An Optimial Way to Encode the Outlines of Variable Sized Arabic Letters in a PostScript Font

Abdelouahad BAYAR
Ecole Supérieure de Technologie, Safi
Cadi Ayyad University
Rue Sidi Aissa, P.O.Box 89
46000, Safi, Morocco
bayar@ucam.ac.ma

Khalid SAMI
Faculty of Sciences, Marrakesh
Cadi Ayyad University
12 BD. My Abdellah, P.B 2390
40000, Marrakesh, Morocco
k_sami@ucam.ac.ma

ABSTRACT

The scripts based on the Arabic alphabet are cursive and their characters are dynamically variable. These characteristics of such scripts need changes in the fonts conception. As a consequence of the cursivity, the representation of Arabic characters in outlines may be different of those concerning the Latin ones. The characters dynamic variability can be materialized through the stretchability of some characters in both the vertical and horizontal directions, at the same time, when justifying lines for instance. In order to get texts written with stretchable characters, represented as outlines, some curves intersections, among the curves composing the characters outlines, are to be determined dynamically. The PostScript procedure to produce a dynamic character, in a dynamic font, is repeated whenever the letter is to draw. So, the determination of the curves intersections coefficients in the procedure, applying an iterative algorithm is of a high cost. A new method for developing an optimal font is to find out. In this paper, a method based on the curves comparison is presented. It allows the determination of the characters with eventually overlapping outlines. Then, a way to approximate the curves intersections coefficients is given. This is enough to remove overlapping in outlines.

Keywords

Arabic Calligraphy, Cursivity, Dynamic Stretching, Outlines Writing, Curves Comparison, Bézier Intersection, Approximation.

1. INTRODUCTION

The Arabic calligraphy rules are as important as grammar in handwriting texts in Arabic alphabet based scripts. Stretching letters is used instead of the ordinary insertion of spaces (blanks) between words, to justify lines in such texts. This stretching; the Keshideh, is almost meaningful and mandatory. There are two types of keshideh, between two connected letters and inside the same letter. These two types of stretching are presented with *bounding boxes* of characters in the figure 1 below. In the first line, there are stretchings in different sizes between the two connected letters ف and ر in the word ظفر. In the following line, the last letter ق of the word حق is stretched in four different sizes.

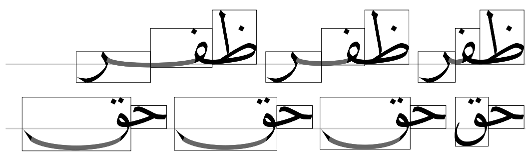


Figure 1. Inter-curvilinear and intra-curvilinear stretchings in Arabic calligraphy.

The keshideh spreads out in both *horizontal* and *vertical* directions and obey to some conditions. The horizontal stretching can go from 0 to 12 diacritic points (the diacritic point is a metric unit in Arabic calligraphy).
Of course, the keshideh is curvilinear and context dependent. A suitable font for typesetting texts in

an Arabic alphabet based script would be based on a dynamic language like PostScript [Ado99] or the instructions for encoding characters should be in TrueType fonts [Wei92]. Unfortunately, such support is not offered by Metafont [Knu86] and consequently, (La)TEX [Lam85] extensions like ArabTex [Lag92] don't support the dynamic character needed in Arabic fonts. So, only some discrete finite dynamism can be found. Currently, the system CurExt [Laz03] used in RydArab [Laz01] provides only 256 stretchings in different sizes in a given document.

Danniel Berry developed a type 3 PostScript font for the system dittroff/ffortid [Ber99]. This system offers a good support for dynamic characters. Unfortunately, the stretching goes only in the horizontal direction and the stretching model doesn't take into account the motion of the nib's head. Big stretchings can show some weakness of the calligraphic quality. A way to fix such flaws can be found in [Bayar].

The main goal of this paper is to provide a support for developing a PostScript font that allows typesetting Arabic texts, in characters represented as outlines, according to the basic calligraphic rules. The existing systems don't offer such support. In the first line of the figure 2 (from right to left), the word ظفر is displayed in outlines, in two options : with and without limits between two consecutive characters. Most of the systems allowing writing Arabic script in outlines, such as Microsoft Word, can produce only the first sample. Of course, The cursivity is better observed in the second way of writing on the figure 2.

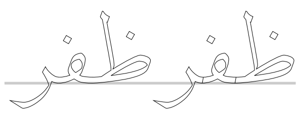


Figure 2. Possibilities of Outlines writing in Arabic.

Managing stretchability is better shown with fonts that provide characters in outlines. An example of stretched outlines is presented in the figure 3. This characters stretchability needs a mathematical formalization and the stretching model presented in [Bayar] will so be deeply improved. That's the goal of the following sections.

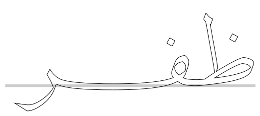


Figure 3. Stretchable outlines in Arabic.

Let us recall that there are *static* letters and *variable sized* ones. Static letters have static shapes as the letter د (standalone DEL) and ر (standalone REH) on the figure 4-a. Variable sized letters are those with stretchable parts. The first line of the figure 4-b displays the letter بـ (BEH in the beginning of a word) in three different curvilinear stretchings whereas the second line presents the letter ـر (REH in the end of a word). Variable sized letters are composed of a static part (in terms of shape; the shape is kept as it is after a shifting transformation) and a dynamic one. On the figure 4-b, the dynamic parts are in gray. For the letter بـ, this part is used to connect the letter to the following letter in the left, since the Arabic writing is a right-to-left. In the same context, the stretchable part in the other letter is the preceding connection in the letter ـر (this is used to connect it to the letter directly in its right).

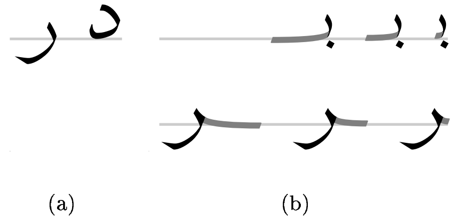


Figure 4. a) Examples of static letters, b) Examples of variable sized letters.

In the following, the paper follows the plan : the second section, presents the mathematical formalization of the nib's motion. In the third section, the way to optimize the implementation of letters in PostScript is given. The paper ends with conclusions and perspectives.

2. THE NIB'S HEAD MOTION-RAZING MODELING AND DECOMPOSITION

There are many *styles* in Arabic writing. Among the most popular ones, we can mention : Farisi, Koufi, Maghribi, Naskh, Thuluth, Rouqaa, DywAni... Samples of such styles are presented on the figure 5. For our present purpose, let us consider the Naskh style (The framed one on the figure 5) since it is the most widely used in digital typography.

In this Part, we develop the mathematical tools to be used for the development of a *stretchable* Naskh font. Of course, this font will support writing in outlines. Let's give first some characteristics of the nib's head motion under Arabic Calligraphy rules.

Figure 5. Most popular Arabic writing styles.

Qalam's Motion Modeling

In the Naskh style, the nib's head or simply the qalam (The pen calligraphers use to write with) behaves as a rectangle of width l and thickness $e = \frac{l}{6}$. This rectangle moves with a constant inclination angle of about $70°$ with regard to the baseline. A nib's head with $l = 12$ mm (and $e = 2$mm) is presented in the figure 6.

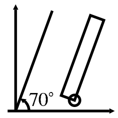


Figure 6. Nib's head in Naskh style with 12 mm width

In the following, let us consider the general case where the pen's head motion can be represented by a polygonal shape with $n+1$ vertices, in translation. Examples are given with rectangles whose sizes and inclinations are different from the Naskh qalam ones for better clarity. Let's represent a pen's head with $n+1$ vertices M_0, M_1, $\ldots$, M_n by $\mathcal{B}(M_0, \mathcal{F})$, where $\mathcal{F} = (\overrightarrow{u_i})_{i=0,\ldots,n}$ is a family of $n+1$ vectors such that $\overrightarrow{u_i} = \overrightarrow{M_0 M_i}$. The point M_0 stands for an *origin* for the pen's head motion. Whenever particular values are associated to that point and to the vectors, all the vertices, and therefore positions, of the pen's head can be determined precisely. When the trajectory of the origin vertex is the parametric curve f defined on $[0,1]$, the motion of a pen $\mathcal{B}(M_0, \mathcal{F})$ will be denoted by $\mathcal{M}(\mathcal{B}(M_0, \mathcal{F}), f, [0,1])$. In the figure 7, the motion is characterized by the start position of the pen's head and a Bézier curve [Bez77] B_0 representing the trajectory of the vertex M_{00}. In the case of the qalam, the origin is the vertex surrounded with a circle on the figure 6.

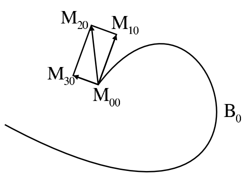


Figure 7. Example of motion

The Motion-Razing Study

The razing of a qalam motion is the set of points darken by the qalam in the plane. Of course, this set is delimited by external curves corresponding to the trajectories of the qalam's vertices and some edges of the qalam. Therefore, we need a mean to help comparing the shifting transforms of curves in order to find out the external curves of a razing. For this, it is necessary to define the notion of *the orthogonal range associated to a parametric function according to a vector.*
Consider :

- the affine space $\mathbb{R}^3$ with the orthonormal basis $R\left(O, \overrightarrow{\imath}, \overrightarrow{\jmath}, \overrightarrow{k}\right)$,
- an affine plane $\mathbb{R}^2$ in $\mathbb{R}^3$ containing the origin O with a direct orthonormal basis $(\overrightarrow{\imath}, \overrightarrow{\jmath})$. Let's denote $\overrightarrow{\mathbb{R}}^2$ the vector plane associated to $\mathbb{R}^2$.
- a parametric function f defined on $[0,1]$ with values in $\mathbb{R}^2$. Suppose that f is continuous on $[0,1]$ and differentiable on $]0,1[$.
- and the vectors $\overrightarrow{u}$, $\overrightarrow{u}_1$, $\overrightarrow{u}_2$ in $\overrightarrow{\mathbb{R}}^2$.

Definition 1 (Direct orthogonal range with respect to a vector)
The Direct orthogonal range associated to f, according to the vector $\overrightarrow{u}$, is the scalar function $\mathcal{R}_{(f,\overrightarrow{u})}$ defined from $[0,1]$ onto $\mathbb{R}$ such that $\mathcal{R}_{(f,\overrightarrow{u})}(t) = \left(\overrightarrow{u} \wedge \overrightarrow{Of(t)}\right) \cdot \overrightarrow{k}$.

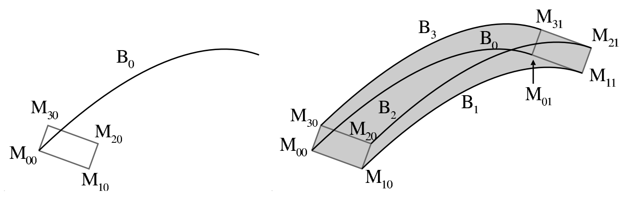
We can remark that the derivative verify $\mathcal{R}'_{(f,\overrightarrow{u})}(t) = \mathcal{R}_{(f',\overrightarrow{u})}(t)$
Now, we can define the way to compare the shifting transforms of a curve.

Definition 2 (Comparison of shifted curves)
We have $t_{\overrightarrow{u}_1}(f) \leq t_{\overrightarrow{u}_2}(f)$ on $[0,1]$ (respectively $t_{\overrightarrow{u}_1}(f) \geq t_{\overrightarrow{u}_2}(f)$) if and only if $\mathcal{R}'_{(f,\overrightarrow{u}_1-\overrightarrow{u}_2)}(t) \leq 0$ (respectively $\mathcal{R}'_{(f,\overrightarrow{u}_1-\overrightarrow{u}_2)}(t) \geq 0$).

The comparison between $t_{\overrightarrow{u}_1}(f)$ and $t_{\overrightarrow{u}_2}(f)$ on $[0,1]$ can be done through the study of the monotony of $\mathcal{R}_{(f,\overrightarrow{u}_1-\overrightarrow{u}_2)}$ on $[0,1]$.
Now, we have the tools necessary for studying the nib's head motions and their associated razings. Before doing that, let's present a particular type of motions, the *normal* motions, and their associated razings.
Consider the motion $M = \mathcal{M}(\mathcal{B}(M_{00}, \mathcal{F}), f, [0,1])$ such that $\mathcal{F} = (\overrightarrow{u_i})_{i=0,\ldots,n}$.

Definition 3 (Normal Motion) *M is said to be* normal *on $[0,1]$ if and only if $\exists\, i \in \{0,\ldots,n\}$, $\exists\, j \in \{0,\ldots,n\}$ such that :*
$t_{\overrightarrow{u}_k}(f) \leq t_{\overrightarrow{u}_i}(f)$ and $t_{\overrightarrow{u}_k}(f) \geq t_{\overrightarrow{u}_j}(f)$ on $[0,1]$ $\forall\, k \in \{0,\ldots,n\}$.

 ISBN 978-86943-15-2



(a) (b)

Figure 8. a) Normal motion, b) The corresponding normal razing.

In this definition, $t_{\overrightarrow{u}_i}(f)$ is the *maximal* curve whereas $t_{\overrightarrow{u}_j}(f)$ is the *minimal* one among the set $\{t_{\overrightarrow{u}_k}(f), \overrightarrow{u}_k \in \mathcal{F}\}$. A normal motion and its corresponding razing are presented on the figure 8. The razing is delimited by the two segments $[M_{30}M_{00}]$, $[M_{00}M_{10}]$, the Bézier curve B_1, the two segments $[M_{11}M_{21}]$, $[M_{21}M_{31}]$ and the Bézier curve B_3.

Lemma 1 *If* $\forall i \in \{0,\dots,n\}$ $\mathcal{R}_{(f,\overrightarrow{u}_{(i+1)\bmod n}-\overrightarrow{u}_i)}$ *is monotone on* $[0,1]$ *then* M *is a normal motion.*

The lemma can be proved through considering the motion on the figure 8. In general, the razing associated to a motion is determined via its decomposition into normal sub-motions. Then, we have the theorem.

Theorem 1 (Decomposition into normal sub-motions)
The motion M *can be decomposed into normal sub-motions on* $[0,1]$ *so that the concatenation of their associated razings constitute exactly the global razing associated to* M.

Proof
Let $\mathcal{F} = (\overrightarrow{u_i})_{i=0,\dots,n}$ ($\overrightarrow{u_0} = \overrightarrow{0}$),
Through studying the sign of $\mathcal{R}'_{(f,\overrightarrow{u}_{(i+1)\bmod n}-\overrightarrow{u}_i)}$ on $]0,1[$ for $i \in \{0,\dots,n-1\}$ we get the set of extrema of $\mathcal{R}_{(f,\overrightarrow{u}_{(i+1)\bmod n}-\overrightarrow{u}_i)}$ on $]0,1[$.
Let $E_i = \{T_{i,1}, T_{i,2}, \dots, T_{i,\phi(i)}\}$ be the set of extrema of $\mathcal{R}_{(f,\overrightarrow{u}_{(i+1)\bmod n}-\overrightarrow{u}_i)}$ ordered in an increasing order ($\phi(i)$ reverses the cardinal of E_i).
The restrictions of the direct orthogonal range $\mathcal{R}_{(f,\overrightarrow{u}_{(i+1)\bmod n}-\overrightarrow{u}_i)}$ to the segments $[0,T_{i,1}[$, $[T_{i,1},T_{i,2}[$, $\dots,[T_{i,\phi(i)-1},T_{i,\phi(i)}[$, $[T_{i,\phi(i)},1]$ are monotone.
Now, let us consider $E = \{T_1, T_2, \dots, T_m\}$ the union of the sets E_i, ordered in an increasing order.
According to the previous reasoning, $\mathcal{R}_{(f,\overrightarrow{u}_{(i+1)\bmod n}-\overrightarrow{u}_i)}$ is monotone on $[0,T_1[$, $[T_1,T_2[$, $\dots,[T_{m-1},T_m[$, $[T_m,1]$ for $i \in \{0,\dots,n-1\}$.
By the lemma 1, the qalam's motion is normal on $[0,T_1[$, $[T_1,T_2[$, $\dots,[T_{m-1},T_m[$, $[T_m,1]$.
The restrictions considered together constitute the exact trajectories of the vertices of the qalam's head.
So, the concatenation of the razings gives the exact razing of the pen's head. ∎
An example is given in the file attached to the paper.

3. OPTIMAL IMPLEMENTATION OF STRETCHABLE OUTLINES

This part deals with the way to solve the problem of implementing letters in their outlines in Arabic calligraphy, taking into account the stretchability. The concepts in use will be given through considering examples.
In this paragraph, the model of stretching a letter is given. Then, the concept of tolerance and digital equivalence will be highlighted. Finally, the ways to implement a static or a dynamic letter will be presented.

The Stretching Model

In variable sized letters, dynamic parts, i.e. the preceding and following connections, are quadratic Bézier curves. Let's denote a quadratic Bézier curve with control points M_0, M_1 and M_2 by $[M_0, M_1, M_2]$. The preceding connections are curves in the set $\mathcal{B}_1$ and the succeeding ones are in $\mathcal{B}_2$. The set $\mathcal{B}_1$ and $\mathcal{B}_2$ are defined in the following. Let $B = [M_0, M_1, M_2]$, then :

- $B \in \mathcal{B}_1 \Leftrightarrow \overrightarrow{M_0M_1} = \lambda\overrightarrow{\imath}$, $\lambda \in \mathbb{R}^-_*$ and $\left(\widehat{\overrightarrow{M_2M_1}, \overrightarrow{\imath}}\right) \leq \frac{\pi}{2}$ (see figure 9-a) where $(\widehat{\overrightarrow{u}_1, \overrightarrow{u}_2})$ stands for the angle between the two vectors $\overrightarrow{u}_1$ and $\overrightarrow{u}_2$.
- $B \in \mathcal{B}_2 \Leftrightarrow \overrightarrow{M_1M_2} = \lambda\overrightarrow{\imath}$, $\lambda \in \mathbb{R}^-_*$ and $\left(\widehat{-\overrightarrow{\imath}, \overrightarrow{M_0M_1}}\right) \leq \frac{\pi}{2}$ (see figure 9-b).

Now, let's present the way to stretch curves in $\mathcal{B}_1$ and $\mathcal{B}_2$. Let $B_1 = [M_{10}, M_{11}, M_{12}]$ be the quadratic Bézier curve with control points $M_{10} = (x_{10}, y_{10})$, $M_{11} = (x_{11}, y_{11})$ and $M_{12} = (x_{12}, y_{12})$ and $B_2 = [M_{20}, M_{21}, M_{22}]$ a stretched version of B_1 with a horizontal stretching value h and a vertical one v. We have :

- if $B_1 \in \mathcal{B}_1$ then $M_{20} = (x_{10}, y_{10} - v)$, $M_{22} = (x_{12} - h, y_{12})$ and $M_{21} =$

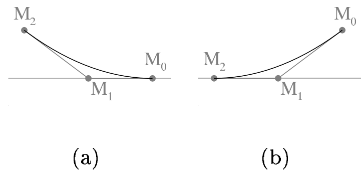


Figure 9. a) Quadratic curve of type 1, b) Quadratic curve of type 2.

$\left(x_{12} - h + \frac{(y_{10}-y_{12}-v)(x_{11}-x_{12})}{y_{11}-y_{12}}, y_{10} - v\right)$.

- if $B_1 \in \mathcal{B}_2$ then $M_{20} = (x_{10}, y_{10})$, $M_{22} = (x_{12} - h, y_{12} - v)$ and $M_{21} = \left(x_{10} + \frac{(y_{12}-y_{10}-v)(x_{11}-x_{10})}{y_{11}-y_{10}}, y_{12} - v\right)$.

In particular, in the font, the vertical stretching value v depends on h. Let $v = \frac{V}{H}h$. Then, V and H stand for the maximal values of the vertical and horizontal stretchings respectively. We can remark that the type of the curve (type 1 or 2) to stretch is preserved after stretching. An example of stretching for the two types are presented on the figure 10.

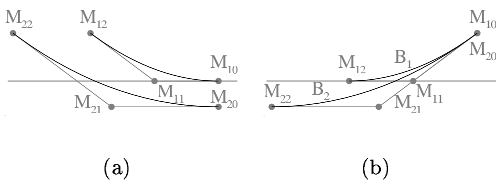


Figure 10. a) Example of stretching a quadratic curve of type 1, b) Example of stretching a quadratic curve of type 2.

In [Bayar], a model based on a cubic Bézier curve has been used easily since we hadn't to deal with the letters outlines. Now, in order to simplify the outlines computing, we used quadratic curves instead of cubic Bézier curves. There would be no loss of visual accuracy since the connections between letters have curvature vectors with small magnitudes.
The keshideh, the stretchable part in a word, is a juxtaposition of a curve in $\mathcal{B}_1$ with an other curve in $\mathcal{B}_2$. When the keshideh has to stretch of h horizontally and v vertically then each of the the components (in $\mathcal{B}_1$ and in $\mathcal{B}_2$) has to stretch of $\frac{h}{2}$ horizontally and v vertically.

Tolerance - Digital Equivalence

In order to implement a font in stretchable outlines, we use approximations. So, we need to consider some precisions or tolerances. In this subsection, we define the notions of digital equivalence and tolerance.

Definition 4 (Digital Equivalence) *Let $M_1 = (x_1, y_1)$ and $M_2 = (x_2, y_2)$ be two points in $\mathbb{Z}^2$. M_1 is digitally equivalent to M_2 if and only if $|x_1 - x_2| \leq 1$ and $|y_1 - y_2| \leq 1$. The digital equivalence is then denoted by $M_1 \equiv M_2$.*

So, for the visual aspect, a point in $\mathbb{Z}^2$ and points in his direct neighbor would then be considered as the same.
To each point in $\mathbb{R}^2$ to draw corresponds an image in $\mathbb{Z}^2$ by a function denoted $\mathcal{N}$. The tolerance of a Bézier curve can then be defined as follows :

Definition 5 (Tolerance of a Bézier curve) *Let B be a Bézier curve and $\tau = \frac{1}{n}$, $n \in \mathbb{N}^*$. The value τ is a tolerance of B if and only if for all t_1 and t_2 in $[0, 1]$ we have :*
$|t_1 - t_2| \leq \tau \Longrightarrow \mathcal{N}(B(t_1)) \equiv \mathcal{N}(B(t_2))$.

The notions of digital equivalence and the tolerance are tools toward the formalization of the visual accuracy. We can define the notion of the optimal tolerance.

Definition 6 (Optimal Tolerance) *Let B be a Bézier curve. The optimal tolerance of B is the greater element of the set $\{\frac{1}{m}/m \in \mathbb{N}*,$ where $\frac{1}{m}$ is a tolerance of $B\}$*

We have the property :

Property 1 *Let us consider the Bézier curves B_1, B_2 and B_3. Let τ_1, τ_2 and τ_3 be their optimal tolerances respectively, then we have the properties :*

- *if B_2 is a stretched version of B_1 then $\tau_2 \leq \tau_1$,*
- *if B_2 and B_3 are curves obtained from B_1 by a Bézier refinement according to a parameter t in $[0, 1]$, then we have $\tau_2 \geq \tau_1$ and $\tau_3 \geq \tau_1$.*

Letter Outlines Computing

3.3.1 Static Letters

When a letter is static, it is directly implemented in the font with the true outline drawn by the qalam's motion (Outlines without overlapping, see figure 11). The true outline is determined with a mean external to the font applying the mathematical notions studied in the paragraph 2.

 ISBN 978-86943-15-2

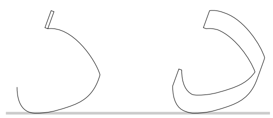


Figure 11. The Qalam's Motion and the corresponding true outlines of the standalone DEL.

3.3.2 Stretchable Letters

The outlines, in the case of variable sized letters, can be implemented in PostScript exactly as in the following example of the letter ر. For the static part of the letter ر, a non closed outline without overlapping is implemented in the font (see figure 12).The outline is then determined out of the font.

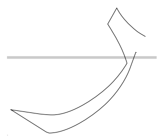


Figure 12. Non closed outline of the static part of End REH.

What about the preceding connection of the letter ر? The preceding connection of ر is a curve in $\mathcal{B}_1$ parametrized with a stretching amount h. The motion of the nib's head is shown on the figure 13-a. Let's denote B^h the trajectory of the motion's origin. For all values of h, the scalar function $\mathcal{R}_{(B^h,\overrightarrow{u}_3)}$ have an extrema on $]0,1[$ (note that $\overrightarrow{u}_2 - \overrightarrow{u}_1 = \overrightarrow{u}_3$). Let t^h be this extrema (the h here is not an exponent. It only means that t depends on h). The direct orthogonal ranges $\mathcal{R}_{(B,\overrightarrow{u}_1-\overrightarrow{u}_0)}$ and $\mathcal{R}_{(B,\overrightarrow{u}_3-\overrightarrow{u}_2)}$ are strictly monotone on $[0,1]$. So, the motions on $\left[0,t^h\right]$ and $\left[t^h,1\right]$ are normal. Consider the following notations :

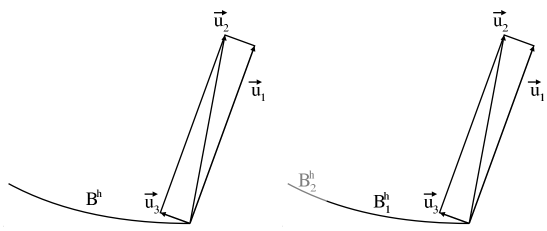


Figure 13. Motion of the stretchable part of End REH.

- t^h is the extrema of $\mathcal{R}_{(B^h,\overrightarrow{u}_3)}$ on $]0,1[$,
- B_1^h and B_2^h are the curves obtained by the decomposition of B^h applying the generalized Bézier algorithm of refinement [Bar85] with respect to t^h (see figure 13-b),
- $B_{1i}^h = t_{\overrightarrow{u}_i}\left(B_1^h\right)$, $i \in \{0,1,2,3\}$ ($\overrightarrow{u}_0 = \overrightarrow{0}$) and
- $B_{2i}^h = t_{\overrightarrow{u}_i}\left(B_2^h\right)$, $i \in \{0,1,2,3\}$.

Then, we have the results :

- On $\left[0,t^h\right]$ the maximal curve is B_{12}^h and the minimal one is B_{10}^h and
- On $\left[t^h,1\right]$ the maximal curve is B_{21}^h and the minimal one is B_{23}^h.

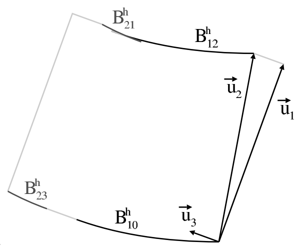


Figure 14. Overlapping of REH preceding connection.

These results are illustrated on the figure 14 with $h = 0$.
The true outline of the preceding connection can't be directly determined because there is an intersection between B_{12}^h and B_{21}^h. Note that this intersection is unique. Let r^h and s^h be the two variables on $[0,1]$ characterizing the intersection coefficients of B_{12}^h and B_{21}^h respectively. Their values are solutions of the equation 1.

$$B_{12}^h(r) - B_{21}^h(s) = (0,0) \qquad (1)$$

The values of t^h, r^h and s^h change as h changes. The values are to be computed dynamically in the font. As B^h is a quadratic curve, the range $\mathcal{R}'_{(B^h,\overrightarrow{u}_3)}(t)$ is a linear function of t. Then, it is easy to determine explicitly t^h in function of h. The formula is given on the equation 2. It was in order to get an easy and explicit determination of t^h that we were to represent the stretchable parts in letters with quadratic curves instead of cubic ones. The function t^h is a monotone increasing function of h.

$$t^h = \frac{47.79081298 + h}{63.67851945 + 1.0887286138 \cdot h} \qquad (2)$$

In general, the determination of the intersection between two quadratic or cubic Bézier curves is done through iterations (there is no explicit solution). Since the PostScript procedure to produce a character, in a dynamic font, is repeated whenever the letter is to draw, the determination of the

 ISBN 978-86943-15-2

curves intersections coefficients in the procedure, applying an iterative algorithm is of a high cost. So, we have to find out a way to minimize the time necessary to compute r^h and s^h.
Let us consider once again the notations :

- τ_{12}^h the tolerance of B_{12}^h depending on h.
- τ_{21}^h the tolerance of B_{21}^h depending on h.

In the Naskh style, the stretching variable h can vary from 0 to H, where H is 6 diacritic points (12 dp for all the keshideh and 6 dp for each of its components). Consider a font in 1000 points size, $H = 580.829699$. The determination of the coefficients r^h and s^h would be based on approximations as follows.
The system Maple can be used to get the following properties :

Property 2

- *The function r^h is monotone increasing of h on $[0, H]$,* *(1)*
- *The function s^h is monotone decreasing of h on $[0, H]$,* *(2)*
- *The functions τ_{12}^h and τ_{21}^h are monotone decreasing on $[0, H]$,* *(3)*
- *and $\tau_{12}^h \leq \tau_{21}^h$ on $[0, H]$. This means that for $h \in [0, H]$, τ_{12}^h is an optimal tolerance of B_{12}^h and a tolerance of B_{21}^h.* *(4)*

Now, we have all the properties necessary to build the function giving the approximated values of r^h and s^h.

Let $r_{min} = r^0$, $r_{max} = r^H$. Consider the sequence $R = (r_i)_{i=0,\ldots,n}$ such that $r_i = r_{min} + \frac{r_{max}-r_{min}}{n} i$, where $n = \left[\frac{r_{max}-r_{min}}{\tau_{12}^H}\right] + 1$. Let $[x]$ be the integer part of x. Let's also consider the two sequences $\mathcal{H} = (h)_{i=0,\ldots,n}$ and $S = (s_i)_{i=0,\ldots,n}$ such that $\{h_i, s_i\}_{i=0,\ldots,n}$ are solutions of the equation 1 where h and s are the unknown and r is given taking r_i as value. The system Maple can help to solve all the equations. In some cases, some of these equations can require some minutes when B^h is quadratic and even some hours when B^h is cubic. That's an other reason to choose quadratic curves to encode dynamic parts. We get the following results :

- $r_{i+1} - r_i < \tau_{12}^H \ \forall i = 0, \ldots, n-1$,
- $r_{i+1} - r_i < \tau_{12}^h \ \forall i = 0, \ldots, n-1 \ \forall h \in [0, H]$ (according to the property 2-(3)). This means that $\mathcal{N}\left(B_{12}^h(r_i)\right) \equiv \mathcal{N}\left(B_{12}^h(r_{i+1})\right) \ \forall h \in [0, H]$,

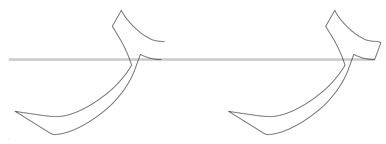


Figure 15. Closed and non closed non overlapping outline.

- $s_i - s_{i+1} < \tau_{21}^h \ \forall i = 0, \ldots, n-1 \ \forall h \in [0, H]$ (according to the property 2-(2 and 4)) . This means that $\mathcal{N}\left(B_{21}^h(s_i)\right) \equiv \mathcal{N}\left(B_{21}^h(s_{i+1})\right) \ \forall h \in [0, H]$.

We have finally an important property that would be useful for the determination of the approximation function, $\forall i \in \{0, \ldots, n-1\}$, $\forall h$, $h_i \leq h \leq h_{i+1}$, we have :

- $r_i \leq r^h \leq r_{i+1}$ and $\mathcal{N}\left(B_{12}^h\left(r^h\right)\right) \equiv \mathcal{N}\left(B_{12}^h(r_i)\right)$ and so, $\mathcal{N}\left(B_{12}^h\left(r^h\right)\right) \equiv \mathcal{N}\left(B_{12}^h(r_{i+1})\right)$.
- $s_{i+1} \leq s^h \leq s_i$ and $\mathcal{N}\left(B_{21}^h\left(s^h\right)\right) \equiv \mathcal{N}\left(B_{21}^h(s_i)\right)$ and so, $\mathcal{N}\left(B_{21}^h\left(s^h\right)\right) \equiv \mathcal{N}\left(B_{21}^h(s_{i+1})\right)$.

Then, we know very well the tolerant values of $B_{12}^h\left(r^h\right)$ and $B_{21}^h\left(s^h\right)$.
Now, consider the sequence $P = (P_i)_{i=0,\ldots,n} = (r_i, s_i)_{i=0,\ldots,n}$. Let's approximate the set of points P_i by a quadratic Bézier curve A such that $A(0) = P_0 = (r_0, s_0)$ and $A(1) = P_n = (r_n, s_n)$. In order to approximate a set of points $\{P_0, \ldots, P_n\}$ by a curve parametrized on $[0, 1]$, we may associate to every point P_i a coefficient t_i in $[0, 1]$ such that the sequence $(t_i)_{i=0,\ldots,n}$ is strictly monotone increasing and $t_0 = 0$ and $t_n = 1$.
First, let us determine $(t_i)_{i=0,\ldots,n}$.
Recall that t^h (defined in the equation 2) is a monotone increasing function of h.
Since r^h is monotone increasing, the sequence $(h_i)_{i=0,\ldots,n}$ deduced through the sequence $(r_i)_{i=0,\ldots,n}$ is monotone increasing. Consider $(t_i)_{i=0,\ldots,n}$ such that $t_i = \frac{t^{h_i} - t^0}{t^H - t^0}$.
$(t_i)_{i=0,\ldots,n}$ is a strictly monotone increasing sequence with $t_0 = 0$ and $t_n = 1$.
Consider that the quadratic Bézier curve to be determined has the control points (r_0, s_0), (x_1, y_1) and (r_n, s_n). We have to determine (x_1, y_1) in order to minimize the value D (see equation 3).

$$D = \sum_{i=0}^{n} \left(P_i - A(t_i)\right)^2 \qquad (3)$$

D can be minimal if the following mandatory conditions are satisfied :

 ISBN 978-86943-15-2

- $\frac{\partial D}{\partial x_1} = 0$ and
- $\frac{\partial D}{\partial y_1} = 0$.

$\frac{\partial D}{\partial x_1} = 0$ implies that $x_1 = \frac{\sum_{i=0}^{n} t_i(1-t_i)\left(r_i-(1-t_i)^2 r_0 - t_i^2 r_n\right)}{2\sum_{i=0}^{n} t_i^2(1-t_i)^2}$ and $\frac{\partial D}{\partial y_1} = 0$ implies that $y_1 = \frac{\sum_{i=0}^{n} t_i(1-t_i)\left(s_i-(1-t_i)^2 s_0 - t_i^2 s_n\right)}{2\sum_{i=0}^{n} t_i^2(1-t_i)^2}$.
Suppose that $A(t) = (r(t), s(t))$. Then A is a good approximation if the condition in the formula 4 is satisfied

$$|r(t_i) - r_i| < \tau_{12}^{H} \ \forall i = 0, \ldots, n \qquad (4)$$

If the condition 4 isn't satisfied the sequences $\mathcal{H}$ and P are subdivided in two pairs of sub-sequences $\left\{(h_i)_{i=0,\ldots,k}, (P_i)_{i=0,\ldots,k}\right\}$ and $\left\{(h_i)_{i=k,\ldots,n}, (P_i)_{i=k,\ldots,n}\right\}$ such that k verify the formula 5 and the process is repeated.

$$|r(t_k) - r_k| = \max_{i=0}^{n} |r(t_i) - r_i| \ \ \forall i = 0, \ldots, n \quad (5)$$

Concretely, for the preceding connection of the letter ر, the determination of the intersection coefficients are computed with the evaluation of a curve A_1 where h is in $[0, 179.595499[$ and another curve A_2 if h is in $[179.595499, 580.818536]$.
The approximation has been tested with arbitrary values of h and we got intersection coefficients with respect of the curves tolerances (the details are on the annexed file).
With these mechanism, we can encode the preceding connection to get ر with closed outlines and therefore to fill the letter and also a non closed outline to be used to write words in outlines (see figure 15).

4. CONCLUSIONS

This is a way to solve the problem of drawing stretchable, variable sized Arabic letters in outlines. This problem has been posed many years ago. Of course, only the Naskh style, where the nib's head is in constant inclination, has been studied. This approach will be developed in a coming paper, to styles where the nib's head can be in translation and rotation at the same time. That is of a big importance for writing Arabic alphabet based texts respecting the basic calligraphic rules. It can also help to compose calligraphic mathematical formulas.

5. REFERENCES

[Lam85] Leslie Lamport, LaTeX-A Document Preparation System, Reading MA: Addison Wesley, 1985.

[Laz01] Azzeddine Lazrek, A package for typesetting Arabic mathematical formulas, Die TeXnische Komodie, DANTE e.V., Vol.13, No. 2, pp. 54–66, 2001.

[Laz03] Azzeddine Lazrek, CurExt, Typesetting variable-sized curved symbols, EuroTeX 2003 Pre-prints : 14th. European TeX conference (Brest, France), pp. 47-71, 2003.

[Ado99] Adobe. PostScript Language Reference Manual, Library of Congress Cataloging-in-Publication Data, Third edition, Adobe Systems Incorporated, 1999.

[Wei92] D. Weise, D. Adler, TrueType and Microsoft Windows 3.1, Technical Report, Microsoft Corporation, Redmond, WA, 1992.

[Knu86] D.E. Knuth, The MetafontBook, Computers and Typesetting, Reading MA: Addison-Wesley, Vol.C, 1986.

[Ber99] Daniel M. Berry, Stretching Letter and Slanted-Baseline Formatting for Arabic, Hebrew and Persian with ditroff/ffortid and Dynamic PostScript Fonts, Software-Practice and Experience, Vol. 29, No. 15, pp.1417-1457, 1999.

[Bayar] A. Bayar and K. Sami, How a font can respect rules of Arabic calligraphy (submitted).

[Bez77] Pierre E. Bézier, Essai de Définition Numérique des Courbes et des Surfaces Expérimentales, Ph.D. dissertation, Université Pierre et Marie Curie, Paris, 1977.

[Bar85] Brian A. Barsky, Arbitrary Subdivision of Bézier Curves, Technical Report UCB.CSD 85/265, Computer Science Division, University of California, Berkeley, California 94720, November, 1985.

[Lag92] Klaus Lagally, ArabTeX - Typesetting Arabic with Vowels and Ligatures, EUROTeX'92, Prague, 1992.