hull problem for curved objects can be addressed in quite a general framework, as the authors do in (Bajaj and Kim, 1991) and especially in (Dobkin and Souvaine, 1990) and (Schäffer and Van Wyk, 1987). The resulting algorithms are theoretical; we want to propose a method suitable to immediate implementation. This is one of the reasons why we restrict our study to some particular piecewise convex parametric curves, not without remarking that these objects represent quite a large class of planar curved objects.

3.1. Point-controled parametric curves

Let $\mathcal{C}$ be a parametric curve controled by the set of points $\mathrm{P}_0, \mathrm{P}_1, \ldots, \mathrm{P}_m$. The object we introduce is a union of such parametric curves fulfilling certain geometric properties. Roughly, the required properties are the following:

- P_0 and P_m are the extremities of the curve $\mathcal{C}$;
- the line $\mathrm{P}_0\mathrm{P}_1$ is tangent to $\mathcal{C}$ in P_0 and the line $\mathrm{P}_{m-1}\mathrm{P}_m$ is tangent to $\mathcal{C}$ in P_m;
- the curve is included in the convex hull of its control points;
- the curve and its control polygon satisfy the variation diminishing property (as classically understood in CAGD, see Farin(1997));
- there exists a subdivision algorithm through which the control polygon converges to

the curve.

These five conditions are satisfied by many types of parametric curves: Bézier and rational Bézier curves, B-splines and NURBS, under certain conditions, etc. For the implementation of the method we have chosen, as a first step, the Bézier curves. That is why, for the clearness of the presentation, we will from now on restrict the discussion to Bézier curves, even if it can be proven that the results are valid for any type of parametric curve fulfilling the above conditions.

The subdivision algorithm mentioned in the last condition will be the de Casteljau subdivision algorithm, that we have chosen of parameter $1/2$. The reasons of this choice are two: firstly, the parameter $1/2$ is the only one assuring quadratic convergence of the control polygon to the Bézier curve, as shown by Calcoen(1996). Secondly, the computations are simpler in this case. On the other hand, if we do not want to use fixed subdivision parameter, we cannot adapt this parameter properly without treating algebraic equations, and as we said in the introduction we interdict ourselves addressing algebraic equations.

For the same reason, we ask all the Bézier curves we deal with to be *completely convex*:

Definition 3.1. – *We say that a Bézier curve is completely convex if its control polygon is simple and convex.*

The variation diminishing property implies the convexity of a completely convex curve. So, we do not need to compute inflection points to obtain a decomposition of our object in piecewise convex curves; this decomposition is necessary, while the inflection points computation requires algebraic equations resolution.

3.2. The polycurve and its control polygon

Let $\mathbb{B}$ be the set of all the simple closed planar curves that can be written as a (finite) union of completely convex Bézier curves. A curve $\mathcal{B} \in \mathbb{B}$ will then be $\mathcal{B} = \cup_{i=1}^{n} \mathcal{B}_i$, where the control polygon of the completely convex Bézier curve $\mathcal{B}_i$ is $\mathcal{P}_i = \mathrm{P}_0^{(i)}\mathrm{P}_1^{(i)}\ldots\mathrm{P}_{m_i}^{(i)}$ for $i \in \{1, \ldots, n\}$. On $\mathbb{B}$, we introduce the relation "obtained by de Casteljau subdivision/fusion of the composing Bézier curves", denoted "$\sim$", and it is easy to see that this is an equivalence relation.

Definition 3.2. – *We will call polycurve either an equivalence class of the relation "obtained by de Casteljau subdivision/fusion of the composing Bézier curves" over $\mathbb{B}$ (an element of $\mathbb{B}/\sim$) or an element of $\mathbb{B}$.*

All the members of a polycurve (equivalence class) have the same geometric support and differ one from another by parametrization, that is why an element of $\mathbb{B}$ will also be called polycurve. Of course, we consider all polycurves (elements of $\mathbb{B}$) which are equivalent through $\sim$ to be equal. By default, the name polycurve will apply from now on to the objects of the set $\mathbb{B}$.

Definition 3.3. – *The control polygon of a polycurve is the union of the control polygons of the Bézier curves composing the polycurve.*

The control polygon of a polycurve is obviously not unique, two different members of a class of $\mathbb{B}/\sim$ represent the same polycurve and have different control polygons.

3.3. Notations

A polycurve $\mathcal{B} \in \mathbb{B}$ is written $\mathcal{B} = \cup_{i=1}^{n} \mathcal{B}_i$, where $\mathcal{B}_i$ are completely convex Bézier curves with control polygon $\mathcal{P}_i = \mathrm{P}_0^{(i)} \mathrm{P}_1^{(i)} \ldots \mathrm{P}_{m_i}^{(i)}$, for all $i \in \{1, \ldots, n\}$. The polycurve's control polygon will then be $\mathcal{P} = \cup_{i=1}^{n} \mathcal{P}_i$. The number n of Bézier curves composing the polycurve will be called the *size* of the polycurve, $|\mathcal{B}|$.

Let $\mathcal{B}$ be a polycurve and n_0 be the minimal size of the members of the equivalence class of $\mathcal{B}$:

$$n_0 = \min\big\{ |\mathcal{B}'| \mid \mathcal{B}' \sim \mathcal{B} \big\}$$

There exists a unique polycurve $\mathcal{B}_{ini}$ such that $\mathcal{B}_{ini} \sim \mathcal{B}$ and $|\mathcal{B}_{ini}| = n_0$ and it will be called the *initial* polycurve of the given equivalence class. Any member of the class that we deal with will be the *current* polycurve at a given time (the initial polycurve is then the current polycurve at the very beginning, when no de Casteljau subdivision has been applied). All the polycurves will be denoted by $\mathcal{B} = \cup_{i=1}^{n} \mathcal{B}_i$, keeping in mind that n varies.

Let $\mathcal{B} = \cup_{i=1}^{n} \mathcal{B}_i$ be the current polycurve and $\mathcal{B}_{ini} = \cup_{i=0}^{n_0} \mathcal{B}_i^{ini}$ the initial polycurve of the same equivalence class. There exists $\mathcal{S} = \{s_1 = 1 < s_2 < \ldots < s_{n_0} \leq n\}$ such that $\mathcal{B}_k^{ini} = \cup_{i=s_k}^{s_{k+1}-1} \mathcal{B}_i$ for all $k \in \{1, \ldots, n_0\}$. The Bézier curves $\mathcal{B}_i$ for $i \in \{s_k, \ldots, s_{k+1}-1\}$ are obtained by successive de Casteljau subdivisions of the Bézier curve $\mathcal{B}_k^{ini}$. In the following, the union of all the Bézier curves coming from the same initial one as a given curve $\mathcal{B}_i$ from the current composition of the polycurve will be denoted by $\widetilde{\mathcal{B}}_i$; so, using the above notations,

$$\widetilde{\mathcal{B}}_{s_k} = \widetilde{\mathcal{B}}_{s_k+1} = \ldots = \widetilde{\mathcal{B}}_{s_{k+1}-1} = \bigcup_{i=s_k}^{s_{k+1}-1} \mathcal{B}_i \qquad \text{for all } k \in \{1, \ldots, n_0\}$$

We will use similar notations for the control polygons of the Bézier curves composing the polycurve. Thus, $\widetilde{\mathcal{P}}_{s_k} = \ldots = \widetilde{\mathcal{P}}_{s_{k+1}-1} = \cup_{i=s_k}^{s_{k+1}-1} \mathcal{P}_i$, but we remark that $\mathcal{P}_k^{ini} \neq \widetilde{\mathcal{P}}_{s_k}$.

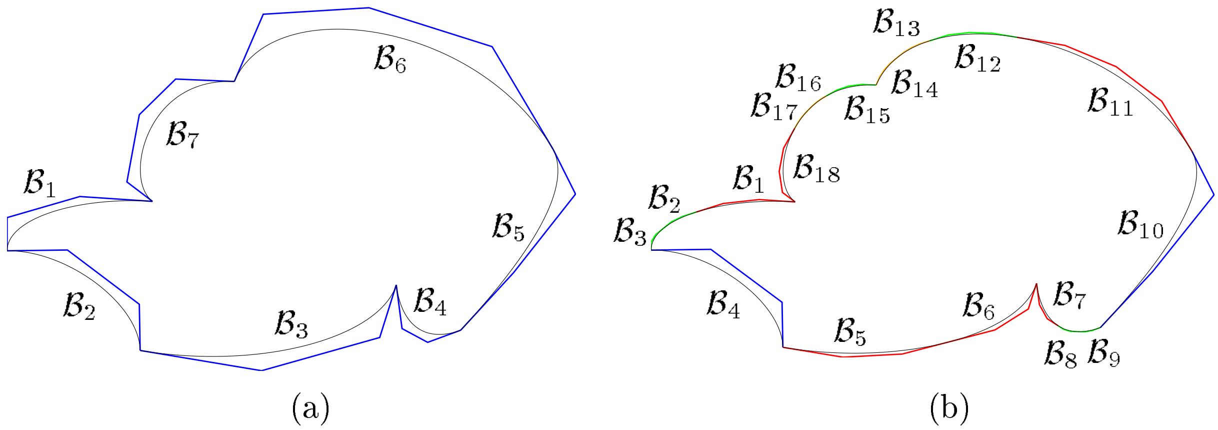


Figure 3. (a) *The initial polycurve;* $n_0 = 7$. (b) *The current polycurve;* $n = 18$. *For example,* $\mathcal{B}_6^{ini} = \widetilde{\mathcal{B}}_{11} = \widetilde{\mathcal{B}}_{12} = \widetilde{\mathcal{B}}_{13} = \widetilde{\mathcal{B}}_{14} = \mathcal{B}_{11} \cup \mathcal{B}_{12} \cup \mathcal{B}_{13} \cup \mathcal{B}_{14}$*, but* $\mathcal{P}_6^{ini} \neq \widetilde{\mathcal{P}}_{11} = \mathcal{P}_{11} \cup \mathcal{P}_{12} \cup \mathcal{P}_{13} \cup \mathcal{P}_{14}$

3.4. Properties of the control polygon

The idea of the method we propose for the computation of a polycurve's convex hull is to work only with its control polygon and never directly with the polycurve. So, we have to establish the conditions under which there is enough information in the control polygon so that the topological structure of the convex hull of the polycurve can be computed. Actually, a step of the computation consists in computing the convex hull of the control polygon, then verifying if the mentioned conditions are fulfilled, and if they are not, subdividing some or all of the Bézier curves composing the polycurve.

This leads us to interest in the properties of the control polygon. Computing the convex hull of a polygon is a widely studied problem. If the polygon is simple, (Graham and Yao 1983) proved that the computation can be done in linear time. If it is not, the complexity grows to $\mathcal{O}(n \log n)$, n being the polygon's size. So, we should examine the existence, for a given polycurve, of a simple control polygon, eventually obtained by de Casteljau subdivision from the initial one. The examples in Figure 4 show us that such a simple control polygon does not always exist.

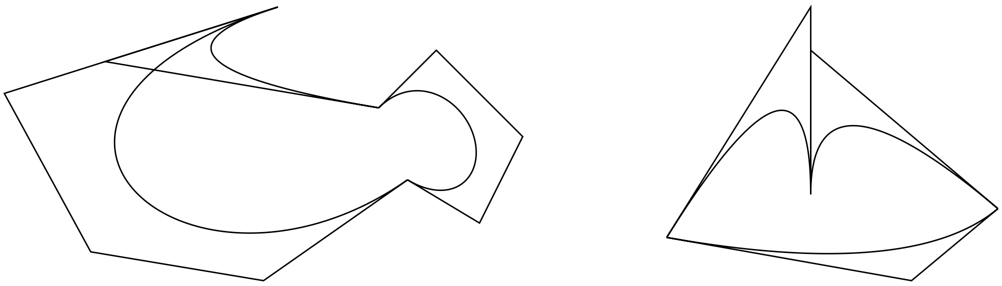


Figure 4. *For none of the two polycurves a simple control polygon can be obtained*

Definition 3.4. – *Let $\mathcal{P} = \mathrm{P}_1\mathrm{P}_2 \ldots \mathrm{P}_m$ be a polygon. We say that $\mathcal{P}$ is pseudo-simple iff for all $i \neq j \in \{1, \ldots, n\}$ such that $[\mathrm{P}_i\mathrm{P}_{i+1}] \cap [\mathrm{P}_j\mathrm{P}_{j+1}] \neq \emptyset$ either the points $\mathrm{P}_i, \mathrm{P}_{i+1}, \ldots, \mathrm{P}_j$ are colinear, or the points $\mathrm{P}_j, \mathrm{P}_{j+1}, \ldots, \mathrm{P}_i$ are colinear.*

This property is obviously weaker then simplicity, any simple polygon is pseudo-simple, but the converse is not true. Yet, pseudo-simple polygons share with simple polygons a very important property: the list of vertices defining the convex hull is an ordered list. More, the Graham-Yao algorithm for computing the convex hull of a simple polygon needs only a minor modification to work, in linear time, on pseudo-simple polygons. So, the existence of a pseudo-simple control polygon of the given polycurve seems sufficing. In fact, this is not true, because we need not only to obtain a pseudo-simple control polygon, but also to be sure that subdivisions applied to (some of) the Bézier curves composing the polycurve will not generate a loss of this property. It is easy to remark that the first example of Figure 4 does not fulfill this condition.

Definition 3.5. – *Let $\mathcal{B} = \cup_{i=1}^{n} \mathcal{B}_i$ be a polycurve, where the control polygon of the Bézier curve $\mathcal{B}_i$ is $\mathcal{P}_i = \mathrm{P}_0^{(i)} \ldots \mathrm{P}_{m_i}^{(i)}$, $i \in \{1, \ldots, n\}$. We say that $\mathcal{B}$ is a polycurve with interior tangency (shortly w.i.t.) iff there exists $i_0 \in \{1, \ldots, n\}$ such that:*

(3.5.i) *the points $\mathrm{P}_{m_{i_0}-1}^{(i_0)}$, $\mathrm{P}_{m_{i_0}}^{(i_0)} = \mathrm{P}_0^{(i_0+1)}$ and $\mathrm{P}_1^{(i_0+1)}$ are collinear;*

(3.5.ii) *$\mathrm{P}_{m_{i_0}-1}^{(i_0)}$ and $\mathrm{P}_1^{(i_0+1)}$ are not separated by $\mathrm{P}_{m_{i_0}}^{(i_0)}$;*

(3.5.iii) *the curves $\mathcal{B}_{i_0}$ and $\mathcal{B}_{i_0+1}$ are in the same closed half-plane defined by the line $\mathrm{P}_0^{(i_0+1)}\mathrm{P}_1^{(i_0+1)}$.*

The first polycurve in Figure 4 is an example of polycurve w.i.t.

Definition 3.6. – *We say that a polycurve $\mathcal{B}$ is with no interior tangency (shortly w.n.i.t.) iff it is not w.i.t.*

Lemma 3.7. – *Let $\mathcal{B}$ be a polycurve w.n.i.t. Then it exists $\gamma_0 > 0$ such that if the polycurve is written $\mathcal{B} = \cup_{i=1}^{n} \mathcal{B}_i$ and $\mathcal{P}_i$, the control polygon of the curve $\mathcal{B}_i$, satisfies $\delta^{\mathrm{H}}(\mathcal{B}_i, \mathcal{P}_i) < \gamma_0$ for all $i \in \{1, \dots, n\}$, then the polycurve's control polygon, $\mathcal{P} = \cup_{i=1}^{n} \mathcal{P}_i$, is pseudo-simple ($\delta^{\mathrm{H}}(\mathcal{A}, \mathcal{B})$ denotes the Hausdorff distance between the sets $\mathcal{A}$ and $\mathcal{B}$).*

The proof of this lemma is a technical case discussion without great theoretical interest, so we will not present it here. We remark only that γ_0 is a measure of the "narrowness" of the given polycurve. Further on, we will always consider that the control polygon of the polycurve is pseudo-simple and it remains so when subdivisions are applied to polycurve's composing Bézier curves.

4. Topological relevance of the control polygon's convex hull to the polycurve's one

4.1. Specifying the notion of topological structure

As we have announced, our purpose is to find the topological structure of a polycurve's convex hull dealing only with its control polygon via the de Casteljau subdivision of the Bézier curves composing the polycurve. But the topological structure of the convex hull intrinsically depends on the writting of the polycurve as union of Bézier curves. So, two members of a class of $\mathbb{B}/\sim$ have the same convex hull, while the topological structures of this convex hull respectively associated to the two members are different. For example, for the polycurve presented Figure 5(a), the set $\mathcal{I}$ of Definition 2.2 is $\mathcal{I} = \{1, 2, 3, 5\}$ and for the one presented Figure 5(b) $\mathcal{I} = \{1, 2, 3, 4, 5, 6, 7, 10, 11\}$; the two polycurves are obviously equivalent by $\sim$.

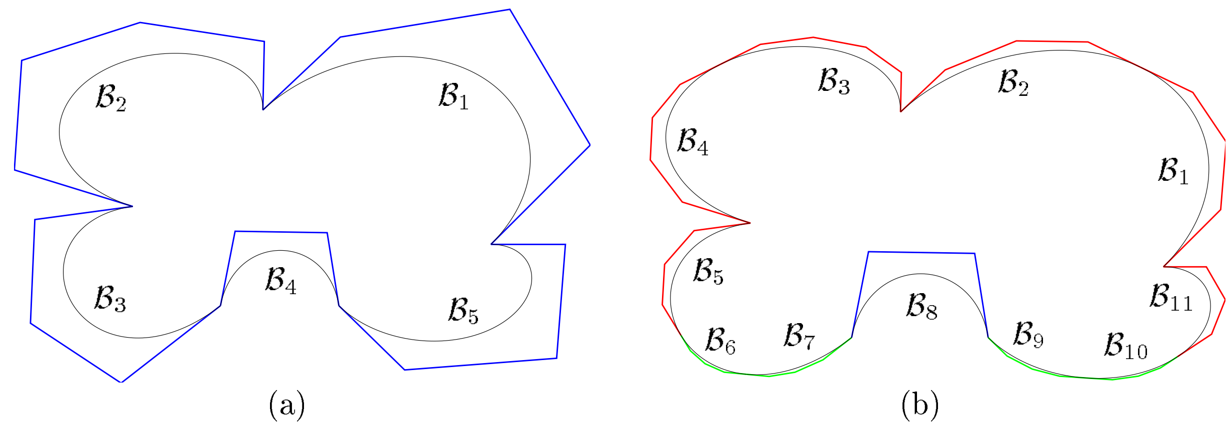


Figure 5. (a) $\mathcal{I} = \{1, 2, 3, 5\}$; (b) $\mathcal{I} = \{1, 2, 3, 4, 5, 6, 7, 10, 11\}$

Thus, talking about the topological structure of a polycurve (seen as equivalence class) does not make much sense if we do not fix the member of the class it refers to. It is natural to chose the initial polycurve of the class with this aim, because firstly it has minimal size and secondly its composing Bézier curves can be recovered from the current polycurve's composing curves by union (in the previous figure, for example, $\mathcal{B}_3$ of figure (a) is $\widetilde{\mathcal{B}}_5 = \mathcal{B}_5 \cup \mathcal{B}_6 \cup \mathcal{B}_7$ of figure (b)).

For clearness' sake, we are going to reformulate the definition of the topological structure of the convex hull in our particular framework, taking into account the remarks above. With this end in view, let us remind the definition of the set of indices $\mathcal{S}$ introduced Subsection 3.3. This set is associated to a given polycurve $\mathcal{B} = \cup_{i=1}^{n} \mathcal{B}_i$ and has the size of the initial polycurve of its class, supposed n_0. We have then $\mathcal{S} = \{s_1 < s_2 < \ldots < s_{n_0}\} \subset \{1, \ldots, n\}$, where

- $\widetilde{\mathcal{B}}_{s_k} \neq \widetilde{\mathcal{B}}_{s_h}$ if $k \neq h$;
- if $i \in \{1, \ldots, n\}$ is such that $\widetilde{\mathcal{B}}_i = \widetilde{\mathcal{B}}_{s_k}$, then $s_k \leq i$.

The first of these two conditions and the equality of the cardinal of $\mathcal{S}$ to the size of the initial polycurve imply $\mathcal{B} = \cup_{k=1}^{n_0} \widetilde{\mathcal{B}}_{s_k}$. The topological structure of the convex hull of the polycurve $\mathcal{B}$ is then given by the following definition:

Definition 4.1. – *We will say that a set $\mathcal{I} = \{i_1 < \ldots < i_p\} \subset \{1, \ldots, n_0\}$ gives the topological structure of $\mathcal{H}(\mathcal{B})$ iff*

$$\mathcal{H} = \bigcup_{k=1}^{p} \left(\widetilde{\mathcal{B}}_{s_{i_k}} \Big|_{(\mathrm{R}_{s_{i_k}}, \mathrm{R}'_{s_{i_k}})} \cup [\mathrm{R}'_{s_{i_k}} \mathrm{R}_{s_{i_{k+1}}}] \right)$$

where $\mathrm{R}_{s_{i_k}} \in \widetilde{\Delta}_{s_{i_{k-1}} s_{i_k}} \cap \widetilde{\mathcal{B}}_{s_{i_k}}$ *and* $\mathrm{R}'_{s_{i_k}} \in \widetilde{\Delta}_{s_{i_k} s_{i_{k+1}}} \cap \widetilde{\mathcal{B}}_{s_{i_k}}$ *(the line $\widetilde{\Delta}_{ij}$ is the common tangent of $\widetilde{\mathcal{B}}_i$ and $\widetilde{\mathcal{B}}_j$ leaving these curves to the left).*

We must remark that the set $\mathcal{I}$ is an invariant of a given polycurve, i.e. it does not depend on the member of the class. Indeed, this set is determined by the composition of the initial polycurve and not by the set $\mathcal{S}$, which changes at de Casteljau subdivision. To give an example, let us look again at Figure 5. In figure (a), the polycurve is in its initial form and we have $\mathcal{S} = \{1, \ldots, 5\}$ and $\mathcal{I} = \{1, 2, 3, 5\}$. In figure (b) we have the current form of the polycurve and $\mathcal{S} = \{1, 3, 5, 8, 9\}$. So $s_1 = 1, s_2 = 3, s_3 = 5, s_4 = 8, s_5 = 9$. The set $\mathcal{I}$ must identify the Bézier curves $\widetilde{\mathcal{B}}_1, \widetilde{\mathcal{B}}_3, \widetilde{\mathcal{B}}_5$ and $\widetilde{\mathcal{B}}_9$. Searching the correspondence in the set $\mathcal{S}$, we obtain $\mathcal{I} = \{1, 2, 3, 5\}$, the same set as for the initial form of the polycurve.

4.2. Formalizing the notion of topological structure

With a wording almost identical to the one of Definition 4.1, we can define the topological structure of the control polygon's convex hull, by just replacing the '$\mathcal{B}$'-s by '$\mathcal{P}$'-s. Our aim is then to obtain, via the de Casteljau subdivision, a control polygon such that its convex hull has the same topological structure as the polycurve's convex hull. Naturally, the first question to ask is if such a control polygon exists. But the notion of topological structure of the convex hull is such an empirical one that this question cannot be answered at this time. That is the reason why we give the following definition:

Definition 4.2. – *Let $\mathcal{B} = \cup_{i=1}^{n} \mathcal{B}_i$ be a polycurve and $\mathcal{P} = \cup_{i=1}^{n} \mathcal{P}_i$ its control polygon. Let $\mathcal{H}$ be the boundary of $\mathcal{H}(\mathcal{B})$ and $\mathcal{Q}$ the boundary of $\mathcal{H}(\mathcal{P})$. We say that $\mathcal{H}(\mathcal{P})$ is topologically relevant to $\mathcal{H}(\mathcal{B})$ iff*

(4.2.i) $\widetilde{\mathcal{P}}_i \cap \mathcal{Q} \neq \emptyset \Longrightarrow \widetilde{\mathcal{B}}_i \cap \mathcal{H} \neq \emptyset$;

(4.2.ii) $\widetilde{\mathcal{P}}_i \subset \mathrm{Int}\left(\mathcal{H}\Big(\bigcup_{\widetilde{\mathcal{B}}_j \neq \widetilde{\mathcal{B}}_i} \mathcal{P}_j \Big) \right) \Longrightarrow \widetilde{\mathcal{B}}_i \subset \mathcal{H}\Big(\bigcup_{\widetilde{\mathcal{B}}_j \neq \widetilde{\mathcal{B}}_i} \mathcal{B}_j \Big)$.

The two conditions of this definition mean, roughly speaking, that if the control polygon of one of the Bézier curves of the initial polycurve's composition is significant for the convex hull of the polycurve's control polygon, then the corresponding Bézier curve will be significant for the polycurve's convex hull, and conversely. The following lemma states that this is really a formalization of the identity of topological structures:

Lemma 4.3. – *The convex hull of the control polygon, $\mathcal{H}(\mathcal{P})$, is topologically relevant to the one of the polycurve, $\mathcal{H}(\mathcal{B})$, iff they have the same topological structure.*

Proof – Let us have a look at the polygon $\mathcal{Q}$, the boundary of $\mathcal{H}(\mathcal{P})$. Using the set $\mathcal{S}$, the control polygon $\mathcal{P}$ can be written $\mathcal{P} = \cup_{k=1}^{n_0} \widetilde{\mathcal{P}}_{s_k}$, where

$$\widetilde{\mathcal{P}}_{s_k} = \bigcup_{i=s_k}^{s_{k+1}-1} \mathcal{P}_i = \mathrm{P}_0^{(s_k)}\mathrm{P}_1^{(s_k)} \ldots \mathrm{P}_{m_{s_k}}^{(s_k)} \mathrm{P}_0^{(s_k+1)} \ldots \mathrm{P}_{m_{s_k}}^{(s_k+1)} \ldots \mathrm{P}_0^{(s_{k+1}-1)} \ldots \mathrm{P}_{m_{s_k}}^{(s_{k+1}-1)}$$

$$\text{for all } k \in \{1, \ldots, n_0\}$$

Then Definition 2.2 makes sense for the *topological structure of* $\mathcal{H}(\mathcal{P})$, and it is the one that we use.

We remark that all the polygons $\mathcal{P}_i$ for $i \in \{s_k, \ldots, s_{k+1} - 1\}$ have the same size, being obtained by subdivision from the same initial control polygon. The polygon $\widetilde{\mathcal{P}}_{s_k}$ has $\widetilde{m}_k = m_{s_k}(s_{k+1} - s_k)$ vertices; we rename them such that

$$\widetilde{\mathcal{P}}_{s_k} = \widetilde{\mathrm{P}}_0^{(k)}\widetilde{\mathrm{P}}_1^{(k)} \ldots \widetilde{\mathrm{P}}_{\widetilde{m}_k}^{(k)} \qquad \text{for all } k \in \{1, \ldots, n_0\}$$

The fact that the Bézier curves composing the initial polycurve are completely convex implies that there exists a set $\Sigma = \{\sigma_1 < \ldots < \sigma_\pi\} \subset \{1, \ldots, n_0\}$ and for each $k \in \{1, \ldots, \pi\}$ there exist $j_{\sigma_k} \in \{0, \ldots, \widetilde{m}_{\sigma_k}\}$ and $\ell_{\sigma_k} \in \{0, \ldots, \widetilde{m}_{\sigma_k} - j_{\sigma_k}\}$ such that

$$\mathcal{Q} = \widetilde{\mathrm{P}}_{j_{\sigma_1}}^{(\sigma_1)}\widetilde{\mathrm{P}}_{j_{\sigma_1}+1}^{(\sigma_1)} \ldots \widetilde{\mathrm{P}}_{j_{\sigma_1}+\ell_{\sigma_1}}^{(\sigma_1)} \widetilde{\mathrm{P}}_{j_{\sigma_2}}^{(\sigma_2)} \ldots \widetilde{\mathrm{P}}_{j_{\sigma_2}+\ell_{\sigma_2}}^{(\sigma_2)} \ldots \widetilde{\mathrm{P}}_{j_{\sigma_\pi}}^{(\sigma_\pi)} \ldots \widetilde{\mathrm{P}}_{j_{\sigma_\pi}+\ell_{\sigma_\pi}}^{(\sigma_\pi)} \qquad (1)$$

The remark to be done is that the variant of the Graham-Yao algorithm that we use for computing the convex hull of $\mathcal{P}$ constructs the maximal size $\mathcal{Q}$. That means that if two or more vertices of $\mathcal{P}$ are collinear and on the boundary of $\mathcal{H}(\mathcal{P})$, they will all be vertices of $\mathcal{Q}$ if this does not imply a loss of coherence. For example, in the case presented Figure 6(a), the vertices $\widetilde{\mathrm{P}}_{j_\alpha}^{(\alpha)}$, $\widetilde{\mathrm{P}}_{j_\beta}^{(\beta)}$ and $\widetilde{\mathrm{P}}_{j_\gamma}^{(\gamma)}$ are all vertices of $\mathcal{Q}$, while in the case Figure 6(b) the vertex $\widetilde{\mathrm{P}}_1^{(\beta)}$ is not.

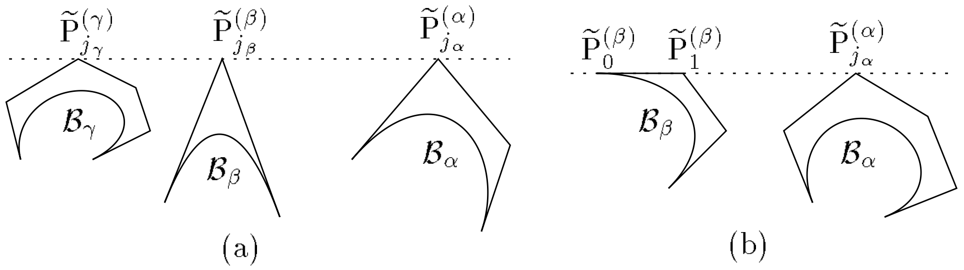


Figure 6. *Two configurations of collinear vertices of $\mathcal{P}$*

Using these notations, the lemma amounts to saying that $\mathcal{H}(\mathcal{P})$ is topologically relevant to $\mathcal{H}(\mathcal{B})$ iff the set Σ gives the topological structure of $\mathcal{H}(\mathcal{B})$, and the proof of this equivalence is straightforward.

$\square$

4.3. Existence of the searched control polygon

Using the results of the prevous subsections, we can prove that for a given polycurve there exists a member of the class such that the set Σ associated to its control polygon gives the topological structure of the polycurve's convex hull. More, any further subdivision applied to this member of the class will not affect this property. The existence of such a member of the class is stated by the following theorem:

Theorem 4.4. – *Let $\mathcal{B}$ be a polycurve w.n.i.t. (Def. 3.6). Then it exists $\varepsilon_0 > 0$ such that if the polycurve is written $\mathcal{B} = \cup_{i=1}^{n} \mathcal{B}_i$ and $\mathcal{P}_i$, the control polygon of the curve $\mathcal{B}_i$, satisfies $\delta^{\mathrm{H}}(\mathcal{B}_i, \mathcal{P}_i) < \varepsilon_0$ for all $i \in \{1, \ldots, n\}$, then the control polygon's convex hull is topologically relevant to the one of the polycurve.*

Proof – Let $\mathcal{I} = \{i_1, \ldots, i_p\}$ be a set giving the topological structure of $\mathcal{H}(\mathcal{B})$. $\mathcal{H}$, the boundary of $\mathcal{H}(\mathcal{B})$, can then be written

$$\mathcal{H} = \bigcup_{k=1}^{p} \left(\widetilde{\mathcal{B}}_{s_{i_k}}\Big|_{(\mathrm{R}_{s_{i_k}}, \mathrm{R}'_{s_{i_k}})} \cup [\mathrm{R}'_{s_{i_k}} \mathrm{R}_{s_{i_{k+1}}}] \right)$$

Similarly, let $\Sigma = \{\sigma_1, \ldots, \sigma_\pi\}$ give the topological structure of the polygon $\mathcal{Q}$, boundary of $\mathcal{H}(\mathcal{P})$. The expression of this polygon is then given by Formula (1). We define

$$\widetilde{\varepsilon}_0 = \min_{1 \le k \le p} \left\{ \min_{\substack{\widetilde{\mathcal{B}}_i \neq \widetilde{\mathcal{B}}_{s_{i_k}}, \widetilde{\mathcal{B}}_{s_{i_{k+1}}} \\ \mathrm{d}(\mathcal{B}_i, \widetilde{\Delta}_{s_{i_k} s_{i_{k+1}}}) > 0}} \mathrm{d}(\mathcal{B}_i, \widetilde{\Delta}_{s_{i_k} s_{i_{k+1}}}) \right\}$$

We remark that the subscript of the second min avoids $\widetilde{\varepsilon}_0$ vanishing when three or more Bézier curves in the composition of the polycurve have a common tangent. If we consider $\mathcal{B}^{ini} = \cup_{i=1}^{n_0} \mathcal{B}_i^{ini}$ to be the initial polycurve of the class, $s_{i_k} = i_k$ for all $k \in \{1, \ldots, p\}$ and $\widetilde{\varepsilon}_0$ has the simpler form

$$\widetilde{\varepsilon}_0 = \min_{1 \le k \le p} \left\{ \min_{\substack{i \neq i_k, i_{k+1} \\ \mathrm{d}(\mathcal{B}_i^{ini}, \Delta_{i_k i_{k+1}}) > 0}} \mathrm{d}(\mathcal{B}_i^{ini}, \Delta_{i_k i_{k+1}}) \right\}$$

For every $i \in \mathcal{S}$, we define

$$\overline{\varepsilon}_i = \sup_{\mathrm{P} \in \widetilde{\mathcal{B}}_i} \mathrm{d}\big(\mathrm{P}, \mathcal{H}(\mathcal{B} \setminus \widetilde{\mathcal{B}}_i)\big) \qquad \text{and} \qquad \overline{\varepsilon}_0 = \min_{\substack{i \in \mathcal{S} \\ \overline{\varepsilon}_i \neq 0}} \overline{\varepsilon}_i$$

The ε_0 we are looking for will then be

$$\varepsilon_0 = \min \left\{ \gamma_0, \frac{\widetilde{\varepsilon}_0}{2}, \frac{\overline{\varepsilon}_0}{2} \right\},$$

where γ_0 is the value given by Lemma 3.7. We prove that the two conditions of the Definition 4.2 of the topological relevance are fulfilled if $\delta^{\mathrm{H}}(\mathcal{B}_i, \mathcal{P}_i) < \varepsilon_0$ for all $i \in \{1, \ldots, n\}$.

Let $i \in \mathcal{S}$ be such that $\widetilde{\mathcal{P}}_i \cap \mathcal{Q} \neq \emptyset$. Then it exists $\kappa \in \{1, \ldots, \pi\}$ such that $i = s_{\sigma_\kappa}$. Let us suppose that $\widetilde{\mathcal{B}}_i \cap \mathcal{H} = \emptyset$, which means $\widetilde{\mathcal{B}}_i \subset \mathrm{Int}(\mathcal{H}(\mathcal{B}))$. It will then exists $k \in \{1, \ldots, p\}$ such that we have the situation presented Figure 7.

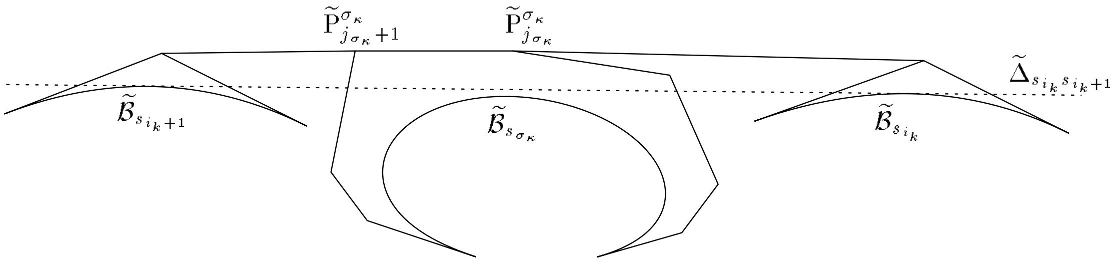


Figure 7. $\widetilde{\mathcal{P}}_i \cap \mathcal{Q} \neq \emptyset$, $\widetilde{\mathcal{B}}_i \cap \mathcal{H} = \emptyset$

Then we can write

$$\widetilde{\varepsilon}_0 \leq \mathrm{d}(\widetilde{\mathcal{B}}_{s_{\sigma_\kappa}}, \widetilde{\Delta}_{s_{i_k}s_{i_{k+1}}}) \leq \mathrm{d}(\widetilde{\mathrm{P}}^{(\sigma_\kappa)}_{j_{\sigma_\kappa}}, \widetilde{\mathcal{B}}_{s_{\sigma_\kappa}}) \leq \delta^{\mathrm{H}}(\widetilde{\mathcal{B}}_{s_{\sigma_\kappa}}, \widetilde{\mathcal{P}}_{s_{\sigma_\kappa}}) = \max_{s_{\sigma_\kappa} \leq i < s_{\sigma_\kappa+1}} \delta^{\mathrm{H}}(\mathcal{B}_i, \mathcal{P}_i) < \varepsilon_0$$

contradiction. So, the supposition $\widetilde{\mathcal{B}}_i \cap \mathcal{H} = \emptyset$ is false and the first property of the topological relevance, (4.1.i), is proven.

For the second one, (4.1.ii), let $i \in \mathcal{S}$ be such that $\widetilde{\mathcal{P}}_i \subset \mathrm{Int}\big(\mathcal{H}(\cup_{j\in\mathcal{S}\setminus\{i\}}\widetilde{\mathcal{P}}_j)\big)$ and let us suppose that $\widetilde{\mathcal{B}}_i \not\subset \mathcal{H}(\cup_{j\in\mathcal{S}\setminus\{i\}}\widetilde{\mathcal{B}}_j)$. Then it exists $\kappa \in \{1, \ldots, \pi\}$ such that we have the configuration of Figure 8.

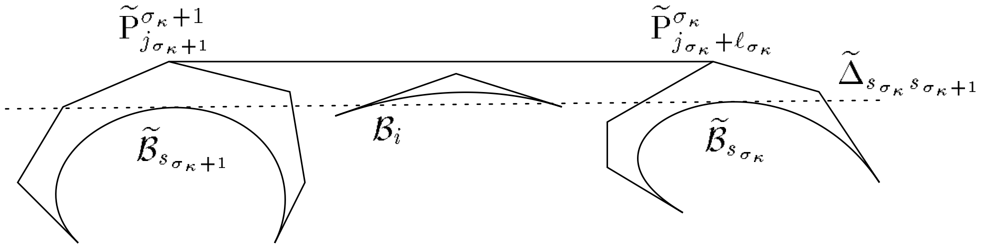


Figure 8. $\widetilde{\mathcal{P}}_i \subset \mathrm{Int}\big(\mathcal{H}(\cup_{\widetilde{\mathcal{B}}_j\neq\widetilde{\mathcal{B}}_i}\widetilde{\mathcal{P}}_j)\big)$, $\widetilde{\mathcal{B}}_i \not\subset \mathcal{H}(\cup_{\widetilde{\mathcal{B}}_j\neq\widetilde{\mathcal{B}}_i}\widetilde{\mathcal{B}}_j)$

We note $\widehat{\mathcal{B}}_i = \widetilde{\mathcal{B}}_i \setminus \mathrm{DP}\big(\widetilde{\Delta}_{s_{\sigma_\kappa}s_{\sigma_\kappa+1}}, \widetilde{\mathcal{B}}_{s_{\sigma_\kappa}}\big)$ and we can write

$$\begin{aligned}\overline{\varepsilon}_0 \leq \sup_{\mathrm{P}\in\widehat{\mathcal{B}}_i} \mathrm{d}(\mathrm{P}, \widetilde{\Delta}_{s_{\sigma_\kappa}s_{\sigma_\kappa+1}}) &< \max\big\{\mathrm{d}(\widetilde{\mathrm{P}}^{(\sigma_\kappa)}_{j_{\sigma_\kappa}+\ell_{\sigma_\kappa}}, \Delta_{s_{\sigma_\kappa}s_{\sigma_\kappa+1}}), \mathrm{d}(\mathrm{P}^{(\sigma_\kappa+1)}_{j_{\sigma_\kappa+1}}, \Delta_{s_{\sigma_\kappa}s_{\sigma_\kappa+1}})\big\} \\ &\leq \max\big\{\delta^{\mathrm{H}}(\widetilde{\mathcal{P}}_{s_{\sigma_\kappa}}, \widetilde{\mathcal{B}}_{s_{\sigma_\kappa}}), \delta^{\mathrm{H}}(\widetilde{\mathcal{P}}_{s_{\sigma_\kappa+1}}, \widetilde{\mathcal{B}}_{s_{\sigma_\kappa+1}})\big\} \leq \max_{1\leq i\leq n} \delta^{\mathrm{H}}(\mathcal{P}_i, \mathcal{B}_i) < \varepsilon_0\end{aligned}$$

contradiction. So, the supposition $\widetilde{\mathcal{B}}_i \not\subset \mathcal{H}(\cup_{\widetilde{\mathcal{B}}_j\neq\widetilde{\mathcal{B}}_i}\mathcal{B}_j)$ is also false and the proof of the theorem is achieved.

□

5. Correctness and convergence of the algorithm

The results of the previous section allow us to conclude that for any polycurve $\mathcal{B}$ (seen as equivalence class) there exists a member whose control polygon permits us to compute the topological structure of the polycurve's convex hull and such that any other member of the class obtained via de Casteljau subdivision from this one satisfies this property. But we do not have yet algebraic-equations-free criterion to find this member, and so the convex hull's topological structure. Indeed, the value ε_0 of Theorem 4.4 is obtained as a distance between a line and a Bézier curve and therefore is the solution of an algebraic equation.

In this section, we establish criteria of topological relevance resting uniquely on the control polygon. Thus, if the control polygon of a polycurve satisfies these criteria, we can deduce that the topological structure of its convex hull is the same as the one of the polycurve's convex hull and as a result we compute this last topological structure. The conditions we give are sufficient, but not necessary, this being the prize to pay for avoiding algebraic equations.

5.1. Polygonal sufficient conditions of topological relevance

To state the announced conditions, we will of course use the polygon $\mathcal{Q}$. But its expression given by Formula (1) is not witty enough, because it does not consider the control polygons of the Bézier curves composing the polycurve separately, dealing only with all the control polygons obtained by subdivision from a same initial one in a group. Thus, we give here another form of the polygon $\mathcal{Q}$, which is equivalent to the first one modulo a renaming of the vertices of $\mathcal{P}$.

We recall that the control polygon of the polycurve was written $\mathcal{P} = \cup_{i=1}^{n} \mathcal{P}_i$, where $\mathcal{P}_i = \mathrm{P}_0^{(i)} \dots \mathrm{P}_{m_i}^{(i)}$. Then it exists $\mathcal{A} = \{a_1 < a_2 < \dots < a_r\} \subset \{1, \dots, n\}$ and $u_a \in \{0, \dots, m_a\}$, $v_a \in \{0, \dots, m_a - u_a\}$ for all $a \in \mathcal{A}$ such that

$$\mathcal{Q} = \mathrm{P}_{u_{a_1}}^{(a_1)} \mathrm{P}_{u_{a_1}+1}^{(a_1)} \dots \mathrm{P}_{u_{a_1}+v_{a_1}}^{(a_1)} \mathrm{P}_{u_{a_2}}^{(a_2)} \mathrm{P}_{u_{a_2}+1}^{(a_2)} \dots \mathrm{P}_{u_{a_2}+v_{a_2}}^{(a_2)} \dots \mathrm{P}_{u_{a_r}}^{(a_r)} \mathrm{P}_{u_{a_r}+1}^{(a_r)} \dots \mathrm{P}_{u_{a_r}+v_{a_r}}^{(a_r)} \qquad (2)$$

The criteria we mentioned above are then given by the following result:

Proposition 5.1. – *Using the previous notations, if for all $k \in \{1, \dots, r\}$ the three following conditions are satisfied, then $\mathcal{H}(\mathcal{P})$ is topologically relevant to $\mathcal{H}(\mathcal{B})$:*

(5.1.i) *any vertex* P *of* $\cup_{i=a_{k-1}+1}^{a_k - 1} \mathcal{P}_i$ *which does not belong to* $\mathrm{DP}\big(\mathrm{P}_0^{(a_k)} \mathrm{P}_1^{(a_k)}, \mathcal{B}_{a_k}\big)$ *is a vertex of* $\widetilde{\mathcal{P}}_{a_k - 1}$*;*

(5.1.ii) *any vertex* P *of* $\cup_{i=a_k+1}^{a_{k+1} - 1} \mathcal{P}_i$ *which does not belong to* $\mathrm{DP}\big(\mathrm{P}_{m_{a_k}-1}^{(a_k)} \mathrm{P}_{m_{a_k}}^{(a_k)}, \mathcal{B}_{a_k}\big)$ *is a vertex of* $\widetilde{\mathcal{P}}_{a_k + 1}$*;*

(5.1.iii) $\{\mathrm{P}_0^{(a_k)}, \mathrm{P}_{m_{a_k}}^{(a_k)}\} \not\subset \mathrm{Int}\Big(\mathcal{H}(\mathcal{P} \setminus \widetilde{\mathcal{P}}_{a_k})\Big)$.

Figures 9 and 10 present examples for this three properties.

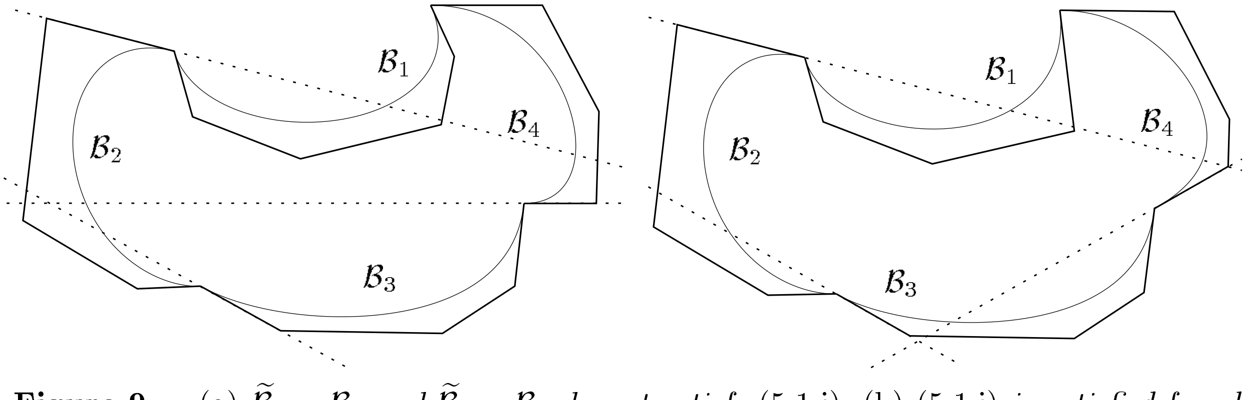


Figure 9. (a) $\widetilde{\mathcal{B}}_2 = \mathcal{B}_2$ *and* $\widetilde{\mathcal{B}}_4 = \mathcal{B}_4$ *do not satisfy* (5.1.i)*;* (b) (5.1.i) *is satisfied for all* $\kappa \in \{1, \dots, \pi\}$.

The condition (5.1.ii) is symmetrical to the condition (5.1.i).

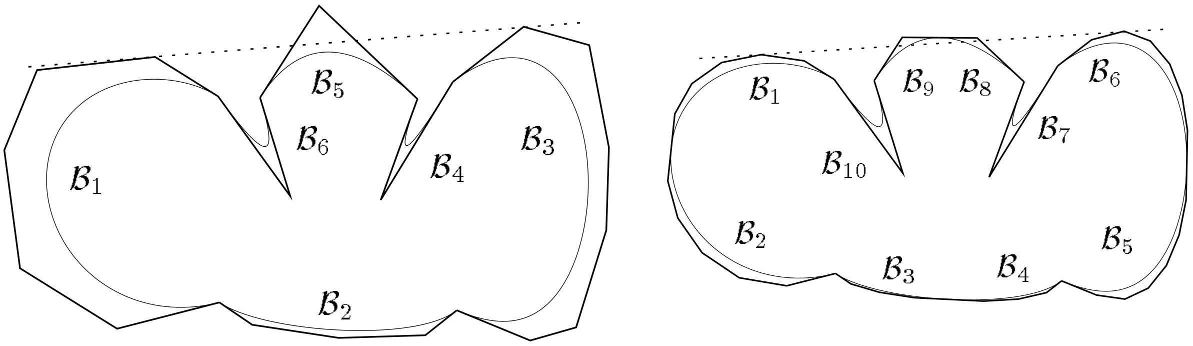


Figure 10. (a) $\mathcal{B}_5$ *does not satisfy* (5.1.iii)*;* (b) $\mathcal{B}_8$ *and* $\mathcal{B}_9$ *satisfy* (5.1.iii)

Proof – (Proposition 5.1) – The property (4.2.i) results straightforward from (5.1.iii). To prove that the condition (4.2.ii) is fulfilled, we take $i \in \{1, \ldots, n\}$ satisfying $\widetilde{\mathcal{P}}_i \subset \mathrm{Int}\Big(\mathcal{H}\big(\cup_{\widetilde{\mathcal{B}}_j \neq \widetilde{\mathcal{B}}_i} \mathcal{P}_j\big)\Big)$ and we show that $\widetilde{\mathcal{B}}_i \subset \mathcal{H}\big(\cup_{k=1}^{r} \widetilde{\mathcal{B}}_{a_k}\big)$.

There exists a unique $k_0 \in \{1, \ldots, r\}$ such that $i \in \{a_{k_0}, \ldots, a_{k_0+1}-1\}$. Let $\mathrm{R}'_{a_{k_0}} \in \widetilde{\mathcal{B}}_{a_{k_0}}$ and $\mathrm{R}_{a_{k_0}+1} \in \widetilde{\mathcal{B}}_{a_{k_0}+1}$ be such that $\widetilde{\Delta}_{a_{k_0} a_{k_0}+1} = \mathrm{R}'_{a_{k_0}} \mathrm{R}_{a_{k_0}+1}$.

Let us suppose that $\widetilde{\mathcal{B}}_i \not\subset \mathcal{H}\big(\cup_{k=1}^{r} \widetilde{\mathcal{B}}_{a_k}\big)$. Then the curve $\mathcal{U}_{a_{k_0} a_{k_0+1}} = \cup_{i=a_{k_0}+1}^{a_{k_0+1}-1} \mathcal{B}_i$ must intersect the boundary of this convex hull, $\delta\Big(\mathcal{H}\big(\cup_{k=1}^{r} \widetilde{\mathcal{B}}_{a_k}\big)\Big)$. As the polycurve $\mathcal{B}$ is a simple curve, $\mathcal{U}_{a_{k_0} a_{k_0+1}}$ has to intersect one of the segments $\left]\mathrm{R}'_{a_k} \mathrm{R}_{a_{k+1}}\right[$ and it is easy to see that if this segment is not $\left]\mathrm{R}'_{a_{k_0}} \mathrm{R}_{a_{k_0+1}}\right[$, the continuity of the polycurve implies (Figure 11)

$$\left(\bigcup_{i=a_k+1}^{a_{k+1}-1} \mathcal{B}_i\right) \cap \left]\mathrm{R}'_{a_k} \mathrm{R}_{a_{k+1}}\right[\neq \emptyset$$

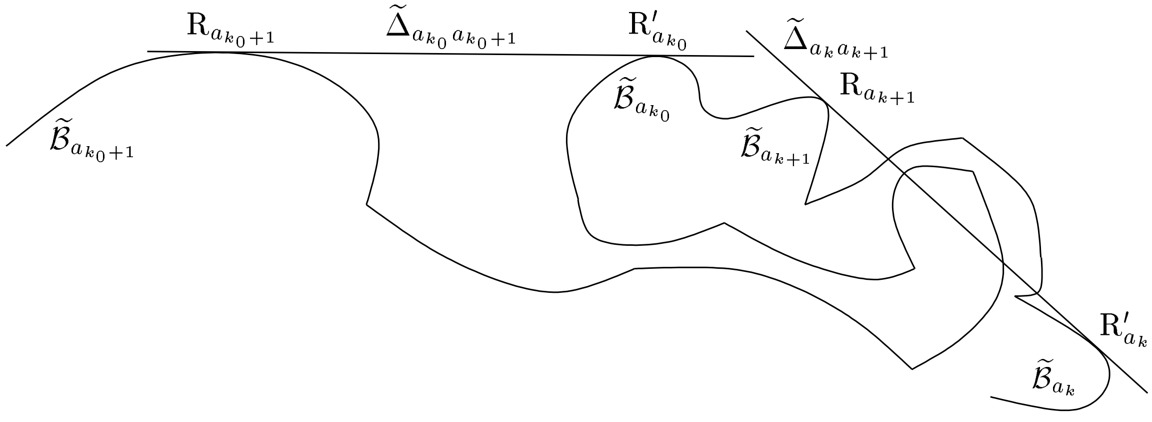


Figure 11. *If* $\mathcal{U}_{a_{k_0} a_{k_0+1}}$ *intersects* $\left]\mathrm{R}'_{a_k} \mathrm{R}_{a_{k+1}}\right[$, *then* $\mathcal{U}_{a_k a_{k+1}}$ *intersects* $\left]\mathrm{R}'_{a_k} \mathrm{R}_{a_{k+1}}\right[$

Thus, we can suppose that $\mathcal{U}_{a_{k_0} a_{k_0+1}} \cap \left]\mathrm{R}'_{a_{k_0}} \mathrm{R}_{a_{k_0+1}}\right[\neq \emptyset$. The conditions (5.1.i) and (5.1.ii) imply that all the vertices of the polygonal line $\cup_{i=a_{k_0}+1}^{a_{k_0+1}-1} \mathcal{P}_i$, except its extremities, belong to the region

$$\mathcal{R}_{a_{k_0} a_{k_0+1}} = \mathrm{DP}\Big(\mathrm{P}^{(a_{k_0})}_{m_{a_{k_0}}-1} \mathrm{P}^{(a_{k_0})}_{m_{a_{k_0}}}, \mathcal{B}_{a_{k_0}}\Big) \cap \mathrm{DP}\Big(\mathrm{P}_0^{(a_{k_0+1})} \mathrm{P}_1^{(a_{k_0+1})}, \mathcal{B}_{a_{k_0+1}}\Big)$$

The two lines determining this region, $\mathrm{P}^{(a_{k_0})}_{m_{a_{k_0}}-1}\mathrm{P}^{(a_{k_0})}_{m_{a_{k_0}}}$ and $\mathrm{P}^{(a_{k_0+1})}_0\mathrm{P}^{(a_{k_0+1})}_1$, are respectively tangent to $\mathcal{B}_{a_{k_0}}$ and $\mathcal{B}_{a_{k_0+1}}$. Let us make the following notations:

$$\mathcal{C}_{k_0} = \mathcal{B}_{a_{k_0}}\Big|_{(\mathrm{R}'_{a_{k_0}},\widetilde{\mathrm{P}}^{(a_{k_0})}_{\widetilde{m}_{a_{k_0}}})} \quad \text{and} \quad \mathcal{C}_{k_0+1} = \widetilde{\mathcal{B}}_{a_{k_0}+1}\Big|_{(\widetilde{\mathrm{P}}^{(a_{k_0+1})}_0,\mathrm{R}_{a_{k_0}+1})}$$

If $\mathrm{P}^{(a_{k_0})}_{m_{a_{k_0}}} \in \mathcal{C}_{k_0}$ and $\mathrm{P}^{(a_{k_0+1})}_0 \in \mathcal{C}_{k_0+1}$, then $\mathcal{R}_{a_{k_0}a_{k_0+1}} \subset \mathrm{DP}\left(\widetilde{\Delta}_{a_{k_0}a_{k_0+1}}, \widetilde{\mathcal{B}}_{a_{k_0}}\right)$ (Figure 12) and the supposition we made immediately leads us to a contradiction.

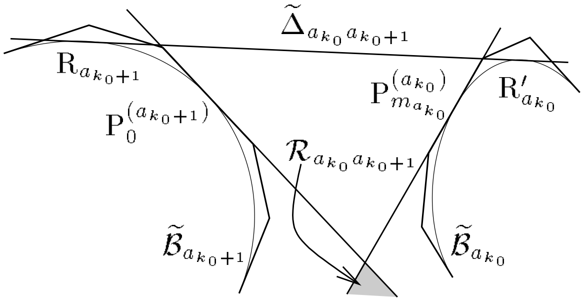


Figure 12.

It is obvious that we cannot have simultaneously $\mathrm{P}^{(a_{k_0})}_{m_{a_{k_0}}} \notin \mathcal{C}_{k_0}$ and $\mathrm{P}^{(a_{k_0+1})}_0 \notin \mathcal{C}_{k_0+1}$. If only one of the two points belongs to the corresponding part of Bézier curve, then $\mathcal{R}_{a_{k_0}a_{k_0+1}} \cap \mathrm{DP}_{op}\left(\widetilde{\Delta}_{a_{k_0}a_{k_0}+1}, \widetilde{\mathcal{B}}_{a_{k_0}}\right) \neq \emptyset$, as shown Figure 13. Yet, even in this case we can easily see that all the vertices of the polygonal line $\cup_{i=a_{k_0}+1}^{a_{k_0+1}-1}\mathcal{P}_i$ must lie in the half-plane $\mathrm{DP}\left(\widetilde{\Delta}_{a_{k_0}a_{k_0}+1}, \widetilde{\mathcal{B}}_{a_{k_0}}\right)$ and the properties of the completely convex Bézier curves imply a contradiction with the supposition $\mathcal{U}_{a_{k_0}a_{k_0+1}} \cap \left]\mathrm{R}'_{a_{k_0}}\mathrm{R}_{a_{k_0+1}}\right[\neq \emptyset$.

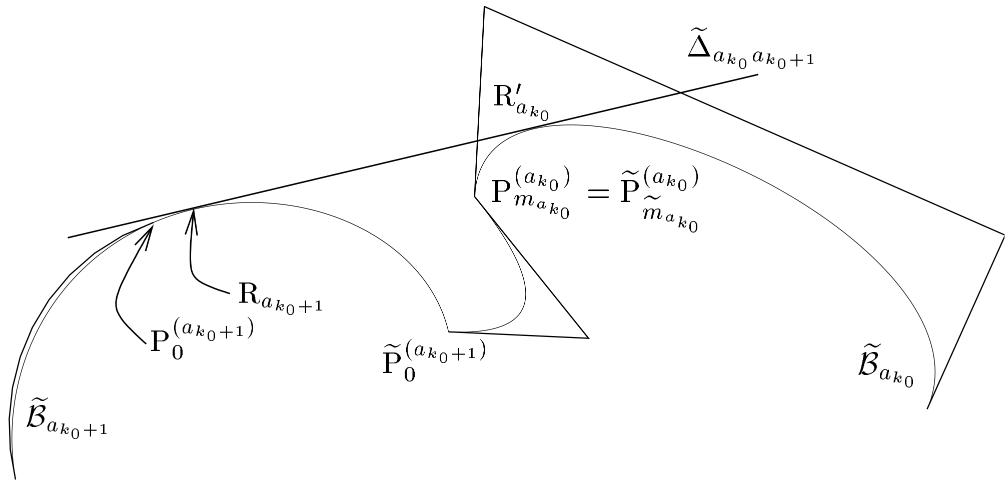


Figure 13.

We have so demonstrated that the second condition of topological relevance is also met and thus the proof of the proposition is achieved.

□

5.2. Degeneracy and ε-degeneracy

The proposition of the previous subsection provides the "polygonal" criteria of topological relevance we were looking for. But the conditions (5.1.i), (5.1.ii) and (5.1.iii) do not assure the convergence of the algorithm because of the degenerated cases, described by the following definition.

Definition 5.2. – *Let $\mathcal{I} = \{i_1, \ldots, i_p\}$ be a set giving the topological structure of $\mathcal{H}(\mathcal{B})$. We say that the polycurve $\mathcal{B}$ is degenerated, or that it presents a degeneracy, iff there exists $k \in \{1, \ldots, p\}$ and $i \in \{1, \ldots, n\} \setminus \{i_k, i_{k+1}\}$ such that $\Delta_{i_k i_{k+1}}$ is tangent to $\mathcal{B}_i$.*

Figure 2 presents a case of degeneracy. It is obvious that in such a case none of the three conditions will ever be fulfilled, no matter the number of subdivisions we apply to the curves $\mathcal{B}_{i_k}$, $\mathcal{B}_{i_{k+1}}$ and $\mathcal{B}_i$, unless the three points of tangency, R'_{i_k}, $\mathrm{R}_{i_{k+1}}$ and $\Delta_{i_k i_{k+1}} \cap \mathcal{B}_i$, correspond, respectively on the Bézier curves they belong to, to values of the parameter which are multiple of 1 over a power of 2 (Figure 14). But it is very rare that, in a degenerated case, the three tangency points correspond all to this type of parameter, so we must introduce a supplementary test to avoid that our algorithm infinitely cycles.

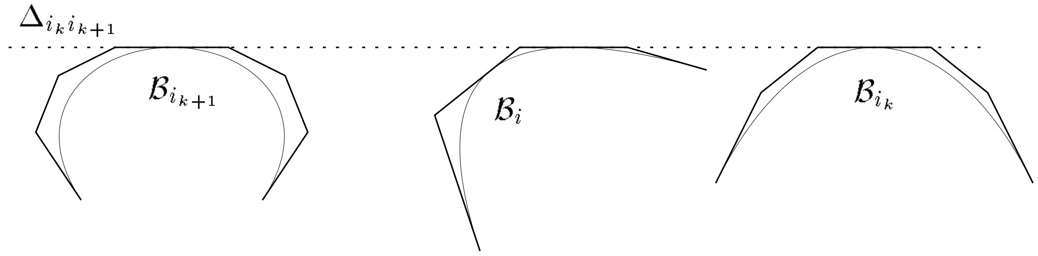


Figure 14. *"Good" degeneracy case*

It is easy to see that the successive control polygons of the polycurve do not suffice to decide whenever a polycurve is degenerated, except for the cases described above and presented Figure 14. So, the test we mentioned will be based on the notion introduced by the following definition.

Definition 5.3. – *Let $\mathcal{I} = \{i_1, \ldots, i_p\}$ be a set giving the topological structure of $\mathcal{H}(\mathcal{B})$. We say that the polycurve $\mathcal{B}$ is ε-degenerated, for a given $\varepsilon \geq 0$, iff there exists $k \in \{1, \ldots, p\}$ and $i \in \{1, \ldots, n\} \setminus \{i_k, i_{k+1}\}$ such that $\mathcal{B}_i \cap B(\Delta_{i_k i_{k+1}}, \varepsilon)$ is a non-empty and connected set, where*

$$B(\Delta_{i_k i_{k+1}}, \varepsilon) = \left\{\mathrm{P} \mid \mathrm{d}(\mathrm{P}, \Delta_{i_k i_{k+1}}) \leq \varepsilon\right\}$$

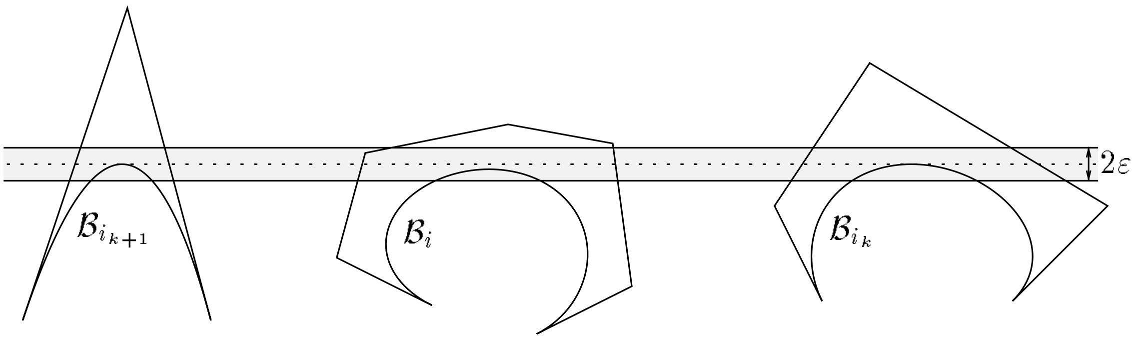


Figure 15. *$\mathcal{B}_i \cap B(\Delta_{i_k i_{k+1}}, \varepsilon)$ is non-empty and connected*

It is obvious that any polycurve $\mathcal{B}$ is ε-degenerated for a value ε big enough.

Definition 5.4. – *We will call parameter of degeneracy (or degeneracy parameter) of the polycurve $\mathcal{B}$ the value ε_0 fulfilling*

$$\varepsilon_0 = \inf\{\varepsilon > 0 \mid \mathcal{B} \textit{ is } \varepsilon - \textit{degenerated}\}$$

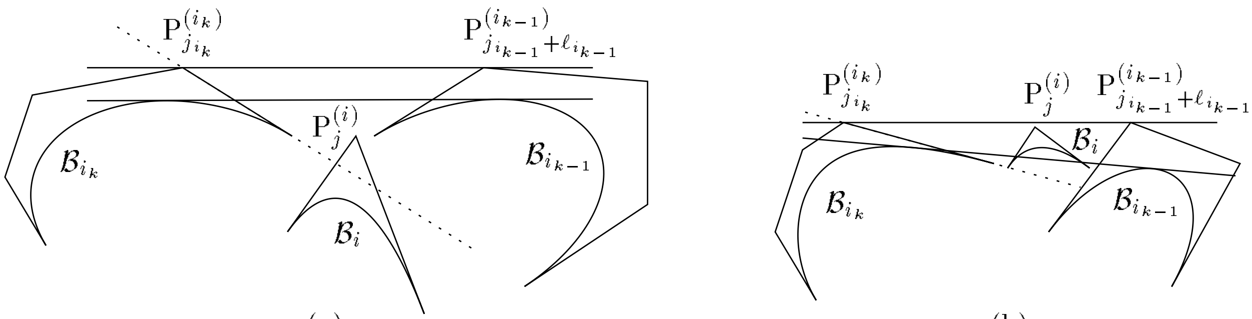
5.3. Evaluation of the degeneracy parameter

Let us suppose that there exists $k \in \{1, \ldots, p\}$ such that the condition (5.5.i) is not satisfied. We want to evaluate the degeneracy parameter ε_0 of the polycurve $\mathcal{B}$.

If (5.5.i) is not fulfilled, there exists $k \in \{1, \ldots, p\}$, $i \in \{i_{k-1}+1, \ldots, i_k - 1\}$ and $j \in \{0, \ldots, m_i\}$ such that $\widetilde{\mathcal{P}}_i \neq \widetilde{\mathcal{P}}_{i_{k-1}}$ and $\mathrm{P}_j^{(i)} \notin \mathrm{DP}\big(\mathrm{P}_0^{(i_k)}\mathrm{P}_1^{(i_k)}, \mathcal{B}_{i_k}\big)$. We will then take

$$\begin{aligned}\varepsilon_0 = \max\Big\{&\delta^{\mathrm{H}}(\mathcal{P}_i, [\mathrm{P}_0^{(i)}\mathrm{P}_{m_i}^{(i)}]) + \mathrm{d}(\mathrm{P}_j^{(i)}, \Gamma_{i_{k-1}i_k}),\\ &\max\big\{\delta^{\mathrm{H}}(\mathcal{P}_{i_{k-1}}, [\mathrm{P}_0^{(i_{k-1})}\mathrm{P}_{m_{i_{k-1}}}^{(i_{k-1})}]), \delta^{\mathrm{H}}(\mathcal{P}_{i_k}, [\mathrm{P}_0^{(i_k)}\mathrm{P}_{m_{i_k}}^{(i_k)}]\big\}\Big\}\end{aligned} \tag{3}$$

This value is a rough upper bound of the wanted ε_0, but a better one could not be obtained based only on the control polygon. Figure 16 presents the two possible positions of $\mathcal{B}_i$ with respect to the line $\Delta_{i_{k-1}i_k}$: the one of Figure 16(a) requires the first term of the max in the formula (2), and the one in Figure 16(b) requires the second term.



(a) (b)

Figure 16. *Positions of $\mathcal{B}_i$ with respect to $\Delta_{i_{k-1}i_k}$*

The condition (5.5.ii) is symmetrical to (5.5.i). If there exists $k \in \{1, \ldots, p\}$ such that the condition (5.5.iii) is not satisfied, we take

$$\varepsilon_0 = \delta^{\mathrm{H}}(\mathcal{P}_{i_k}, [\mathrm{P}_0^{(i_k)}\mathrm{P}_{m_{i_k}}^{(i_k)}]) \tag{4}$$

6. The convex hull algorithm

6.1. The sketch of the algorithm

```
 1. Topological_ch(polycurve 𝓑, real ε, top_ch 𝓗)
 2.       set_of_integers 𝓘, 𝓙
 3.       integer M
 4.       integer i = 0
 5.    begin
 6.         input_data(𝓑, M)
 7.         control_polygon_conv_hull(𝓑,𝓘)
 8.         topological_relevance(𝓑,𝓘,𝓙)
 9.         while 𝓙 ≠ ∅ and i ≤ M
10.              begin
11.              subdivision(𝓑,𝓙)
12.              i = i + 1
13.              control_polygon_conv_hull(𝓑,𝓘)
14.              topological_relevance(𝓑,𝓘,𝓙)
15.              end
16.         if i > M
17.              begin
18.              input_data(ε)
19.              ε-degeneracy(𝓑,𝓘,ε,𝓙)
20.              while 𝓙 ≠ ∅
21.                   begin
22.                   subdivision(𝓑,𝓙)
23.                   control_polygon_conv_hull(𝓑,𝓘)
24.                   ε-degeneracy(𝓑,𝓘,ε,𝓙)
25.                   end
26.              end
27.         output_results(𝓑,𝓘,𝓗)
28.    end
```

This algorithm is separated in two main parts. The degenerated polycurves are quite rare, that is why we start by trying to compute the exact topological structure of the polycurve's convex hull (the cycle lines 9-15). If after an user given M number of iterations the control polygon still does not satisfy the conditions (5.1.i), (5.1.ii), (5.1.iii), an user given tolerance ε is read and the cycle of lines 20-25 iterates until either we obtain a control polygon fulfilling (5.1.i), (5.1.ii), (5.1.iii) or we can decide that the polycurve is ε-degenerated.

6.2. The procedure `control_polygon_conv_hull`

This procedure is called lines 7, 13 and 23 and it computes the convex hull of the control polygon of the polycurve $\mathcal{B}$. It takes as input the polycurve (the control polygon is provided by this structure) and returns the set $\mathcal{I} = \{i_1, \ldots, i_p\}$ of formula (1). As we have remarked in the proof of Lemma 4.3, the set $\mathcal{I}$ is the largest possible. This algorithm is linear in the size of the control polygon, so its complexity is, in the worst case, $\mathcal{O}(nd)$.

6.3. *The procedure* `topological_relevance`

This procedure, called lines 8 and 14, verifies, for all $k \in \{1, \ldots, p\}$, if the conditions (5.1.i), (5.1.ii) and (5.1.iii) are fulfilled. The input of this routine is the polycurve $\mathcal{B}$ and the set $\mathcal{I}$ computed by `control_polygon_conv_hull` and its output is the set $\mathcal{J}$ of all the indices i_k for which at least one of the mentioned conditions is not satisfied.

The complexity of `topological_relevance` is also $\mathcal{O}(nd)$. Indeed, every vertex of a control polygon $\mathcal{P}_i$ will be processed $\mathcal{O}(1)$ times for the condition (5.1.i) and $\mathcal{O}(1)$ times for the condition (5.1.ii), if i does not belong to the set of indices $\mathcal{A}$ defined Subsection 5.1, or $\mathcal{O}(1)$ times for the condition (5.1.iii), if it does. The sum of all these costs gives the announced complexity for the procedure.

6.4. *The procedure* `subdivision`

This procedure, called lines 11 and 22, subdivides the Bézier curves identified by the set of indices $\mathcal{J}$ and brings the polycurve $\mathcal{B}$ up-to-date after these subdivisions. The de Casteljau algorithm for the subdivision of a Bézier curve of degree d_0 runs in $\mathcal{O}(d_0^2)$, so this procedure's complexity will be $\mathcal{O}(d^2 \operatorname{card}(\mathcal{J}))$. Let us study the size of the set $\mathcal{J}$ for a more precise evaluation of this procedure's cost.

The following properties are intuitive and very easy to prove.

Property 6.1. – *If all the vertices* $\mathrm{P}_0^{(i)}, \mathrm{P}_1^{(i)}, \ldots, \mathrm{P}_{m_i}^{(i)}$ *of the polygon* $\mathcal{P}_i$ *are vertices of* $\mathcal{Q}$*, then the Bézier curve* $\mathcal{B}_i$ *entirely lies on the curve* $\mathcal{H}$.

Property 6.2. – *If the only vertices the polygon* $\mathcal{P}_i$ *participates to* $\mathcal{Q}$ *with are in the set* $\{\mathrm{P}_0^{(i)}, \mathrm{P}_{m_i}^{(i)}\}$*, then the only points the curve* $\mathcal{B}_i$ *participates to* $\mathcal{H}$ *with are also in the set* $\{\mathrm{P}_0^{(i)}, \mathrm{P}_{m_i}^{(i)}\}$.

Using these two properties, we subdivide only those Bézier curves which have a control polygon participating to the polygon $\mathcal{Q}$ with some vertices, but neither with all its vertices, nor only with its extremities.

All the curves $\mathcal{B}_i$ are completely convex, so if two vertices P_1 and P_2 of the polygon $\widetilde{\mathcal{P}}_i$ are vertices of $\mathcal{Q}$, then one (and only one) of the following cases will occur:

1. all the vertices of $\widetilde{\mathcal{P}}_i$ "between" P_1 and P_2 are vertices of $\mathcal{Q}$;
2. $\mathrm{P}_1 = \widetilde{\mathrm{P}}_0^{(i)}$, $\mathrm{P}_2 = \widetilde{\mathrm{P}}_{\widetilde{m}_i}^{(i)}$ and no other vertex of $\widetilde{\mathcal{P}}_i$ is a vertex of $\mathcal{Q}$.

So, among the curves contained in $\widetilde{\mathcal{P}}_i$ at most two will satisfy the condition above, having a control polygon participating to the polygon $\mathcal{Q}$ with some vertices, but not with all its vertices. That means that $\operatorname{card}(\mathcal{J}))$ is inferior to $2n_0$ at each iteration, n_0 being the number of Bézier curves initially composing the polycurve. This also allows us to give an upper bound of the number n of Bézier curves of the current composition of the polycurve depending of the number of applied iterations κ: $n \leq 2\kappa n_0$.

6.5. *The procedure* ε`-degeneracy`

This procedure, called lines 19 and 24, starts by verifying, for all $k \in \{1, \ldots, p\}$, if the conditions (5.1.i), (5.1.ii), (5.1.iii) are fulfilled. For every i_k for which one of the conditions (5.1.i), (5.1.ii), (5.1.iii) is not satisfied, it computes the value ε_0 given by the formulae (3) and (4) and it compares it to ε. If $\varepsilon_0 > \varepsilon$, the index i_k is inserted in the set $\mathcal{J}$ identifying the Bézier curves to be subdivided. This procedure's complexity is also $\mathcal{O}(nd)$.

6.6. *The cost of* `Topological_ch`

To evaluate the complexity of this algorithm, we need to know the total number of iterations that a polycurve must pass through, which is not possible. Indeed, the conditions (5.5.i), (5.5.ii), (5.5.iii) do not allow an a priori estimation of this number of iterations. If we consider that κ_T denotes the total number of iteration, it is not difficult to see that the cost of the algorithm will be $\mathcal{O}(\kappa_T n_0 d^2 + \kappa_T^2 n_0 d)$. The experimental results give us an estimation for the average number of iterations that a polycurve will have to pass through, as we will see next section.

7. Examples and numerical results

The following figures present some examples of polycurves and their convex hulls. For each one, relevant values are given:

- m_0 is the size of the control polygon before any subdivision
- m_f is the size of the final control polygon, the one providing the topological structure of $\mathcal{H}(\mathcal{B})$
- κ_T is the number of iterations applied
- m_f/m_0 is the growing rate of the control polygon

In the previous section, we have seen that $n_f \leq 2\kappa_T n_0$, where n_f is the number of Bézier curves composing the final polycurve. The similar inequality for the sizes of the control polygons is not true, because not all of the Bézier curves in the composition of the polycurve have the same degree. Yet, we can say that in the general case m_f is 'about' $2\kappa_T m_0$. In the following, we will see that the average m_f/m_0 is far from reaching $2\kappa_T$.

Figure 17 presents a degenerated polycurve. We set the integer M in the algorithm to 16, and after M iterations the value of degeneracy ε_0 given Section 5 is inferior to 10^{-4}.

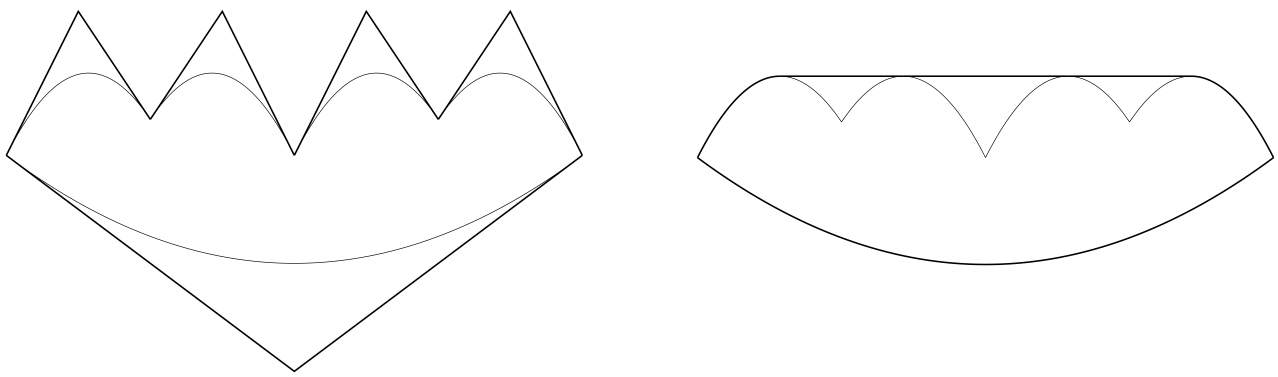


Figure 17. $m_0 = 10$, $m_f = 138$, $\kappa_T = 16$, $m_f/m_0 = 13.8$

The following figures present some other examples.

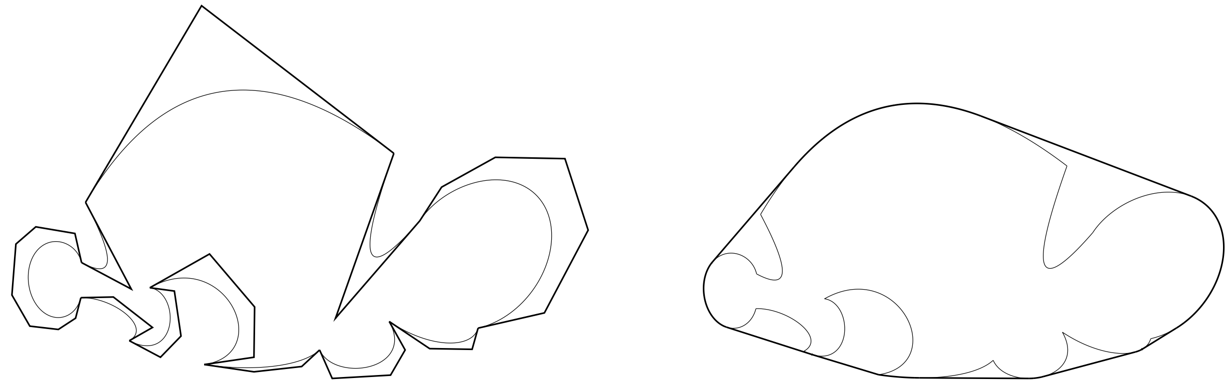


Figure 18. $m_0 = 41$, $m_f = 184$, $\kappa_T = 8$, $m_f/m_0 = 4.4878$

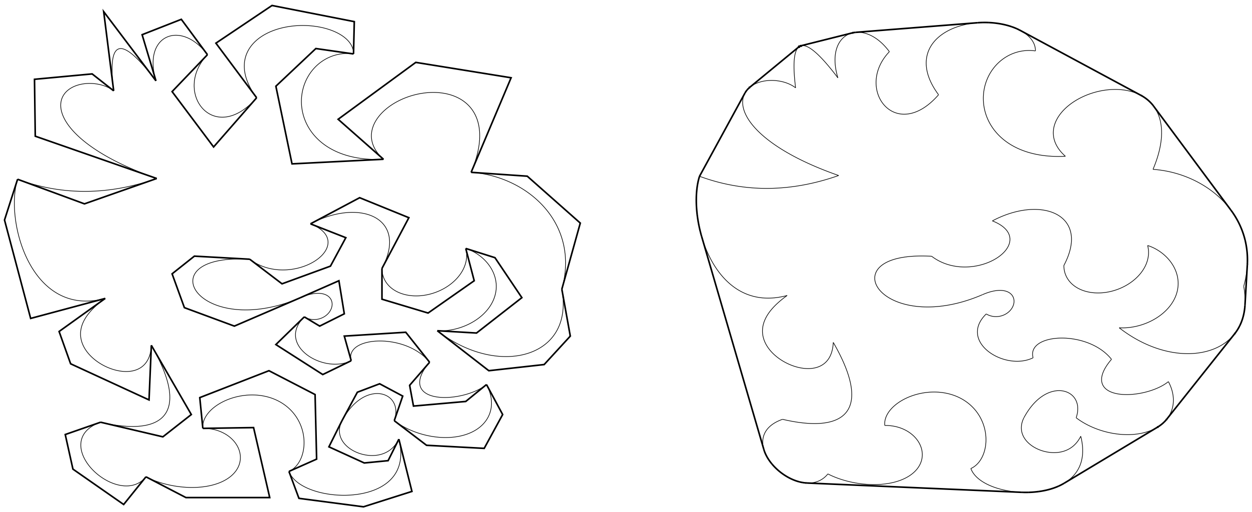


Figure 19. $m_0 = 106$, $m_f = 328$, $\kappa_T = 9$, $m_f/m_0 = 3.1$

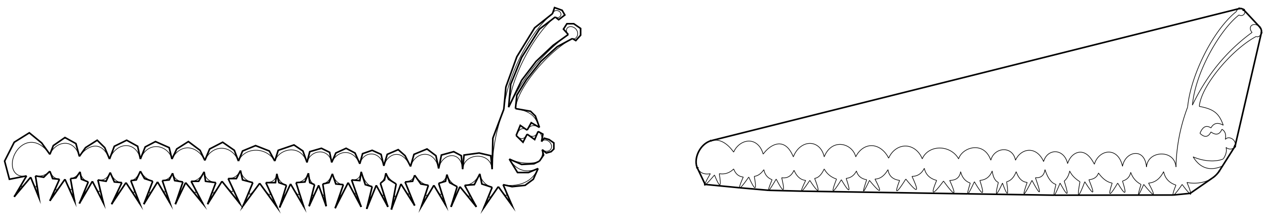


Figure 20. $m_0 = 253$, $m_f = 677$, $\kappa_T = 15$, $m_f/m_0 = 2.676$

Figure 21. $m_0 = 191$, $m_f = 703$, $\kappa_T = 15$, $m_f/m_0 = 3.68$

All these values have been computed for about one hundred polycurves. For comparison, a method computing an approached convex hull of a given polycurve has been implemented. This method consists in subdividing all the Bézier curves given by the set $\mathcal{I}$ of the Formula (1) and which satisfy neither the Property 6.1 nor the Property 6.2 until

$$\max_{k=1}^{p} \delta^{\mathrm{H}}(\mathcal{B}_{i_k}, \mathcal{P}_{i_k}) < \varepsilon$$

for an user given value ε. For $\delta^{\mathrm{H}}(\mathcal{B}_{i_k}, \mathcal{P}_{i_k})$, which can not be computed using only the control polygon, the upper bound $\delta^{\mathrm{H}}(\mathcal{P}_{i_k}, [\mathrm{P}_0^{i_k}\mathrm{P}_{m_{i_k}}^{(i_k)}])$ has been used. It is easy to prove that if the condition above is satisfied, then the Hausdorff distance between $\mathcal{Q}$ and $\mathcal{H}$ is also inferior to ε, so that $\mathcal{Q}$ represents a ε-approximation of $\mathcal{H}$. We remark that using the Properties 6.1 and 6.2 greatly reduces the cost of this algorithm, the inconvenient being that the obtained result is not a polygon, but a polycurve.

The table bellow gives the average values of the quantities in the first column computed on the set of polycurves. The second column gives these average values for the initial composition of the polycurves, the third one for the final structure of the polycurves obtained running the algorithm computing the topological structure of $\mathcal{H}$ and the columns 4 to 6 give these values for the final structure of the polycurves obtained running the algorithm computing the ε-approximation of $\mathcal{H}$ for ε respectively equal to 10^{-1}, 10^{-2} and 10^{-3}.

	Initial polycurve	Topological convex hull	Approached convex hull		
			$\varepsilon = 10^{-1}$	$\varepsilon = 10^{-2}$	$\varepsilon = 10^{-3}$
n	12.63	37.74	44.29	57.55	70.59
m	42.56	148.87	177.235	233.67	288.35
κ_T	—	6.11	4.7	6.33	7.88
m_f/m_0	—	3.62	5.11	6.7	8.25

We remark that the algorithm computing the approached convex hull generates a larger growth of the control polygon than the one computing the topological structure of $\mathcal{H}$, even when the number of needed iterations is smaller, as it happens for $\varepsilon = 10^{-1}$. This reveals that the subdivisions required by the second algorithm are far more selective. This becomes even more obvious when we add that the ε-approached convex hull is not always topologically relevant to $\mathcal{H}$: when $\varepsilon = 10^{-1}$ the topological relevance is not satisfied for 55.3% of the treated polycurves, when $\varepsilon = 10^{-2}$ there are 44.7% of the polycurves whose control polygon does not fulfill the conditions of topological relevance and when $\varepsilon = 10^{-3}$ there are still 29.41% polycurves in the same case. These results prove once again that the nature itself of the problem of computing the topological structure of a polycurve's convex hull is different from the one of the problem of computing an approximation of this convex hull.

We also remark that the growth of the control polygon generated by the algorithm presented in this paper is in practice far smaller than the theoretical one given Section 6, and so the complexity of the algorithm will also be smaller than the theoretical one, because it directly depends on the control polygon's size.

8. Conclusion

In this paper, we have presented a robust convex hull algorithm for a wide class of non-linear planar objects, named polycurves. Our approach consists in using suitable converging polygonal approximations of the treated object and it emphasizes the topological structure of the convex hull rather then its explicit description. So, we are obtaining a complete and topologically correct expression of the convex hull's boundary, the main advantage of the method being that no algebraic equations are to be solved.

Yet, as it is presented here, our method is not perfectly accurate. This is due to the use of floating point arithmetic for the computations performed on linear objects. So, a natural improvement is to employ either more precise (lazy or interval) arithmetic for these computations, or determinant methods for precise geometric predicates (Brönnimann and Yvinec, 1997). Also, as we mentioned in the introduction, it can be of interest to study the choice of the subdivision parameter for a faster convergence of the algorithm we propose.

As we have remarked Section 3, the proposed algorithm works not only on polycurves composed by completely convex Bézier curves, but also on polycurves which are union of completely convex rational Bézier curves, B-splines or NURBS. Moreover, an extension to general convex parametric curves worths being considered, because feasible under certain (weak) conditions.

On the other hand, other important topics of the computational geometry can be addressed via the same method. For example, effective results have been obtained for the computation of the incidence graph of an arrangement of polycurves, Neagu(1998). Our approach is also convenient for some other problems, as the lower bound computation for a set of polycurves or the incidence graph computation for the Voronoi diagram of a set of polycurves.

Another natural development seems to be the generalisation to "polysurfaces", composite completely convex (Bézier) surfaces. But, if the concrete techniques used in 2D can be easily tranported in 3D, the interest of the topological structure of the convex hull is smaller in the space. Indeed, in the plane, the objects joining the pieces of the polycurve's boundary intervening in the convex hull's boundary are segments, which are (implicitely) perfectly defined. In the space, the similar objects are complex (ruled) surfaces of an unknown type. Thus, the topological structure contains less information on the expression of the convex hull in the 3D case than in the 2D case. Nevertheless, as we mentioned, the method itself can be extended to the space problem with no difficulty.

9. Bibliography

Bajaj, C. and Kim, M.-S. (1991), Convex Hulls of Objects Bounded by Algebraic Curves, Algorithmica 6, 533-553.

Brönnimann, H., and Yvinec, M. (1997), Efficient exact evaluation of signs of determinants, Proc. 13^{th} ACM Symposium on Computational Geometry, 166-173.

Calcoen, E. (1996), *Approximations polygonales d'objets convexes du plan pour la Géométrie Algorithmique*, PhD thesis, University Joseph Fourier, Grenoble, France.

Chazelle, B. (1996), *Application Challenges to Computational Geometry*, CG Impact Task Force Report, on Web at http://www.cs.princeton.edu/~chazelle/.

Cho, W., Maekawa, T. and Patrikalakis, N.M. (1996), Topological reliable approximation of composite Bézier curves, Computer Aided Geometric Design 13, 497-520.

Dobkin, D.P. and Souvaine, D.L. (1990), Computational Geometry in a Curved World, Algorithmica 5, 421-457.

Farin, G. (1997), *Curves and surfaces for computer aided geometric design: a practical guide*, 4th ed., London Academic Press.

Graham, R.L. (1972), An efficient algorithm for determining the convex hull of a finite planar set, Information Processing Letters 1, 132-133.

Graham, R.L. and Yao, F.F. (1983), Finding the convex hull of a simple polygon, Journal of Algorithms 4, 324-331.

Hoschek, J. and Lasser, D. (1993), *Fundamentals of Computing Aided Geometric Design*, A K Peters, Wellesley, Massachusetts.

Neagu, M. (1998), *Courbes de Bézier en Géométrie Algorithmique : approximation et cohérence topologique*, PhD thesis, University Joseph Fourier, Grenoble, France.

Preparata, F. and Shamos, M.I. (1985), *Computational Geometry: An Introduction*, Springer-Verlag, New York.

Schäffer, A. and Van Wyk, C. (1987), Convex Hulls of Piecewise-Smooth Jordan Curves, Journal of Algorithms 8, 66-94.