

Arc Length of Rational Bézier Curves and Use for CAD Reparametrization

Maharavo Randrianarivony

Abstract—The length Λ of a given rational Bézier curve is efficiently estimated. Since a rational Bézier function is nonlinear, it is usually impossible to evaluate its length exactly. The length is approximated by using subdivision and the accuracy of the approximation Λ_n is investigated. In order to improve the efficiency, adaptivity is used with some length estimator. A rigorous theoretical analysis of the rate of convergence of Λ_n to Λ is given. The required number of subdivisions to attain a prescribed accuracy is also analyzed. An application to CAD parametrization is briefly described. Numerical results are reported to supplement the theory.

Keywords—Adaptivity, Length, Parametrization, Rational Bézier.

I. INTRODUCTION

RATIONAL Bézier curves are important CAD entities in computational geometry because they can represent both the free-form setting and the algebraic one. Thus, they can describe circular arcs and most interesting conic sections. On the other hand, free-form Bézier curves are special case of them. The main contribution in this paper is as follows:

- Algorithm for length estimation of such curves,
- Theoretical investigation using subdivisions and bounds,
- Exponential convergence speed $\mathcal{O}(2^{-n})$,
- Practical computer implementation of the theory.

Related works are as follows. Roulier has proposed a length estimation algorithm but only for Bézier curves [11]. Walter *et al.* did not really evaluate lengths but they have approximated the arc length parametrization which is a very closely related task. A similar approach was proposed by Floater who used cubic spline for the approximation [5]. Subdivision technique was used by Hain who proposed some approach to stop the subdivision recursion [6]. In this paper, subdivisions are also used but for the rational case. The structure of this paper is as follows. It starts by formulating the problem more accurately in the next section. The main result of this paper is found in section III where the approximation method is introduced and the error is analyzed. A possible improvement of the method by using adaptivity is found in section IV. Section V will be devoted to a brief application in CAD parametrizations. Finally, numerical results are shown at the end of the paper.

M. Randrianarivony is with the Faculty for Mathematics, University of Bonn, Germany, email: maharavo@informatik.tu-chemnitz.de.

II. PROBLEM FORMULATION

The objective is to design an algorithm for estimating the length of a curve \mathbf{x} inside an interval $[a, b]$. That is, the following expression should be evaluated

$$\Lambda := \int_a^b \|\mathbf{x}'(t)\| dt. \quad (1)$$

Without loss of generality, it is assumed that the curve is defined on $[0, 1]$ (i.e. $a = 0$, $b = 1$) and the whole length is computed. The general case where $[a, b] \neq [0, 1]$ can be treated in a very similar way.

The curve is supposed to be a rational Bézier curve

$$\mathbf{x}(t) := \frac{\sum_{i=0}^m \omega_i \mathbf{b}_i B_i^m(t)}{\sum_{i=0}^m \omega_i B_i^m(t)}, \quad (2)$$

where B_i^m denotes the Bernstein polynomial [3], [2] and $\mathbf{b}_i = [b_{i,1}, b_{i,2}, b_{i,3}] \in \mathbb{R}^3$ are the control points. Additionally, the weights ω_i are assumed to be uniformly bounded. That is, there exist two positive constants R_1, R_2 such that

$$R_1 < \left| \sum_{i=0}^m \omega_i B_i^m(t) \right| < R_2 \quad \forall t \in [0, 1]. \quad (3)$$

It is denoted by $\tilde{\mathbf{x}}(t) = [\tilde{x}_1(t), \tilde{x}_2(t), \tilde{x}_3(t)]$ and $\omega(t)$ the numerator and the denominator of formula (2) where $\mathbf{x}(t) = [x_1(t), x_2(t), x_3(t)]$. The numerator $\tilde{\mathbf{x}}$ is a Bézier curve where its control points are given by $\tilde{\mathbf{b}}_i := \omega_i \mathbf{b}_i$.

Since the expression in (2) contains rational quotient and the one in (1) has square root and derivatives such as

$$\Lambda = \int_0^1 \sqrt{x_1'(t)^2 + x_2'(t)^2 + x_3'(t)^2} dt, \quad (4)$$

it is very difficult to compute the integral exactly. In fact, the integrand is given by

$$\frac{1}{\omega(t)^2} \sqrt{\sum_{j=1}^3 [\tilde{x}_j'(t)\omega(t) - \omega'(t)\tilde{x}_j(t)]^2}. \quad (5)$$

For the same reason, traditional methods using polynomial approximation of the integrand would require too high polynomial degree. Hence, geometric methods are used for the approximation.

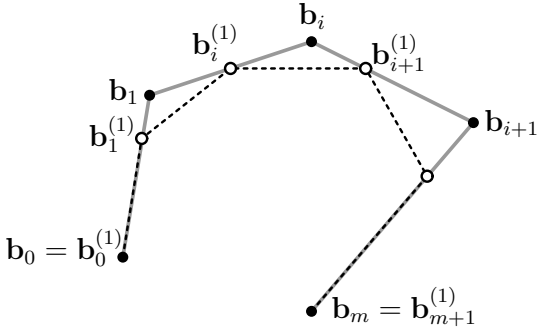


Fig. 1. Degree elevation in which $L[b_i^{(1)}, b_i, b_{i+1}^{(1)}] \leq L[b_i^{(1)}, b_{i+1}^{(1)}]$.

III. APPROXIMATION AND ITS ACCURACY

In this section, the exact length Λ will be approximated by a sequence Λ_n and the error $|\Lambda - \Lambda_n|$ is analyzed. Before going into the technical details, a short motivation about the approach is given now. The value of Λ_n is computed by finding a lower bound \mathcal{L}_n and an upper bound \mathcal{U}_n such that

$$\mathcal{L}_n \leq \Lambda \leq \mathcal{U}_n \quad \text{for every } n \geq 0. \quad (6)$$

If those bounds have the property that their difference $|\mathcal{L}_n - \mathcal{U}_n|$ converges to zero, then a good choice for the approximation is their average $\Lambda_n := 0.5(\mathcal{L}_n + \mathcal{U}_n)$ or any other convex combination:

$$\Lambda_n := \alpha \mathcal{L}_n + (1 - \alpha) \mathcal{U}_n \quad \text{for } 0 < \alpha < 1. \quad (7)$$

Thus, the speed of convergence of Λ_n to Λ depends on that of the decay $|\mathcal{L}_n - \mathcal{U}_n|$ of the bounds. The main advantage of this method is that the bounds \mathcal{L}_n and \mathcal{U}_n can be computed algorithmically and the rate of convergence is exponential.

Quadrature rules will not be used to estimate the integral in (1) because the structure of the function \mathbf{x} is known [11], [13]. The preferred method here is to apply subdivision recursively while using some flatness criterion [6], [4] in order to know if the curve is close to be linear.

A. Preliminary results

Before stating the main theorem, consider the following simple lemma. At first glance the lemma seems evident because of the famous convex hull property [3]. But a closer look reveals that the convex hull property alone cannot justify the claim, especially if it takes the weights into consideration. The lemma is proved by using rational degree elevation. Note that the degree elevation is not used in practice for that it is exclusively for proving purpose. Before going any further, note that the following bound is not yet the upper bound \mathcal{U}_n which is sought.

Lemma 1: For any rational Bézier curve of the form (2), its length is smaller than

$$\sum_{i=0}^{m-1} \|b_i - b_{i+1}\|. \quad (8)$$

Proof: For a finite sequence of 3D points $\mathbf{P} = \{\mathbf{p}_i\}_{i=0}^n$, denote

$$L[\mathbf{P}] := L[\mathbf{p}_0, \dots, \mathbf{p}_n] := \sum_{i=0}^{n-1} \|\mathbf{p}_i - \mathbf{p}_{i+1}\|. \quad (9)$$

The degree elevated rational Bézier curve of \mathbf{x} is given [3] by

$$\mathbf{x}(t) = \frac{\sum_{i=0}^{m+1} \omega_i^{(1)} b_i^{(1)} B_i^m(t)}{\sum_{i=0}^m \omega_i^{(1)} B_i^m(t)}, \quad (10)$$

where the new weights are

$$\omega_i^{(1)} := c_{i,m} \omega_{i-1} + (1 - c_{i,m}) \omega_i \quad \text{with } c_{i,m} := i/(m+1),$$

and the new control points are

$$b_i^{(1)} := \frac{c_{i,m} \omega_{i-1} b_{i-1} + (1 - c_{i,m}) \omega_i b_i}{c_{i,m} \omega_{i-1} + (1 - c_{i,m}) \omega_i}. \quad (11)$$

Thus, $b_i^{(1)}$ is a convex combination of b_{i-1} and b_i because the weights are positive. Therefore, it yields (see Fig. 1)

$$L[b_i^{(1)}, b_{i+1}^{(1)}] \leq L[b_i^{(1)}, b_i, b_{i+1}^{(1)}]. \quad (12)$$

Denote by $\mathbf{B}^{(0)}$ the initial control polygon and by $\mathbf{B}^{(p)}$ ($p \geq 1$) the next control polygons after repeated degree elevations. Since $b_i^{(1)}$ is a convex combination of b_{i-1} and b_i , it gives

$$L[\mathbf{B}^{(0)}] = L[b_0, b_1^{(1)}, b_1, b_2^{(1)}, b_2, \dots, b_{m-1}, b_m^{(1)}, b_m].$$

Relation (12) produces $L[\mathbf{B}^{(1)}] = L[b_0^{(1)}, \dots, b_{m+1}^{(1)}] \leq L[\mathbf{B}^{(0)}]$. A repeated application of that gives

$$L[\mathbf{B}^{(p)}] \leq L[\mathbf{B}^{(p-1)}] \leq \dots \leq L[\mathbf{B}^{(1)}] \leq L[\mathbf{B}^{(0)}]. \quad (13)$$

Since it is well known [3] that the control polygon of the curve tends to the curve itself, it gives $\Lambda = L[\mathbf{B}^{(\infty)}] \leq L[\mathbf{B}^{(0)}]$. ■

B. Rational Bézier subdivision

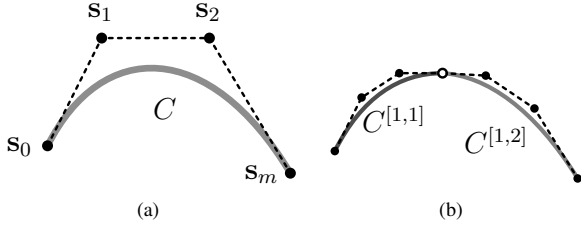
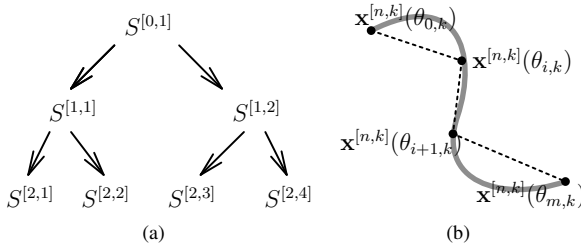
First, some notions related to successive subdivisions [7] of an arbitrary Bézier function C are recalled:

$$C(t) = \sum_{i=0}^m s_i B_i^m(t). \quad (14)$$

Let $s_i^{(j)}$ be the points which are found by using the de Casteljau [3] algorithm at $t = 0.5$, i.e. $s_i^{(j+1)} := 0.5(s_i^{(j)} + s_{i+1}^{(j)})$ and $s_i^{(0)} := s_i$. The function $C^{[0,1]} := C$ can be split into two Bézier functions $C^{[1,1]}$ and $C^{[1,2]}$ (see Fig. 2) which have respectively the control points $s_i^{[1,1]} := s_0^{(1)}$ and $s_i^{[1,2]} := s_i^{(m-i)}$ and such that

$$C^{[0,1]}(t) = \begin{cases} C^{[1,1]}(t) & \forall t \in [0, 0.5], \\ C^{[1,2]}(t) & \forall t \in [0.5, 1]. \end{cases} \quad (15)$$

That process can be applied successively in order to obtain from each Bézier function $C^{[p-1,i]}$ two Bézier functions $C^{[p,2i-1]}$ and $C^{[p,2i]}$. That is, after applying subdivisions n times, the curves $C^{[n,1]}, C^{[n,2]}, \dots, C^{[n,2^n]}$ are obtained as


 Fig. 2. Subdividing a given Bézier curve: the breakpoint is at $t = 0.5$.

 Fig. 3. (a) Recursive subdivisions (b) Local lower bound from $\mathbf{x}^{[n,k]}(\theta_{i,k})$.

explained in Fig. 3(a). Each function $C^{[n,k]}$ coincides with C on the interval $[p_{k-1}, p_k]$ where $p_k := k/2^n$ and its control points are denoted by $\mathbf{s}_i^{[n,k]}$ for $k = 1, \dots, 2^n$ and $i = 0, \dots, m$.

Now, the above subdivision technique is applied to the numerator and denominator. The functions $\tilde{\mathbf{x}}(\cdot)$ and $\omega(\cdot)$ will be subdivided into functions $\tilde{\mathbf{x}}^{[n,k]}$ and $\omega^{[n,k]}$ having the control points $\tilde{\mathbf{b}}_i^{[n,k]}$ and $\omega_i^{[n,k]}$. On each subinterval $[p_k, p_{k+1}]$, the rational Bézier $\mathbf{x}^{[n,k]} := \tilde{\mathbf{x}}^{[n,k]}/\omega^{[n,k]}$ is used. Thus, defining $\mathbf{b}_i^{[n,k]} := \tilde{\mathbf{b}}_i^{[n,k]}/\omega_i^{[n,k]}$ gives for all $\tau \in [p_k, p_{k+1}]$:

$$\mathbf{x}^{[n,k]}(\tau) = \frac{\sum_{i=0}^m \omega_i^{[n,k]} \mathbf{b}_i^{[n,k]} B_i^m(s)}{\sum_{i=0}^m \omega_i^{[n,k]} B_i^m(s)} \quad \text{with} \quad s = \frac{\tau - p_k}{p_{k+1} - p_k}. \quad (16)$$

Furthermore, the next restriction property is valid:

$$\tilde{\mathbf{x}}^{[n,k]} = \tilde{\mathbf{x}}|_{[p_{k-1}, p_k]}, \quad \text{and} \quad \omega^{[n,k]} = \omega|_{[p_{k-1}, p_k]}. \quad (17)$$

By considering the interval $[p_{k-1}, p_k]$, the next expression is introduced for $i = 0, \dots, m$

$$\theta_{i,k} := (i/m)p_k + (1-i/m)p_{k-1} \quad \text{with} \quad p_k = k/2^n. \quad (18)$$

Theorem 1: Suppose that the rational Bézier in (2) has been subdivided n times. Then, the following accuracy order is valid for all $k = 1, \dots, 2^n$ and $i = 0, \dots, m$:

$$\|\mathbf{x}^{[n,k]}(\theta_{i,k}) - \mathbf{b}_i^{[n,k]}\| = \mathcal{O}(2^{-2n}). \quad (19)$$

Proof: The boundedness (3) and the restriction property (17) yield $|\omega^{[n,k]}(t)| = |\omega(t)| > R_1$. Hence, there exists K_1 such that

$$\begin{aligned} \left\| \mathbf{x}^{[n,k]}(\theta_{i,k}) - \frac{\omega_i^{[n,k]} \mathbf{b}_i^{[n,k]}}{\omega^{[n,k]}(\theta_{i,k})} \right\| &= \left\| \frac{\tilde{\mathbf{x}}^{[n,k]}(\theta_{i,k}) - \omega_i^{[n,k]} \mathbf{b}_i^{[n,k]}}{\omega^{[n,k]}(\theta_{i,k})} \right\| \\ &\leq K_1 \left\| \tilde{\mathbf{x}}^{[n,k]}(\theta_{i,k}) - \omega_i^{[n,k]} \mathbf{b}_i^{[n,k]} \right\|. \end{aligned} \quad (20)$$

Similarly,

$$\left\| \mathbf{b}_i^{[n,k]} - \frac{\omega_i^{[n,k]} \mathbf{b}_i^{[n,k]}}{\omega^{[n,k]}(\theta_{i,k})} \right\| \leq K_2 \left| \omega^{[n,k]}(\theta_{i,k}) - \omega_i^{[n,k]} \right|. \quad (21)$$

As a consequence, it gives

$$\begin{aligned} \|\mathbf{x}^{[n,k]}(\theta_{i,k}) - \mathbf{b}_i^{[n,k]}\| &\leq K_1 \left\| \tilde{\mathbf{x}}^{[n,k]}(\theta_{i,k}) - \omega_i^{[n,k]} \mathbf{b}_i^{[n,k]} \right\| + \\ &K_2 \left| \omega^{[n,k]}(\theta_{i,k}) - \omega_i^{[n,k]} \right|. \end{aligned} \quad (22)$$

On the other hand, consider the blossom function \mathcal{P} of the polynomial $\tilde{\mathbf{x}}^{[n,k]}$. A relation with the control points [12] is obtained:

$$\tilde{\mathbf{b}}_i^{[n,k]} = \mathcal{P}(\underbrace{p_{k-1}, \dots, p_{k-1}}_{m-i}, \underbrace{p_k, \dots, p_k}_i). \quad (23)$$

Thus, the following Taylor development holds:

$$\begin{aligned} \tilde{\mathbf{b}}_i^{[n,k]} &= \mathcal{P}(\theta_{i,k}, \dots, \theta_{i,k}) + \\ &\sum_{p=1}^{m-i} (p_{k-1} - \theta_{i,k}) \frac{\partial}{\partial x_p} \mathcal{P}(\theta_{i,k}, \dots, \theta_{i,k}) + \\ &\sum_{p=m-i+1}^m (p_k - \theta_{i,k}) \frac{\partial}{\partial x_p} \mathcal{P}(\theta_{i,k}, \dots, \theta_{i,k}) + \\ &\mathcal{O}(|p_k - p_{k-1}|^2). \end{aligned}$$

Since \mathcal{P} is symmetric, all partial derivatives in the above relation are the same. The fact that $(m-i)(p_k - \theta_{i,k}) + i(p_{k-1} - \theta_{i,k}) = 0$ induces $\tilde{\mathbf{b}}_i^{[n,k]} = \mathcal{P}(\theta_{i,k}, \dots, \theta_{i,k}) + \mathcal{O}(|p_k - p_{k-1}|^2)$. As a consequence, $\tilde{\mathbf{b}}_i^{[n,k]} = \tilde{\mathbf{x}}^{[n,k]}(\theta_{i,k}) + \mathcal{O}(2^{-2n})$. The same analysis can be repeated to the blossom of the polynomial ω in order to obtain $\omega_i^{[n,k]} = \omega(\theta_{i,k}) + \mathcal{O}(2^{-2n})$. Therefore, it is deduced from (22) that $\|\mathbf{x}^{[n,k]}(\theta_{i,k}) - \mathbf{b}_i^{[n,k]}\| = \mathcal{O}(2^{-2n})$. ■

C. Upper and Lower Bounds

At the n -th subdivision, the true length Λ is the sum of the lengths $\lambda(k, n)$ of the subcurves $\mathbf{x}^{[n,k]}$ such as

$$\Lambda = \sum_k \lambda(k, n). \quad (24)$$

The approximation result (19) can now be used to deduce the accuracy in length computation.

Theorem 2: Define for all $k = 0, \dots, 2^n - 1$

$$l(k, n) := \sum_{i=0}^{m-1} \|\mathbf{x}^{[n,k]}(\theta_{i,k}) - \mathbf{x}^{[n,k]}(\theta_{i+1,k})\|, \quad (25)$$

$$u(k, n) := \sum_{i=0}^{m-1} \|\mathbf{b}_i^{[n,k]} - \mathbf{b}_{i+1}^{[n,k]}\|. \quad (26)$$

For any $\alpha \in]0, 1[$, the sequence $\Lambda_n := \sum_k (\alpha l(k, n) + (1 - \alpha)u(k, n))$ converges to the exact length Λ in dyadic order:

$$|\Lambda - \Lambda_n| = \mathcal{O}(2^{-n}). \quad (27)$$

Proof: The length $\lambda(k, n)$ of the curve $\mathbf{x}^{[n,k]}$ verifies the next relation

$$l(k, n) \leq \lambda(k, n) \leq u(k, n), \quad (28)$$

where the second inequality is due to the preceding Lemma and the first one is obvious (Fig. 3(b)). On the other hand, the difference $D(k, n) := |u(k, n) - l(k, n)|$ of those local bounds can be estimated as follows

$$\begin{aligned} D(k, n) &= \sum_{i=0}^{m-1} \|\mathbf{b}_i^{[n,k]} - \mathbf{b}_{i+1}^{[n,k]}\| - \\ &\quad \|\mathbf{x}^{[n,k]}(\theta_{i,k}) - \mathbf{x}^{[n,k]}(\theta_{i+1,k})\| \\ &\leq \sum_{i=0}^{m-1} \|\mathbf{b}_i^{[n,k]} - \mathbf{x}^{[n,k]}(\theta_{i,k})\| - \\ &\quad \|\mathbf{b}_{i+1}^{[n,k]} - \mathbf{x}^{[n,k]}(\theta_{i+1,k})\| + \\ &\quad \|\mathbf{x}^{[n,k]}(\theta_{i,k}) - \mathbf{x}^{[n,k]}(\theta_{i+1,k})\| - \\ &\quad \|\mathbf{x}^{[n,k]}(\theta_{i,k}) - \mathbf{x}^{[n,k]}(\theta_{i+1,k})\| \\ &\leq \sum_{i=0}^{m-1} \|\mathbf{b}_i^{[n,k]} - \mathbf{x}^{[n,k]}(\theta_{i,k})\| + \\ &\quad \|\mathbf{b}_{i+1}^{[n,k]} - \mathbf{x}^{[n,k]}(\theta_{i+1,k})\|. \end{aligned}$$

By using the previous theorem with the last inequality, it is deduced that

$$D(k, n) = |u(k, n) - l(k, n)| = \mathcal{O}(2^{-2n}). \quad (29)$$

As a consequence, it gives $|u(k, n) - \lambda(k, n)| = \mathcal{O}(2^{-2n})$ and $|l(k, n) - \lambda(k, n)| = \mathcal{O}(2^{-2n})$. Hence, the accuracy of the length estimation is given as

$$\begin{aligned} |\Lambda - \Lambda_n| &= \left| \sum_{k=0}^{2^n} \lambda(k, n) - [\alpha l(k, n) + (1 - \alpha)u(k, n)] \right| \\ &\leq \sum_{k=0}^{2^n} |\alpha(\lambda(k, n) - l(k, n)) + \\ &\quad (1 - \alpha)(\lambda(k, n) - u(k, n))| \\ &= 2^n \mathcal{O}(2^{-2n}) = \mathcal{O}(2^{-n}). \end{aligned}$$

By using relation (24), the lower and upper bounds \mathcal{L}_n and \mathcal{U}_n that were introduced in the beginning are

$$\mathcal{L}_n := \sum_k l(k, n) \leq \Lambda \leq \mathcal{U}_n := \sum_k u(k, n). \quad (30)$$

Corollary 1: For any prescribed accuracy $\varepsilon > 0$, the expected number n of subdivisions to have an accuracy $|\Lambda - \Lambda_n| < \varepsilon$ is of order

$$\left\lceil \log_2 \frac{1}{\varepsilon} \right\rceil, \quad (31)$$

where $\lceil x \rceil$ denotes the smallest integer larger than x .

IV. IMPROVEMENT BY USING ADAPTIVITY

In the preceding section, a method which always subdivides each rational Bézier curve into two was developed. In this section, an improvement of that approach is discussed. In fact, it will be shown how to develop an adaptive strategy in order to only apply subdivisions at positions where they

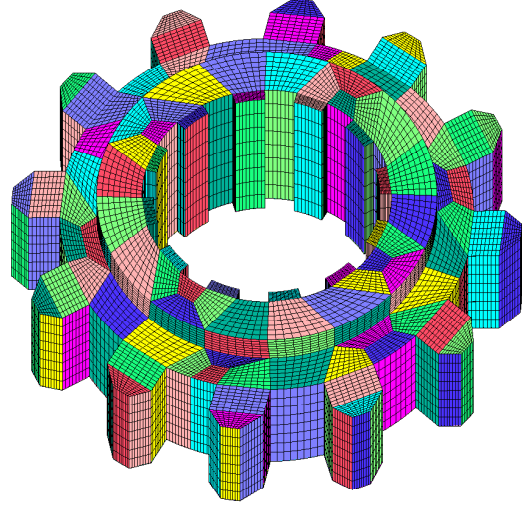


Fig. 4. Globally continuous mappings on some four-sided CAD patches.

are necessary. In practice, when the rational Bézier curve is almost linear, there is no need to subdivide it any more. The goal is then to identify positions where further subdivisions are required without deteriorating the accuracy. As a consequence, a certain metric to quantify the error inside a subcurve is needed. The quantities $l(k, n)$ and $u(k, n)$ of relation (28) are appropriate values for evaluating the flatness of the subcurve $\mathbf{x}^{[n,k]}$. It was proven in (29) that the difference between $l(k, n)$ and $u(k, n)$ decays to zero. That is, subdivisions should be applied only at positions where $D(k, n) = |l(k, n) - u(k, n)|$ is large. One can even devise an adaptive strategy which subdivides only the rational Bézier curves corresponding to

$$D(k, n) \geq \frac{\varepsilon}{2^n}. \quad (32)$$

It is because if $D(k, n) < \varepsilon/2^n$, then all subcurves of $\mathbf{x}^{[n,k]}$ have error smaller than $\varepsilon/2^n$ so that refinement is unnecessary. By doing that, subdivisions are needed only at subintervals where the flatness metric $D(k, n)$ indicates that the local upper bound $l(k, n)$ and lower bound $u(k, n)$ are still very different from one another. That adaptive method is summarized in the next algorithm where a list of subcurves $\mathcal{S} = \{\mathbf{x}_0, \mathbf{x}_1, \dots\}$ is updated. The value $D(k, n)$ of a subcurve $\mathbf{x}_p \in \mathcal{S}$ is denoted by $\text{ESTIM}(\mathbf{x}_p)$.

Algorithm: Adaptive length computation of \mathbf{x}

- 1: Choose accuracy $\varepsilon > 0$.
 - 2: Estimate n by using (31).
 - 3: Initialize the set of subcurves as $\mathcal{S} := \{\mathbf{x}\}$.
 - 4: for $(i = 1, \dots, n)$
 - 5: Find all $\mathbf{x}_p \in \mathcal{S}$ with $\text{ESTIM}(\mathbf{x}_p) \geq \varepsilon/2^n$.
 - 6: Subdivide \mathbf{x}_p and compute $l(k, n)$ and $u(k, n)$.
 - 7: $\Lambda_n := \Lambda_n + 0.5(l(k, n) + u(k, n))$. Update \mathcal{S} .
 - 8: end for
-

V. BRIEF APPLICATION TO CAD PARAMETRIZATION

In some earlier works [8], [9], [10], a model composed of trimmed surfaces [1] should be split into four-sided patches \mathcal{P}_i . Some functions ψ_i were needed such that for $(u, v) \in [0, 1]^2$,

$$\psi_i(u, v) \in \mathbf{R}^3 \quad \text{and} \quad \mathcal{P}_i = \text{Im}(\psi_i). \quad (33)$$

The generation of such mappings was mainly a composition of a base function given by a CAD exchange like IGES and Coons maps [3] which are defined on the unit square $[0, 1]^2$. The main goal was that the mappings are globally continuous. Such a task can be illustrated by Fig. 4. For two incident four-sided patches \mathcal{P}_i and \mathcal{P}_j , the images of u -constant or v -constant isolines of ψ_i and ψ_j should match at the interface. It was proved [10] that if the chord length reparametrization of the boundary curves is used then two adjacent Coons patches verify such matching conditions. That is, each boundary curve κ must be reparametrized into $\tilde{\kappa}$ where $\kappa = \tilde{\kappa} \circ \chi$ in which

$$\chi(t) = \int_a^t \left\| \frac{d\rho}{dt}(\theta) \right\| d\theta \quad (34)$$

where ρ is a well chosen function. The work presented in this paper is important when generating the chord length reparametrization. A complete detail of such a reparametrization using curve length could be found in [10].

VI. NUMERICAL RESULTS

In order to observe the practical efficiency of the former theory, it has been implemented in C/C++. Now, numerical investigation will be shown for the dependence on n of the error $|\Lambda - \Lambda_n|$ and the bounds $\mathcal{L}_n, \mathcal{U}_n$. Thus, a rational Bézier curve is considered in which $m = 3$ and the control points with the corresponding weights are

$$\begin{aligned} \mathbf{b}_0 &= [0.143, 3.021, 2.045], & \omega_0 &= 1.2, \\ \mathbf{b}_1 &= [1.945, 4.192, 2.223], & \omega_1 &= 0.9, \\ \mathbf{b}_2 &= [2.043, 0.012, 2.185], & \omega_2 &= 1.5, \\ \mathbf{b}_3 &= [3.543, 2.078, 2.865], & \omega_3 &= 0.6, \end{aligned}$$

where the expected value of the length is 4.9918739152. A plot of the error in terms of n is depicted in Fig. 5 which confirms the theory. Note that the vertical axis is logarithmically scaled. Additionally, the numerical behavior of the difference of the lower bound \mathcal{L}_n and upper bound \mathcal{U}_n is seen in Table. I which is also conform to the theoretical prediction.

VII. CONCLUSION

A method based on subdivision for estimating the lengths of a rational Bézier curve was presented. A lower bound and an upper bound which are easy to estimate were found such that their difference decays to zero exponentially. Additionally, an adaptive strategy has been devised to locate positions to apply further subdivisions. The theoretical approach has been supported by numerical results and CAD applications.

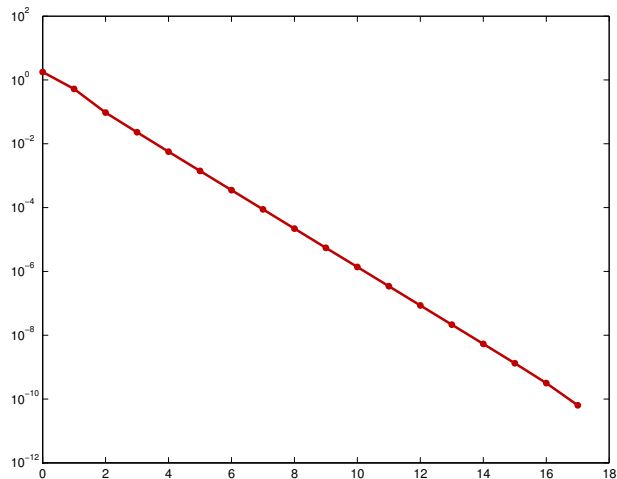


Fig. 5. Logarithmically scaled error $|\Lambda - \Lambda_n|$ in function of n .

TABLE I
BEHAVIOR OF THE UPPER AND LOWER BOUNDS WITH RESPECT TO n .

n	Lower bound \mathcal{L}_n	Upper bound \mathcal{U}_n	Difference $ \mathcal{L}_n - \mathcal{U}_n $
0	4.5703476367	8.9798491325	4.4095e+00
2	4.9693082543	5.2046867676	2.3538e-01
4	4.9904688238	5.0045723724	1.4104e-02
5	4.9915226799	4.9950382924	3.5156e-03
7	4.9918519637	4.9920714912	2.1953e-04
9	4.9918725432	4.9918862629	1.3720e-05
10	4.9918735722	4.9918770021	3.4299e-06
11	4.9918738294	4.9918746869	8.5748e-07
13	4.9918739098	4.9918739634	5.3592e-08
15	4.9918739148	4.9918739182	3.3495e-09
16	4.9918739151	4.9918739159	8.3744e-10

ACKNOWLEDGMENT

The author is very grateful to Prof. Brunnett from the Technical University of Chemnitz who has introduced him to the topic of Geometric Modeling and CAD. He thanks also Prof. Schneider from the University of Berlin for supporting his research at the Institute of Computer Science in the University of Kiel. Last but not least, the author wants to acknowledge the help of Prof. Harbrecht from the University of Bonn who allowed him to pursue his current Post-Doctoral research.

REFERENCES

- [1] G. Brunnett, "Geometric design with trimmed surfaces", *Computing Supplementum*, vol. 10, pp. 101–115, 1995.
- [2] C. de Boor, *A practical guide to splines*, New York: Springer Verlag, 1978.
- [3] G. Farin, *Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide*, Boston: Academic Press, 1997.
- [4] O. Figueiredo, J. Revelles and R. Hersch, "Digitization of Bézier curves and patches using discrete geometry", in *Proc. 8th Int. Conf. Discrete Geometry for Computer Imagery*, Marne-la-Vallée, 1999, pp. 388–398.
- [5] M. Floater, "Chordal cubic spline interpolation is fourth order accurate", *IMA J. Numer. Anal.*, vol. 26, pp. 25–33, 2006.

- [6] T. Hain, "Rapid termination evaluation for recursive subdivision of Bézier curves", in: *Proc. Int. Conf. on Image Science, Systems, and Technology*, Las Vegas, 2002, pp. 323–328.
- [7] H. Prautzsch, W. Boehm and M. Paluszny, *Bézier and B-Spline techniques*, Berlin: Springer Verlag, 2002.
- [8] M. Randrianarivony, "Geometric processing of CAD data and meshes as input of integral equation solvers". Ph.D. dissertation, Dept. Comput. Science, Chemnitz University of Technology, Chemnitz, Germany, 2006.
- [9] M. Randrianarivony and G. Brunnett, "Molecular surface decomposition using geometric techniques", in *Proc. Conf. Bildverarbeitung für die Medizin*, Berlin, 2008, pp. 197–201.
- [10] M. Randrianarivony and G. Brunnett, "Preparation of CAD and Molecular Surfaces for Meshfree Solvers", in *Proc. Int. Workshop Meshfree Methods for PDE*, Bonn, 2007, pp. 231–245.
- [11] J. Roullet, "Specifying the arc length of Bézier curves", *Computer Aided Geometric Design*, Vol. 10, pp. 25–56, 1993.
- [12] H. Seidel, "Polar forms for geometrically continuous spline curves of arbitrary degree", *ACM Trans. Graph.*, Vol. 12, pp. 1–34, 1993.
- [13] M. Walter and A. Fournier, "Approximate arc length parametrization", in *Proc. 9th Brazilian Symposium on Computer Graphics and Image Processing*, Brazil, 1996, pp. 143–150.

Maharavo Randrianarivony obtained his Master of Science from the Mathematics Department of the University of Chemnitz, Germany in 2001. He was mainly interested in numerical mathematics and computer science. He obtained his Ph.D. from the Department of Computer Science of the same University in 2006. The topic of his Ph.D. was in computer aided geometric design and CAD.

Right after his studies, he has done a PostDoc research for one year at the Institute of Computer Science in Kiel, Germany. He has treated there various scientific topics related to Geometric Modeling and CAD. Now, he holds a position as an Advanced Researcher at the Faculty of Mathematics of the University of Bonn, Germany.