Recent advances on Accuracy and Stability in Approximation and C.A.G.D.

J.M. Peña

University of Zaragoza, Spain

Effects of finite precision arithmetic on numerical algorithms:

- Roundoff errors.
- Data uncertainty.

Key concepts:

- *Conditioning*: it measures the sensibility of solutions to perturbations of data.
- *Growth factor*: it measures the relative size of the intermediate computed numbers with respect to the initial coefficients or to the final solution.
- *Backward error*: if the computed solution is the exact solution of a perturbated problem, it measures such perturbation.
- *Forward error*: it measures the distance between the exact solution and the computed solution.

(Forward error) $\leq$ *(Backward error)* $\times$ *(Condition)*

N.J. Higham. *Accuracy and Stability of Numerical Algorithms, second ed.*. SIAM, Philadelphia, PA, 2002.

Error analysis

Given $a \in \mathbf{R}$, the computed element in floating point arithmetic will be denoted by either $\mathrm{fl}(a)$ or by $\hat{a}$.

Models:

$$\mathrm{fl}(a \,\mathrm{op}\, b) = (a \,\mathrm{op}\, b)(1 + \delta), \quad |\delta| \leq u,$$

$$\mathrm{fl}(a \,\mathrm{op}\, b) = \frac{(a \,\mathrm{op}\, b)}{(1 + \varepsilon)}, \quad |\varepsilon| \leq u,$$

with u the unit roundoff and *op* any of the elementary operations $+$, $-$, $\times$, $/$.

Given $k \in \mathbf{N}_0$ such that $ku < 1$, let us define

$$\gamma_k := \frac{ku}{1 - ku}.$$

We shall deal with quantities θ_k satisfying that their absolute value is upperly bounded by γ_k.

$$\gamma_k := \frac{ku}{1 - ku}.$$

$$|\theta_k| \le \gamma_k$$

Properties:

a) $(1 + \theta_k)(1 + \theta_j) = 1 + \theta_{k+j}$,

b) $\gamma_k + \gamma_j + \gamma_k \gamma_j \le \gamma_{k+j}$,

c) $\gamma_k + u \le \gamma_{k+1}$,

d) if $\rho_i = \pm 1$, $|\delta_i| \le u$ $(i = 1, \dots, k)$ then

$$\prod_{i=1}^{k} (1 + \delta_i)^{\rho_i} = 1 + \theta_k.$$

Accurate algorithm: the relative error is bounded by $\mathcal{O}(\varepsilon)$, where ε is the machine precision. They are called **HRA** (with High Relative Accuracy) algorithms.

Admissible operations in algorithms with high relative precision: products, quotients, sums of numbers of the same sign and sums/subtractions of exact data. They are called **NIC** (no inaccurate cancellation) algorithms:

The only **forbidden** operation is true subtraction, due to possible cancellation in leading digits.

J. Demmel, I. Dumitriu, O. Holtz, P. Koev: Accurate and efficient expression evaluation and linear algebra, *Acta Numer.* **17** (2008), 87-145.

Evaluating $x + y + z$ is **not possible with HRA**.

For some **structured** classes of matrices, HRA algorithms can be found. But we also know that this is not possible for other classes of structured matrices. For instance, the **determinant of a Toeplitz** matrix **cannot be evaluated with HRA**.

$$B = \begin{pmatrix} a_0 & a_1 & a_2 & a_3 & a_4 \\ a_{-1} & a_0 & a_1 & a_2 & a_3 \\ a_{-2} & a_{-1} & a_0 & a_1 & a_2 \\ a_{-3} & a_{-2} & a_{-1} & a_0 & a_1 \\ a_{-4} & a_{-3} & a_{-2} & a_{-1} & a_0 \end{pmatrix}$$

Even if HRA algorithms exist, if the matrices are ill-conditioned, then they may need to be **re-parameterized**: first task to construct algorithms with HRA.

With these new parameters, new algorithms can be designed to compute the desired values (eigenvalues, singular values, inverses or solutions of the corresponding linear systems) with HRA.

In order to guarantee accurate computations for some special classes of matrices, it is crucial to find an **adequate parametrization** adapted to the special classes of matrices:

- For **diagonally dominant** M-matrices: the off-diagonal entries and the row sums.
- For nonsingular **totally positive** matrices: the multipliers of its Neville elimination (which correspond to their bidiagonal factorzation).

TP and SR matrices

Definition. A matrix is *strictly totally positive* (STP) if all its minors are positive and it is *totally positive* (TP) if all its minors are nonnegative.

Definition. A matrix is called *sign-regular* (SR) if all $k \times k$ minors of A have the same sign (which may depend on k) for all k. If, in addition, all minors are nonzero, then it is called *strictly sign-regular* (SSR).

Variation diminishing properties of sign-regular matrices A: if A is a nonsingular $(n+1) \times (n+1)$ matrix, then A is sign-regular if and only if the number of changes of strict sign in the ordered sequence of components of $A\mathbf{x}$ is less than or equal to the number of changes of strict sign in the ordered sequence $(x_0, \ldots, x_n)$, for all $\mathbf{x} = (x_0, \ldots, x_n)^T \in \mathbf{R}^{n+1}$.

I.J. Schoenberg: Über Variationsderminderende lineare Transformationem. *Math. Z.* **32** (1930), 321-328.

Factorizations in terms of bidiagonal matrices

If K is TP and nonsingular, then we can write

$$K = L_{n-1} L_{n-2} \cdots L_1 D U_1 \cdots U_{n-2} U_{n-1},$$

where the matrices L_i (resp., U_i) are nonnegative lower (resp., upper) triangular **bidiagonal** with unit diagonal and D is a **diagonal** matrix with **positive** diagonals.

Uniqueness of the factorization, under certain conditions, in:

M. Gasca, J.M. P.: A matricial description of Neville elimination with applications to total positivity. *Linear Alg. Appl.* **202** (1994), 33–54.

Tridiagonal (Jacobi) SR nonsingular matrices have been characterized in:

A. Barreras, J.M. P.: Characterizations of Jacobi sign regular matrices, *Linear Algebra and its Applications* **436** (2012), pp. 381-388.

If K is SSR and nonsingular, then we can write

$$K = L_{n-1}L_{n-2}\cdots L_1 D U_1 \cdots U_{n-2}U_{n-1},$$

where the matrices L_i (resp., U_i) are nonnegative lower (resp., upper) triangular **bidiagonal** with unit diagonal and D is a **diagonal** matrix with nonzero diagonals:

M. Gasca, J.M. P.: A test for strict sign-regularity. *Linear Alg. Appl.* **197-198** (1994), 133–142.

If K is nonsingular TP and stochastic, then we can write

$$K = F_{n-1} F_{n-2} \cdots F_1 G_1 \cdots G_{n-2} G_{n-1},$$

with

$$F_i = \begin{pmatrix} 1 & & & & & & & \\ 0 & 1 & & & & & & \\ & \ddots & \ddots & & & & & \\ & & 0 & 1 & & & & \\ & & & \alpha_{i+1,1} & 1-\alpha_{i+1,1} & & & \\ & & & & \ddots & \ddots & & \\ & & & & & \alpha_{n,n-i} & 1-\alpha_{n,n-i} \end{pmatrix}$$

and

$$G_i = \begin{pmatrix} 1 & 0 & & & & & & \\ & \ddots & \ddots & & & & & \\ & & 1 & 0 & & & & \\ & & & 1-\alpha_{1,i+1} & \alpha_{1,i+1} & & & \\ & & & & \ddots & \ddots & & \\ & & & & & 1-\alpha_{n-i,n} & \alpha_{n-i,n} \\ & & & & & & 1 \end{pmatrix},$$

where, $\forall\,(i,j)$, $0 \leq \alpha_{i,j} < 1$.

Interpretation in **CAGD** of this factorization as a **corner cutting algorithm**: the most important algorithms in CAGD.

The bidiagonal factorization of nonsingular TP matrices is associated to an elimination procedure alternative to Gauss elimination called **Neville elimination**. It requires $O(n^3)$ elementary operations to check if an $n \times n$ matrix is either TP or STP:

M. Gasca, J.M. P.: Total positivity and Neville elimination. *Linear Algebra Appl.* **165** (1992), 25-44.

Neville elimination produces zeros in each column by adding to each row an adequate multiple of the previous one (instead of a multiple of the pivot row as in Gauss elimination).

Neville elimination (NE)

If A is a nonsingular matrix of order n, it consists of $n-1$ steps:

$$A = A^{(1)} \to A^{(2)} \to \ldots \to A^{(n)} = U,$$

$$A^{(t)} = \begin{pmatrix} a_{11}^{(t)} & a_{12}^{(t)} & \ldots & \ldots & \ldots & \ldots & a_{1n}^{(t)} \\ 0 & a_{22}^{(t)} & \ldots & \ldots & \ldots & \ldots & a_{2n}^{(t)} \\ \vdots & 0 & \ddots & & & & \vdots \\ \vdots & \vdots & & \ddots & & & \vdots \\ \vdots & \vdots & & & a_{tt}^{(t)} & \ldots & a_{tn}^{(t)} \\ \vdots & \vdots & & & \vdots & & \vdots \\ 0 & 0 & \ldots & \ldots & a_{nt}^{(t)} & \ldots & a_{nn}^{(t)} \end{pmatrix}.$$

$$a_{ij}^{(t+1)} = \begin{cases} a_{ij}^{(t)} & i \le t \\ a_{ij}^{(t)} - \frac{a_{it}^{(t)}}{a_{i-1,t}^{(t)}} a_{i-1,j}^{(t)} & i \ge t+1,\ a_{i-1,t}^{(t)} \ne 0 \\ a_{ij}^{(t)} & i \ge t+1,\ a_{i-1,t}^{(t)} = 0 \end{cases}$$

The element

$$p_{ij} := a_{ij}^{(j)}, \quad 1 \le j \le i \le n,$$

is called (i,j) *pivot* of the NE of A.

The element

$$m_{ij} := p_{ij}/p_{i-1,j}, \quad 1 \le j \le i \le n,$$

is called (i,j) *multiplier* of the NE of A.

The *complete Neville elimination* (CNE) of A: NE of A until obtaining U and NE of $V := U^T$.

- **Error analysis and recent applications of NE.**

ALONSO P., GASCA M., P. J.M.: “Backward error analysis of Neville elimination” (1997). *Applied Numerical Mathematics* **23**, pp. 193-204.

ALONSO P., DELGADO J., GALLEGO R., P. J.M.: “Neville elimination: an efficient algorithm with application to Chemistry” (2010). *Journal of Mathematical Chemistry* **48**, pp. 3-20.

ALONSO P., DELGADO J., GALLEGO R., P. J.M.: “A collection of examples where Neville elimination outperforms Gaussian elimination” (2010). *Applied Mathematics and Computation* **216**, pp. 2525-2533.

ALONSO P., DELGADO J., GALLEGO R., P. J.M.: “Growth Factors of Pivoting Strategies Associated to Neville Elimination” (2011). *Journal of Computational and Applied Mathematics* **235**, pp. 1755-1762.

Neville elimination (NE) leads to a factorization of a nonsingular totally positive matrix in terms of **bidiagonal** factors, and the elements appearing in the factorization (the **multipliers of NE**) are natural parameters of the matrix.

P. Koev: Accurate computations with totally nonnegative matrices, *SIAM J. Matrix Anal. Appl.* **29** (2007), no. 3, 731–751.

P. Koev: Accurate Eigenvalues and SVDs of Totally Nonnegative Matrices, *SIAM J. Matrix Anal. Appl.* **27** (2005), 1-23.

This factorization has been used to obtain **accurate computations** with **subclasses** of nonsingular totally positive matrices. In particular, accurate computation of their **inverses** , **SVD** and **eigenvalues**.

J. Demmel and P. Koev: The Accurate and Efficient Solution of a Totally Positive Generalized Vandermonde Linear System, *SIAM J. Matrix Anal. Appl.* **27** (2005), 142-152.

J.J. Martínez, J.M. P.: Fast algorithms of Bjrck-Pereyra type for solving Cauchy-Vandermonde linear systems, *Appl. Numer. Math.* **26** (1998), 343-352.

A. Marco, J.J. Martínez: A fast and accurate algorithm for solving Bernstein-Vandermonde linear systems, *Linear Algebra Appl.* **422** (2007), 616-628.

A. Marco, J.J. Martínez: Accurate computations with Said-Ball-Vandermonde matrices, *Linear Algebra Appl.* **432** (2010), 2894-2908.

J. Delgado, J.M. P.: Accurate computations with collocation matrices of rational bases (2013). *Applied Mathematics and Computation* **219**, pp. 4354-4364.

J. Delgado, J.M. P.: Fast and accurate algorithms for Jacobi-Stirling matrices (2014). *Applied Mathematics and Computation* **236**, pp. 253-259.

J. Delgado, J.M. P.: Accurate computations with collocation matrices of q-Bernstein polynomials (2015). *SIAM Journal on Matrix Analysis and its Applications* **36**, 880-893

We shall illustrate the use of Neville elimination to obtain bidiagonal decompositions with the case of STP matrices.

Theorem. *A matrix A is* **strictly totally positive** *if and only if the Neville elimination of A and A^T can be performed without row exchanges, all the mutipliers of the Neville elimination of A and A^T are positive and all the diagonal pivots of the Neville elimination of A are positive.*

Theorem. *Let A be a* **strictly totally positive** *matrix. Then A and A^{-1} admit factorizations in the form*

$$A^{-1} = G_1 G_2 \cdots G_n D^{-1} F_n F_{n-1} \cdots F_1 \quad , \quad A = \overline{F}_n \overline{F}_{n-1} \cdots \overline{F}_1 D \overline{G}_1 \cdots \overline{G}_n,$$

respectively, where F_i and $\overline{F}_i$, $i \in \{1, \ldots, n\}$, are the lower triangular *bidiagonal matrices given by*

$$F_i = \begin{pmatrix} 1 & & & & & & & \\ 0 & 1 & & & & & & \\ & \ddots & \ddots & & & & & \\ & & 0 & 1 & & & & \\ & & & -m_{i,i-1} & 1 & & & \\ & & & & -m_{i+1,i-1} & 1 & & \\ & & & & & \ddots & \ddots & \\ & & & & & & -m_{n,i-1} & 1 \end{pmatrix}$$

and

$$\overline{F}_i = \begin{pmatrix} 1 & & & & & & & & \\ 0 & 1 & & & & & & & \\ & \ddots & \ddots & & & & & & \\ & & 0 & 1 & & & & & \\ & & & m_{i0} & 1 & & & & \\ & & & & m_{i+1,1} & 1 & & & \\ & & & & & \ddots & \ddots & & \\ & & & & & & m_{n,n-i} & 1 \end{pmatrix},$$

G_i *and* $\overline{G}_i$, $i \in \{1, \dots, n\}$, *are the upper triangular bidiagonal matrices* *whose trasposes are given by*

$$G_i^T = \begin{pmatrix} 1 & & & & & & & & \\ 0 & 1 & & & & & & & \\ & \ddots & \ddots & & & & & & \\ & & 0 & 1 & & & & & \\ & & & -\widetilde{m}_{i,i-1} & 1 & & & & \\ & & & & -\widetilde{m}_{i+1,i-1} & 1 & & & \\ & & & & & \ddots & \ddots & & \\ & & & & & & -\widetilde{m}_{n,i-1} & 1 \end{pmatrix}$$

and

$$\overline{G}_i^T = \begin{pmatrix} 1 & & & & & & & & \\ 0 & 1 & & & & & & & \\ & \ddots & \ddots & & & & & & \\ & & 0 & 1 & & & & & \\ & & & \widetilde{m}_{i0} & 1 & & & & \\ & & & & \widetilde{m}_{i+1,1} & 1 & & & \\ & & & & & \ddots & \ddots & & \\ & & & & & & \widetilde{m}_{n,n-i} & 1 \end{pmatrix},$$

and D the diagonal matrix $diag(p_{00}, p_{11} \ldots, p_{nn})$. The entries m_{ij}, $\widetilde{m}_{ij}$ are the multipliers of the Neville elimination of A and A^T, respectively, and the entries p_{ii} are the diagonal pivots of A.

For many **subclasses** of nonsingular totally positive matrices (Vandermonde matrices, Cauchy matrices, Cauchy-Vandermonde matrices, Bernstein-Vandermonde matrices, rational Bernstein collocation matrices, Jacobi-Stirling matrices, Pascal matrices, q-Bernstein collocation matrices,....), many **accurate computations** can be assured: **inverses** , **SVD** and **eigenvalues**.

Open problems:

- Finding **more classes** of matrices with accurate computations.
- Accurate solution of **linear systems** even for the previous classes.

Condition number

$$\kappa(A) := \|A\|_\infty \, \|A^{-1}\|_\infty.$$

The Skeel condition number:

$$\mathrm{Cond}(A) := \| \, |A^{-1}| \, |A| \, \|_\infty.$$

- $\mathrm{Cond}(A) \le \kappa(A)$
- $\mathrm{Cond}(DA) = \mathrm{Cond}(A)$ for any nonsingular diagonal matrix D

Minimal eigenvalue of TP matrices

DELGADO J., P. J.M.: "Progressive iterative approximation and bases with the fastest convergence rates" (2007). *Computer Aided Geometric Design* **24**, pp. 10-18.

Theorem. The minimal eigenvalue of a Bernstein collocation matrix is always greater than or equal to the minimal eigenvalue of the corresponding collocation matrix of another NTP basis of the space.

Optimal conditioning of Bernstein collocation matrices

DELGADO J., P. J.M.: "Optimal conditioning of Bernstein collocation matrices" (2009). *SIAM J. Matrix Anal. Appl.* **31**, 990-996.

Theorem. Let $(b_0, \dots, b_n)$ be the **Bernstein basis**, let $(v_0, \dots, v_n)$ be another NTP basis of P_n on $[0, 1]$, let $0 \leq t_0 < t_1 < \cdots < t_n \leq 1$ and $V := M\binom{v_0,\dots,v_n}{t_0,\dots,t_n}$ and $B := M\binom{b_0,\dots,b_n}{t_0,\dots,t_n}$. Then

$$\kappa_\infty(B) \leq \kappa_\infty(V).$$

$$\mathrm{Cond}(A) := \| \, |A^{-1}| \, |A| \, \|_\infty .$$

Theorem. Let $(b_0, \ldots, b_n)$ be the **Bernstein basis**, let $(v_0, \ldots, v_n)$ be another TP basis of P_n on $[0, 1]$, let $0 \leq t_0 < t_1 < \cdots < t_n \leq 1$ and $V := M\binom{v_0,\ldots,v_n}{t_0,\ldots,t_n}$ and $B := M\binom{b_0,\ldots,b_n}{t_0,\ldots,t_n}$. Then

$$\mathrm{Cond}(B^T) \leq \mathrm{Cond}(V^T).$$

ALONSO P., DELGADO J., GALLEGO R., P. J.M. (2013): "Conditioning and accurate computations with Pascal matrices". *Journal of Computational and Applied Mathematics* **252**, pp. 21-26.

Definition. A system of functions $(u_0, \ldots, u_n)$ is *totally positive* (TP) if all its collocation matrices are totally positive.

TP systems of functions are interesting due to the *variation diminishing* properties of totally positive matrices

Definition. A TP basis $(u_0, \ldots, u_n)$ is *normalized totally positive* (NTP) if

$$\sum_{i=0}^{n} u_i(t) = 1, \quad \forall t \in I.$$

Collocation matrices of NTP systems are TP and stochastic

In CAGD, NTP bases are associated with **shape preserving representations**.

The **normalized B-basis** is the basis with **optimal shape preserving properties**.

The **Bernstein** basis is the normalized B-basis of the space of polynomials of degree less than or equal to n on a compact interval $[a, b]$:

$$b_i(t) := \binom{n}{i} \left(\frac{t-a}{b-a}\right)^i \left(\frac{b-t}{b-a}\right)^{n-i}, \quad i = 0, \ldots, n.$$

CARNICER J.M., P. J.M.: "Shape preserving representations and optimality of the Bernstein basis" (1993). *Advances in Computational Mathematics* **1**, pp. 173-196.

CARNICER J.M., P. J.M.: "Totally positive bases for shape preserving curve design and optimality of B-splines" (1994). *Computer Aided Geometric Design* **11**, pp. 633-654.

basis $u = (u_0, \ldots, u_n)$ of a real vector space $\mathcal{U}$ of functions defined on a subset K of $\mathbf{R}^s$ and a function $f \in \mathcal{U}$,

$$f(x) = \sum_{i=0}^{n} c_i u_i(x).$$

We want to know how sensitive a value $f(x)$ is to any perturbations of a given maximal relative magnitude ε in the coefficients $c_0, \ldots, c_n$ corresponding to the basis. The corresponding perturbation $\delta f(x)$ of the change of $f(x)$ can be bounded by means of a **condition number**

$$C_u(f, x) := \sum_{i=0}^{n} |c_i u_i(x)|,$$

for the evaluation of $f(x)$ in the basis u:

$$|\delta f(x)| \leq C_u(f(x))\varepsilon.$$

R. T. Farouki & V. T. Rajan (1988): On the numerical condition of polynomials of algebraic curves and surfaces 1. Implicit equations. *Comput. Aided Geom. Design* **5**, 215-252.

Farouki, R. T. & Goodman, T. N. T. (1996): On the optimal stability of Bernstein basis. *Math. Comp.* **65**, 1553–1566.

Relative condition number:

$$c_u(f,x) := \frac{C_u(f,x)}{|f(x)|} \left(= \frac{\sum_{i=0}^n |c_i u_i(x)|}{|\sum_{i=0}^n c_i u_i(x)|} \right).$$

Let $\hat{f}(x)$ be the computed value wit floating point arithmetic.

$$\hat{f}(x) = \sum_{i=0}^{n} \bar{c}_i u_i(x).$$

Backward error analysis provides bounds for

$$\frac{|\bar{c}_i - c_i|}{|c_i|} \quad \text{or} \quad |\bar{c}_i - c_i|$$

Forward error analysis provides bounds for

$$|f(x) - \hat{f}(x)|$$

(Forward error) $\leq$ *(Backward error)* $\times$ *(Condition)*

The natural partial order for real-valued functions induces a corresponding partial order on the bases for $\mathcal{U}$, via

$$u \preceq v \quad \text{if and only if} \quad C_u(f,t) \le C_v(f,t), \ \forall f \in \mathcal{U}, \ \forall t \in I.$$

Given a set $\mathcal{B}$ of bases of bases of a vector space $\mathcal{U}$ of functions defined on I, we say that a basis $b \in \mathcal{B}$ is **optimally stable** for the evaluation of functions among all bases of $\mathcal{B}$ if it is minimal with respect to this partial order among all bases in $\mathcal{B}$. We shall consider the set $\mathcal{B}$ of bases of $\mathcal{U}$ formed by functions with constant sign (i.e., each basis function is either nonnegative or nonpositive).

Theorem. The normalized B-bases are optimally stable.

Extension of optimally stable bases beyond total positivity context.

For spaces of **univariate** functions:

P. J.M.: "On the optimal stability of bases of univariate functions" (2002). *Numerische Mathematik* **91**, pp. 305-318.

P. J.M.: "A note on the optimal stability of bases of univariate functions" (2006). *Numerische Mathematik* **103**, pp. 151-154.

For spaces of **multivariate** functions:

LYCHE T., P. J.M.: "Optimally stable multivariate bases" (2004). *Advances in Computational Mathematics* **20**, pp. 149-159.

The tensor product b^{mn} of Bernstein bases is **optimally stable** on $[0, 1] \times [0, 1]$.

The tensor product of B-splines bases is **optimally stable**.

The Bernstein basis B of multivariate polynomials defined on a triangle (resp., tetrahedron) is **optimally stable**.

The error analysis of the corresponding evaluation algorithms performed in:

MAINAR E., P. J.M.: "Running error analysis of evaluation algorithms for bivariate polynomials in barycentric Bernstein form" (2006). *Computing* **77**, 97-111.

MAINAR E., P. J.M.: "Evaluation algorithms for multivariate polynomials in Bernstein Bézier form" (2006). *Journal of Approximation Theory* **143**, 44-61.

DELGADO J., P. J.M.: "Error analysis of efficient evaluation algorithms for tensor product surfaces" (2008). *Journal of Computational and Applied Mathematics* **219**, pp. 156-169.

Rational Bézier surfaces

Given the double-index $\alpha = \alpha_1\alpha_2$, with $0 \le \alpha_1 \le m$, $0 \le \alpha_2 \le n$, we can define the corresponding basis function

$$r_\alpha(x,y) := \frac{w_\alpha\, b^m_{\alpha_1}(x)\, b^n_{\alpha_2}(y)}{\sum_{\alpha_1=0}^{m}\sum_{\alpha_2=0}^{n} w_\alpha\, b^m_{\alpha_1}(x)\, b^n_{\alpha_2}(y)}.$$

The previous basis is **optimally stable**.

DELGADO J., P. J.M.: "A Corner Cutting Algorithm for Evaluating Rational Bézier Surfaces and the Optimal Stability of the Basis" (2007). *SIAM J. Scient. Comput.* **29**, pp. 1668-1682.

The usual method to evaluate rational Bézier surfaces uses the projection operator. In contrast, we propose a new evaluation method such that *all* steps are convex combinations. It is a **robust** algorithm with optimal growth factor.

Both previous algorithms are more stable than evaluation algorithms of nested type and with lower complexity which have also been considered.

We have also analyzed the running error analysis of the projection and the new evaluation algorithm. A posteriori error bounds are calculated simultaneously with the evaluation algorithm without increasing the computational cost considerably.

DELGADO J., P. J.M.: "Running Relative Error for the Evaluation of Polynomials" (2009). *SIAM Journal on Scientific Computing* **31** , pp. 3905-3921.

DELGADO J., P. J.M.: "Running error for the evaluation of rational Bézier surfaces" (2010). *Journal of Computational and Applied Mathematics* **233**, pp. 1685-1696.

DELGADO J., P. J.M.: "Running error for the evaluation of rational Bézier surfaces through a robust algorithm". *Journal of Computational and Applied Mathematics* (2011). *Journal of Computational and Applied Mathematics* **235**, pp. 1781-1789.

Triangular rational Bézier surfaces

The Bernstein polynomials of degree n on a triangle, $(b_{\mathbf{i}}^n)_{|\mathbf{i}|=n}$ are defined by $b_{\mathbf{i}}^n(\tau) = \frac{n!}{i_0! i_1! i_2!} \tau_0^{i_0} \tau_1^{i_1} \tau_2^{i_2}$, $|\mathbf{i}| = n$.

Now let us consider the rational Bernstein basis of order n $(\phi_{\mathbf{i}})_{|\mathbf{i}|=n}$ given by $\phi_{\mathbf{i}} = \frac{w_{\mathbf{i}}\, b_{\mathbf{i}}^n}{\sum_{|\mathbf{i}|=n} w_{\mathbf{i}}\, b_{\mathbf{i}}^n}$, where $(w_{\mathbf{i}})_{|\mathbf{i}|=n}$ is a sequence of positive weights.

The previous basis is **optimally stable**.

DELGADO J., P. J.M.: "On the evaluation of rational triangular Bézier surfaces and the optimal stability of the basis" (2013). *Advances in Computational Mathematics* **13**, pp. 701-721.

DELGADO J., PEÑA J.M.: "Algorithm 960: POLYNOMIAL: An object-oriented Matlab library of fast and efficient algorithms for polynomials". To appear in *Transactions on Mathematical Software.*

Construction and evaluation of polynomials in Bernstein form:

- Efficient constructions for the coefficients of a polynomial **in Bernstein form when** the polynomial is **not** given with this representation are provided.
- The presented **adaptative evaluation algorithm** uses VS (Volk and Schumaker) algorithm, de Casteljau algorithm and a compensated VS algorithm.

Construction of the Bernstein form from interpolation conditions

Given a sequence of parameters $(t_i)_{0\leq i\leq n}$ verifying $0 < t_0 < t_1 < \cdots < t_n < 1$ and a sequence of points $q = (q_i)_{0\leq i\leq n}$, there **exists a unique** $p(t) \in \mathcal{P}_n$ satisfying $p(t_i) = q_i$ for $0 \leq i \leq n$.

The interpolation conditions can be formulated as the following **Bernstein-Vandermonde linear system** of equations (BV linear system):

$$B(c_0, c_1, \ldots, c_n)^T = (q_0, q_1, \ldots, q_n)^T,$$

where B is the collocation matrix of the basis $(b_0^n, b_1^n, \ldots, b_n^n)$ at $t_0, t_1, \ldots, t_n$.

We only need to obtain the bidiagonal decomposition of the inverse of a BV matrix with high relative accuracy in order to solve the above BV linear system of equations **with high accuracy**, obtaining in this way the corresponding interpolation polynomial in Bernstein form.

Evaluation algorithms of a polynomial

Given the Bernstein basis $b_n := (b_0^n(t), b_1^n(t), \ldots, b_n^n(t))$, $t \in [0,1]$, and $p(t) = \sum_{i=0}^n c_i\, b_i^n(t)$, we can evaluate $p(t)$ with the **de Casteljau** algorithm.

L.L. Schumaker, W. Volk: "Efficient evaluation of multivariate polynomials", *Comput. Aided Geom. Design* **3** (1986), 149-154.

The **VS** basis $z_n := (z_0^n(t), z_1^n(t), \ldots, z_n^n(t))$, $t \in [0,1]$, is given by $z_i^n(t) = t^i(1-t)^{n-i}$, $i = 0, 1, \ldots, n$.

The corresponding condition numbers coincide $C_{z_n}(p(t)) = C_{b_n}(p(t))$. The **VS algorithm** has **linear** time complexity, whereas the de Casteljau algorithm has quadratic time complexity. Nevertheless, in the case of extremely ill-conditioned polynomials, the de Casteljau algorithm can outperform VS algorithm in terms of accuracy.

The usual polynomial evaluation algorithm is the Horner algorithm, which uses the **monomial** basis $m_n := (m_0^n(t), m_1^n(t), \ldots, m_n^n(t))$, $t \in [0,1]$, is given by $m_i^n(t) = t^i$, $i = 0, 1, \ldots, n$

Forward error bounds for the relative errors when evaluating the polynomial by the **Horner**, **de Casteljau** and **VS** algorithms:

$$\left|\frac{fl(p(t)) - p(t)}{p(t)}\right| \leq k \cdot n \cdot u \frac{C_{\mathcal{U}}(p(t))}{|fl(p(t))|} + \mathcal{O}(u^2),$$

assuming that $|fl(p(t))| > u\,k \cdot n \cdot C_{\mathcal{U}}(p(t))$ and $k\,n\,u < 1$, with $k = 2$ for Horner and de Casteljau, $k = 4$ for VS, $\mathcal{U} = m_n$ for Horner and $\mathcal{U} = b_n$ for de Casteljau and VS, where u is the unit roundoff.

The relative error bounds of de Casteljau and VS algorithms are lower than that of Horner algorithm due to the better conditioning of their bases.

Running error analysis for these algorithms have also been performed.

Graillat, Langlois and Louvet presented a **compensated** Horner algorithm for the evaluation of a polynomial represented in the monomial basis.

The compensated algorithm is accurate for not too ill-conditioned polynomials. In fact, the compensated version of an algorithm delays the effects of the bad conditioning in the accuracy of the results.

The key tool to obtain more accurate results is to apply what Ogita, Rump and Oishi call **error-free transformations**.

Let us recall the typical improvement of these compensated algorithms for the evaluation accuracy. If we have an evaluation algorithm of a polynomial $p(t)$ represented in a basis $\mathcal{U}$ with a forward error bound of the form:

$$|p(t) - fl(p(t))| \leq (|p(t)|C_{\mathcal{U}}(p(t))) \times \mathcal{O}(u),$$

then the compensated algorithm produces a computed evaluation $fl(p(t))$ satisfying

$$|p(t) - fl(p(t))| \leq |p(t)|u + (|p(t)|C_{\mathcal{U}}(p(t))) \times \mathcal{O}(u^2).$$

Adaptative evaluation algorithm of a polynomial in Bernstein form

In general, VS and de Casteljau algorithms provide similar approximations with respect to accuracy. The same occurs with its compensated versions.

In general, the de Casltejau and VS algorithms provide accurate enough approximations except for very ill-conditioned polynomials, where their compensated versions can be very useful. The compensated version of the de Casteljau algorithm is very expensive computationally and it provides results with an accuracy very similar to those provided by the compensated VS algorithm.

We propose an **adaptative evaluation algorithm** using the **VS algorithm** except when the relative error bound requires more accuracy. In this case, we apply either the de Casteljau algorithm or a compensated VS algorithm, depending on the computational cost.

For degrees $n \leq 32$ de Casteljau algorithm has a lower computational cost than that of to the compensated VS algorithm, whereas for other degrees the compensated VS algorithm is more efficient.

If $n \geq 33$ we evaluate the polynomials by the compensated VS algorithm, and otherwise, first we evaluate them by the de Casteljau algorithm and only the points where the corresponding test error is not satisfactory are evaluated by the compensated VS algorithm.