The Butcher Factor

Robert M. Corless

March 2024

Western University, Canada

woRK 2024, Celebrating the 91st birthday of John C. Butcher

Announcing Maple Transactions

a “Diamond” class open access journal with no page charges

Now listed by DBLP

mapletransactions.org

Interpolational polynomials

"...a theorem of great antiquity...the simple theorems of
polynomial interpolation upon which
much practical numerical analysis rests."
—Philip J. Davis, Interpolation and Approximation
quoted page 290 in Hairer & Wanner II

- Although the idea is very old, new things come up from time to time (see e.g. [4])
- I will talk about a useful technique introduced in a paper of John's published in 1967, namely [3]. I call this technique *the Butcher factor*.
- We used the Butcher factor in [2] in 2011 for Birkhoff interpolation, but I think it deserves wider attention.

Polynomial bases and numerical condition

Avoid changing the polynomial basis you use, because the condition number of the change-of-basis matrix is usually exponential in the degree. [FFT is unusual: condition number is just 1.]

The Lagrange basis is frequently the best-conditioned*.

The word "interpolation" is usually taken to mean "convert from a Lagrange basis to the monomial basis." But it doesn't have to mean that. The barycentric forms of the Lagrange interpolational polynomial are efficient and perfectly (componentwise!) numerically stable.

Hermite interpolational bases, which use derivative data as well, are not as good, but they are not bad, when the confluencies are not large.

* Corless & Watt, 2004, "Bernstein bases are optimal, but, sometimes, Lagrange bases are better."

Example

Suppose the values of a polynomial p are known at the nodes $\tau = [-1, -{}^1\!/\!_2, {}^1\!/\!_2, 1]$. Say they are $[\rho_0, \rho_1, \rho_2, \rho_3]$. Then, without converting to a monomial basis, the polynomial can be written as

$$p(z) = w(z)\left(-\frac{2\rho_0/3}{z+1} + \frac{4\rho_1/3}{z+1/2} - \frac{4\rho_2/3}{z-1/2} + \frac{2\rho_3/3}{z-1}\right) \tag{1}$$

where the *node polynomial* is $w(z) = (z+1)\,(z+1/2)\,(z-1/2)\,(z-1)$.

Second barycentric form

Equivalently but sometimes better, the polynomial may be written as

$$p(z) = \frac{-\frac{2\rho_0/3}{z+1} + \frac{4\rho_1/3}{z+1/2} - \frac{4\rho_2/3}{z-1/2} + \frac{2\rho_3/3}{z-1}}{-\frac{2/3}{z+1} + \frac{4/3}{z+1/2} - \frac{4/3}{z-1/2} + \frac{2/3}{z-1}} \tag{2}$$

and a further improvement can be made by cancelling common factors in the numerator and denominator (this helps to avoid overflow and underflow for larger examples).

These look ridiculous, but I tell you they're beautiful. Berrut & Trefethen 2004 and N.J. Higham 2004 proved them to be componentwise numerically stable: evaluation is fast, and robust.

Background: Partial Fractions and the Cauchy Integral Formula

Suppose

$$\frac{1}{(z-\theta)z^2(z-1)^2} = \frac{1/\theta^2(\theta-1)^2}{z-\theta} - \frac{1/\theta}{z^2} - \frac{(1+2\theta)/\theta^2}{z} + \frac{(2\theta-3)/(\theta-1)^2}{z-1} + \frac{1/(1-\theta)}{(z-1)^2} \tag{3}$$

is written in terms of partial fractions.

Background continued

Then we can rewrite the equation using a contour that encloses all poles, as follows. This equation is valid for polynomials $p(z)$ of grade* 3:

$$0 = \frac{1}{2\pi i} \oint_C \frac{p(z)}{(z-\theta)z^2(z-1)^2}\, dz \tag{4}$$

using that partial fraction expansion together with the Cauchy Integral Formula

$$\frac{f^{(j)}(a)}{j!} = \frac{1}{2\pi i} \oint_C \frac{f(z)}{(z-a)^{j+1}}\, dz \tag{5}$$

as follows.

* the word "grade" means "degree at most." This is a convention widely used in papers on polynomial eigenvalue problems.

Background continued

$$
\begin{aligned}
0 = &\frac{1}{\theta^2(\theta-1)^2}p(\theta) \\
&-\frac{1}{\theta}p'(0)-\frac{(1+2\theta)}{\theta^2}p(0) \\
&+\frac{(2\theta-3)}{(\theta-1)^2}p(1)+\frac{1}{(1-\theta)}p'(1)\,. \qquad (6)
\end{aligned}
$$

Isolating the term containing $p(\theta)$ and multiplying by $\theta^2(\theta-1)^2$ gives the unique grade 3 Hermite interpolational polynomial with given values and derivatives at 0 and 1.

Explicitly

$$
\begin{aligned}
p(\theta) = &(2\theta + 1)(\theta - 1)^2 \, p(0) \\
&+ \theta^2 (3 - 2\theta) \, p(1) \\
&+ \theta (\theta - 1)^2 \, D(p)(0) \\
&+ (\theta - 1) \, \theta^2 D(p)(1) \ . \qquad (7)
\end{aligned}
$$

[It's better numerically and for efficiency *not* to write these this way, when the grade is much higher. For cubic Hermite, it doesn't matter.] This technique is usable by hand, but also works well in a computer algebra system such as Maple where one can compute residues easily.

"Takes all the fun out of it." — G. V. Parkinson

Some applications

General Hermite interpolational polynomials can be constructed using the node polynomial $w(z) = \prod_{i=0}^{n}(z-\tau_i)^{s_i}$ by expanding

$$\frac{1}{w(z)} = \sum_{i=0}^{n}\sum_{j=0}^{s_i-1} \frac{\beta_{i,j}}{(z-\tau_i)^{j+1}} \tag{8}$$

which gives the "generalized barycentric weights" $\beta_{i,j}$. These can be further used to construct differentiation matrices [1] for polynomials given by the Hermite interpolational data on these nodes with the given confluencies s_j. That is, the data are local Taylor series for $f(z) = \rho_{i,0} + \rho_{i,1}(z-\tau_i) + \rho_{i,2}(z-\tau_i)^2 + \cdots \rho_{i,s_i-1}(z-\tau_i)^{s_i-1}$ known at each node, with known coefficients $\rho_{i,j} = f^{(j)}(\tau_i)/j!$ incorporating the factorials.

Barycentric Hermite Interpolational Polynomial

The first and second forms are

$$p(z) = w(z) \sum_{i=0}^{n} \sum_{j=0}^{s_i-1} \sum_{k=0}^{j} \frac{\beta_{i,j}\rho_{i,k}}{(z-\tau_i)^{j+1-k}} \tag{9}$$

and

$$p(z) = \frac{\sum_{i=0}^{n} \sum_{j=0}^{s_i-1} \sum_{k=0}^{j} \frac{\beta_{i,j}\rho_{i,k}}{(z-\tau_i)^{j+1-k}}}{\sum_{i=0}^{n} \sum_{j=0}^{s_i-1} \frac{\beta_{i,j}}{(z-\tau_i)^{j+1}}} \tag{10}$$

Both forms are quite numerically stable, unless the confluencies get too large. The second form allows scaling the $\beta_{i,j}$ to reduce the impact of overflow or underflow.

Enter the Butcher Factor

If we insert a polynomial factor $B(z)$ in the numerator, of degree* m, then we can choose it so as to force m residues to be zero. Fix such a choice for $B(z)$. Then

$$0 = \frac{1}{2\pi i} \oint_C \frac{B(z)p(z)}{(z-\theta)w(z)}\, dz \tag{11}$$

where, say, the node polynomial $w(z)$ is $\prod_{k=0}^{n}(z-\tau_k)^{s_k}$. If $d = -1 + \sum_{k=0}^{n} s_k$ then the above formula will be valid for all polynomials $p(z)$ of grade $d - m$.

* We need degree and not just grade, to know the accuracy/order.

The first use

John published [3] in 1967, where he used a factor of degree $m = 2$ to find an interpolational polynomial that would allow him to add derivative evaluations at $x_n - hu$ and $x_n - hv$ to a multistep formula in order to break some order barriers. The node polynomial was

$$w(z) = (z + hu)^2 (z + hv)^2 \prod_{j=0}^{k} (z + hj)^2 \tag{12}$$

where $u \neq v$ and neither u nor v were in $\{0, 1, \ldots, k\}$. He wrote the factor (he used $\varphi(z)$ in place of $B(z)/w(z)$) in a way equivalent to the following:

$$\frac{B(z)}{K(k!)^2 h^{2k+2}} = (z + hu + hU/2)(z + hv)^2 - (z + hv + hV/2)(z + hu)^2$$

for some constants K, U, and V to be determined in order to make two residues zero and one residue -1. [The z^3 terms cancel.]

He gave elegant explicit formulae for these constants.

John's solution

$$\frac{1}{U} = \sum_{j=0}^{k} \frac{1}{j-u} \tag{13}$$

$$\frac{1}{V} = \sum_{j=0}^{k} \frac{1}{j-v} \tag{14}$$

$$\begin{aligned} \frac{1}{K} =& H_k \left(\frac{2}{u} + \frac{U}{u^2} - \frac{2}{v} - \frac{V}{v^2} \right) \\ & + \frac{1}{u^2} + \frac{U}{u^3} - \frac{1}{v^2} - \frac{V}{v^3} \, . \end{aligned} \tag{15}$$

Here H_k is the harmonic sum $1 + 1/2 + 1/3 + \cdots 1/k$. Notice that this solution is for symbolic integer k.

Birkhoff interpolation

If some of the Hermite interpolational data is *missing*, then we have what is called a *Birkhoff* interpolation problem. Not all such are uniquely solvable.

Example 1: Asking for a degree two polynomial that satisfies $p(0) = p_0$, $p'(0) = d_0$, and $p'(1) = d_1$ has a unique solution: $p(x) = p_0 + d_0 x + (d_1 - d_0)x^2/2$.

Example 2: Asking for a degree two polynomial that satisfies $p(0) = p_0$, $p(1) = p_1$, and $p'(1/2) = d_{1/2}$ either has no solution or infinitely many. This problem is not "poised" or "R-regular." [Suppose $p_0 = p_1$—then $p'(1/2) = 0$ no matter what p_0 is.]

Butcher factors for Birkhoff interpolation

In [2] we used Butcher factors to solve some quite general Birkhoff interpolational problems. If the problem was poised, we were able to do so *except* on certain algebraic surfaces.

Setting many residues to zero

If $w(z) = \prod_{\ell=0}^{k}(z-\tau_\ell)^2$ and we want the residue of $B(z)/w(z)$ at $z = \tau_j$ for $1 \le j \le k$ to be zero, put $y = \prod_{\ell \ge 1 \& \ell \ne j}(z-\tau_\ell)^2$ and expand it in a local Taylor series at $z = \tau_j$, for instance by taking logarithms:

$$\ln y(z) = \sum_{\ell \ge 1\ \&\ \ell \ne j} -2\ln(z-\tau_\ell) + 2\pi i K \tag{16}$$

$$\frac{y'(z)}{y(z)} = -2 \sum_{\ell \ge 1\ \&\ \ell \ne j} \frac{1}{z-\tau_\ell}\,. \tag{17}$$

Continued

So the local Taylor series for $y(z)$ is

$$y(z) = y(\tau_j) - 2y(\tau_j) \left(\sum_{\ell \geq 1 \,\&\, \ell \neq j} \frac{1}{\tau_j - \tau_\ell} \right) (z - \tau_j) + O(z - \tau_j)^2 \,. \quad (18)$$

Multiply by $B(z) = B(\tau_j) + B'(\tau_j)(z - \tau_j) + O(z - \tau_j)^2$ and we have, since $y(\tau_j) \neq 0$,

$$B'(\tau_j) - 2B(\tau_j) \left(\sum_{\ell \geq 1 \,\&\, \ell \neq j} \frac{1}{\tau_j - \tau_\ell} \right) = 0 \,. \quad (19)$$

as the required condition. Typically we will also require the residue at $z = \tau_0$ to be -1. This gives $k + 1$ equations, normally sufficient for a degree k Butcher factor. But sometimes we can do *better*.

An elegant use of the CIF

In the paper, John* gave an elegant construction of Gaussian quadrature on Legendre points by using the Birkhoff interpolation problem $P(-1) = 0$ and $P'(x_i) = p(x_i)$ for $1 \le i \le n$. The polynomial satisfying that has $P(1) = \int_{-1}^{1} p(x)\,dx$. Insisting that this works for all polynomials of degree $2n$ gives that the x_i are the zeros of a Legendre polynomial. In this case we could take the Butcher factor to be just $B(z) = 1$. [See the paper.]

* Yes, that part of the paper was all his.

Some details

The following holds for all polynomials $f(z)$ of grade $2n$:

$$0 = \frac{1}{2\pi i} \oint_C \frac{f(z)}{(z+1)(z-1)P^2(z)}\, dz\,. \tag{20}$$

where $P(z) = \prod_{k=1}^{n}(z - x_k)$. If this gives a quadrature formula, then the residue at each x_k must be zero. Choosing x to be one of the x_k and setting the residue to zero gives (after some work!)

$$\frac{2x}{x^2-1} + \frac{P''(x)}{P'(x)} = 0\,. \tag{21}$$

Rearranging that gives $(x^2-1)P''(x) + 2xP'(x) = 0$ at each of the n nodes x_k. So that polynomial must be a multiple of $P(x)$ (having the same zeros and being the same grade and having leading coefficient $n(n+1) \neq 0$). Therefore $(z^2-1)P''(z) + 2zP'(z) - n(n+1)P(z) = 0$ so $P(z)$ is a multiple of a Legendre polynomial. $\natural$

Some Maple code

In September 2009 I wrote a Maple procedure, `BHBIP.mpl`, to use this method. It can construct the Hermite–Birkhoff interpolational polynomial explicitly, or it can “fill in” the missing data.

Piers Lawrence wrote a Matlab version of the code.

Filling in missing data

If only one piece of data is missing, then we do not need a Butcher factor. Simply expanding $1/w(z)$ in partial fractions gives us (after contour integration) an equation for the missing piece of information.

More generally, if we are missing m pieces of information, we can use a Butcher factor of degree $m-1$ for each one.

Even better, if we wish to fill in *all* the missing data, we can do this all at once by inverting an $m-1$ by $m-1$ matrix*.

* This is nearly the only time I have ever explicitly used a matrix inverse in any of my codes. The reason is that each column of the inverse gives the coefficients of the Butcher factor of degree $m-1$ needed to identify one of the missing pieces of data.

A more complicated example

Suppose we know that $p(0) = y_0$, $p'(0) = d_0$, $p'(c_1) = f_1$, $p'(c_2) = f_2$, $p(1) = y_1$, and $p'(1) = d_1$. So the function values at $x = c_2$ and $x = c_1$ are missing. Then the fill-in technique needs only two one-by-one matrices to set the residues separately to zero. Formulae for the missing data look like

$$p(c_1) = ay_0 + bd_0 + cf_1 + df_2 + ey_1 + fd_1 \tag{22}$$

where (for instance)

$$b = -\frac{\left(3c_1c_2 - 5c_2^2 - c_1 + 3c_2\right) c_1 \left(c_1 - 1\right)^3}{2c_2\left(10c_1c_2 - 5c_1 - 5c_2 + 3\right)} . \tag{23}$$

Notice the nontrivial combination of c_1 and c_2 in the denominator. For this technique to succeed, that polynomial cannot be zero. But in fact the problem is not poised if

$$2c_1c_2\left(c_2 - 1\right)\left(c_1 - 1\right)\left(10c_1c_2 - 5c_1 - 5c_2 + 3\right)\left(c_2 - c_1\right) = 0 . \tag{24}$$

The forbidden c_1 and c_2

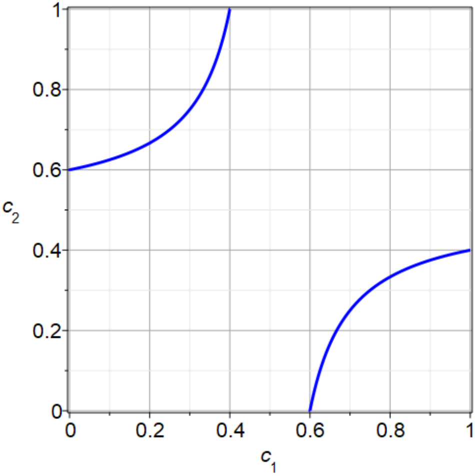

Figure 1: If the point (c_1,c_2) lies on the blue curves, or else if $c_1 c_2(c_1 - c_2) = 0$, the Birkhoff interpolation problem is not poised.

Human vs Computer

Any given Birkhoff interpolation problem of explicit dimension can instead be solved directly by setting up a linear system of equations. The matrix will be a submatrix of a confluent Vandermonde matrix. For computers, it's not clear at first that the Butcher factor technique offers much advantage. The matrices needed are smaller, yes, but the setup is more involved.

A *big* disadvantage of the Vandermonde approach if floating-point arithmetic is involved is that this changes the basis, which if the grade is at all large may introduce serious numerical instability. It is (generally speaking) *much* better to stay in the same polynomial basis, if you can. Indeed, my preferred way to write a Butcher factor is in a Lagrange interpolational basis!

Human considerations

The gain in insight is potentially considerable, and moreover one can (as John did in 1967) solve problems of symbolic dimension, for arbitrary integers k. [One can do that with the code as well, but only by solving a few cases $k = 1$, $k = 2$, $k = 7$, whatever, and *guessing* the general form. If we're lucky we could prove it afterwards.]

A disadvantage of the Butcher factor is that sometimes, even if the problem is poised, the technique can fail. The Butcher factor is not allowed to have a zero at any of the nodes (and if we are unlucky, this can happen, say for symmetry reasons). The technique needs to be used carefully.

One such example

Suppose $\tau = [-1, 0, 1]$ are the nodes, and we know $p(\tau_0) = \rho_0$, $p''(\tau_1) = d_2$, and $p(\tau_2) = \rho_2$. This problem is poised: using the Vandermonde approach produces $p(z) = (\rho_0 + \rho_2 + d_2)/2 + (\rho_2 - \rho_0)z/2 + d_2 z^2/2$ easily enough. But asking for a Butcher factor that makes zero the two residues at $z = 0$ corresponding to $p(0)$ and $p'(0)$ forces $B(z) = \alpha(1 - z^2)$, which cancels the factors $(z+1)(z-1)$ of the node polynomial.

So the technique doesn't always work.

Even so, the method can be useful

If one has code to evaluate Hermite interpolational polynomials already, and to differentiate them, then the "fill-in" technique allows easy conversion of Birkhoff data to usable Hermite interpolational polynomials, with *no* introduced numerical instability.

Quadrature

Another trick is that one can directly integrate Hermite data: if $p(t)$ has known local Taylor series at various nodes, then its antiderivative $P(t)$ with $P(t_0) = 0$ and $P'(t) = p(t)$ has local Taylor series known at those nodes, as well, except for the as-yet undefined function values at t_1, t_2, $\ldots$. That's a Birkhoff interpolational problem.

Example

Suppose $\tau = [-1, 0, 1]$ and $p(\tau) = [p_0, p_1, p_2]$. Then $P(\tau) = [[0,p_0], [NaN,p_1], [NaN,p_2]]$ (Maple uses `undefined` for `NaN`). Using the fill-in technique on this gives

$$P(\tau) = \left[[0, p_0], \left[\frac{5p_0}{12} + \frac{2p_1}{3} - \frac{p_2}{12}, p_1 \right], \left[\frac{p_0}{3} + \frac{4p_1}{3} + \frac{p_2}{3}, p_2 \right] \right] \tag{25}$$

and we see the familiar Simpson's rule formula pop out at the end. Note that p was grade 2, while P must be grade 3. [As is well-known, Simpson's rule has an extra degree of accuracy, one more than is shown here, because one residue is zero.]

If instead the nodes are $[-1, -1/2, 1/2, 1]$ then the integral at the end is $\frac{p_0}{9} + \frac{8p_1}{9} + \frac{8p_2}{9} + \frac{p_3}{9}$ which is accurate at least for p of grade 3.

This isn't completely straightforward if there are other missing data, and I have examples where the approach surprisingly fails.

But it's pretty good

Suppose for example that the nodes are $[-1, -{}^1\!/_2, {}^1\!/_2, 1]$ and we know $f(-1) = p_0$, $f({}^1\!/_2) = p_2$, and $f(1) = p_3$ but for some reason we only know $f'(-{}^1\!/_2) = d_1$. Finding a polynomial interpolant for that data is already a Birkhoff problem. Now suppose that what we really want is not f but its integral across the interval. Then the Maple code tells us that

$$\int_{-1}^{1} f(x)\,dx \approx \frac{25p_0}{9} + \frac{8d_1}{3} - \frac{16p_2}{9} + p_3 \tag{26}$$

and this formula will be exact for polynomials $f(x)$ of grade 3.

Dufton's quadrature method

In one of the annotated bibliographies of the 1933 book that John and I will talk about on Friday, we find a reference to a paper by von Mises which analyzes the following formula originally due to A. F. Dufton:

$$\int_0^1 f(t)\,dt \approx \frac{1}{4}\left(f(0.1) + f(0.4) + f(0.6) + f(0.9)\right) . \tag{27}$$

Dufton's derivation of the formula split the interval in two, but we will do something different.

Using a Butcher factor

Consider

$$\frac{B(z)}{z(z-{}^{1}\!/\!{}_{10})^2(z-{}^{4}\!/\!{}_{10})^2(z-{}^{6}\!/\!{}_{10})^2(z-{}^{9}\!/\!{}_{10})^2(z-1)} \tag{28}$$

and expand it in partial fractions, choosing $B(z)$ to make the residues at $k/10$ zero for $k = 1$, 4, 6, and 9.

Then the method gives the formula

$$I \approx \frac{22}{90} f\left(\frac{1}{10}\right) + \frac{23}{90} f\left(\frac{4}{10}\right) + \frac{23}{90} f\left(\frac{6}{10}\right) + \frac{22}{90} f\left(\frac{9}{10}\right) \tag{29}$$

and this is exact for all f of grade 3 and is thus fourth-order accurate. Dufton's formula is only second-order accurate: but $22/90 \approx 0.244$ and $23/90 \approx 0.256$ so the error coefficients will be *small*. It's a pretty good rule, really!

Might we do better?

We can't get fourth order accurate without awkward abscissæ for hand computation (those are the constraints we set ourselves).

Can we find a similar rule that has smaller error, and is similarly convenient? If we apply the Butcher factor approach for *symbolic* abscissæ, we can solve them to find

$$x = \frac{1}{2} \pm \frac{\sin\theta}{\sqrt{6}}, \frac{1}{2} \pm \frac{\cos\theta}{\sqrt{6}} \tag{30}$$

will have all weights equal to $1/4$ for any θ, and give fourth order accuracy. The $\sqrt{6}$ is a problem, but we can choose θ so that one of $\sin(\theta)/\sqrt{6}$ or $\cos(\theta)/\sqrt{6}$ is rational; that will make two of the points rational.

But not all four

No matter what we do, at least two of the nodes will be irrational. But can we choose a *little* better than Dufton's rule?

Yes. If we take $1/2 - \sin(\theta)/\sqrt{6} = 1/8$ then it turns out* that $1/2 - \cos(\theta)/\sqrt{(}6) = 1/2 - \sqrt{15}/24 = 0.3386$ is "pretty close" to $1/3$. This gives the rule

$$\int_0^1 f(t)\,dt \approx \frac{1}{4}\left(f(1/8) + f(1/3) + f(2/3) + f(7/8)\right) \tag{31}$$

which is (slightly) better than Dufton's rule (and still eminently practical for hand computation). Could we do better yet? Maybe, but it is only an academic game, potentially to be "relegated to the category of useless things!"

* found by brute force and continued fractions

A comparison

$f(t)$	New error/Dufton error
$\exp(t)$	0.40
$\sin(t)$	0.64
$W(t)$	-0.39
t^5	-1.5
$\sqrt{t}$	3.0
$\ln(1+t)$	0.18

We see that when the integrand has a sharp slope at one end, the Dufton rule does better; both rules are "open" but the new one samples closer to the interior.

The correct weights for this new rule are $16/65 \approx 0.2461$ and $33/130 \approx 0.2538$ and that would make the rule fourth order accurate. Dufton's rule had $22/90 \approx 0.244$ and $23/90 \approx 0.256$, which were farther from $1/4$.

Differentiation and the Squire–Trapp formula

For analytic $f(z)$,

$$f'(z) = \frac{\Im(f(z+ih))}{h} + O(h^2) \tag{32}$$

with typically* almost no degradation from rounding errors as $h \to 0$ and $O(h^2)$ goes to unit roundoff.

The formula works very well to differentiate a Hermite–Birkhoff interpolational polynomial (remember that you are differentiating the polynomial and not the function it approximates). This technique compares very well to using the differentiation matrix for Hermite interpolational polynomials, especially if only a few values of the derivative are needed.

* If your function $f(z)$ is not coded well, there can be problems. For instance, $\Gamma(z)$ in Matlab for large $z > 0$.

Thank you for listening.

This work supported by NSERC, and by the Spanish MICINN.

I would like to particularly thank Erik Postma and my other co-authors, and especially to thank John C. Butcher for teaching me the contour integral technique for interpolation, by which I (re)derived all these formulae.

Happy Birthday, John!

References

[1] Amirhossein Amiraslani, Robert M Corless, and Madhusoodan Gunasingam. **"Differentiation matrices for univariate polynomials".** In: *Numerical Algorithms* 83 (2020), pp. 1–31.

[2] John C Butcher et al. **"Polynomial algebra for Birkhoff interpolants".** In: *Numerical Algorithms* 56.3 (2011), pp. 319–347.

[3] John C. Butcher. **"A Multistep Generalization of Runge-Kutta Methods With Four or Five Stages".** In: *J. ACM* 14.1 (1967), pp. 84–99. ISSN: 0004-5411. DOI: `http://doi.acm.org/10.1145/321371.321378`.

[4] Lloyd N. Trefethen. **Approximation theory and approximation practice.** SIAM, 2013.