High-precision methods for zeta functions
Part 2: derivatives

Fredrik Johansson
INRIA Bordeaux

UNCG Summer School in Computational Number Theory
May 18–22, 2015
Computing derivatives

\[ f(a + x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!} x^n \]

Reasons we might want to compute derivatives

- We are interested in the derivatives / Taylor coefficients themselves
- Removing singularities (computing limits)
- Analytic operations (integrals, finding roots or extreme points).

Newton’s method:

\[ x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \]
Numerical differentiation

Finite difference:

\[ f^{(n)}(a) \approx \frac{1}{h^n} \sum_{k=0}^{n} (-1)^{k+n} \binom{n}{k} f(a + kh), \quad h \sim 2^{-p} \]

Cauchy integral formula:

\[ f^{(n)}(a) = \frac{n!}{2\pi i} \oint_{\gamma} \frac{f(z)}{(z - a)^{n+1}} \, dz \]
Symbolic differentiation

\[
sage: \text{diff}(1/\cos(x), x, 10)
\]

\[
\frac{50521}{\cos(x)} + \frac{1326122\sin(x)^2}{\cos(x)^3} + \\
\frac{6749040\sin(x)^4}{\cos(x)^5} + \frac{13335840\sin(x)^6}{\cos(x)^7} + \\
\frac{11491200\sin(x)^8}{\cos(x)^9} + \frac{3628800\sin(x)^{10}}{\cos(x)^{11}}
\]

\[
\zeta^{(n)}(s) = ???
\]
We work with objects

\[ f = f_0 + f_1 x + \ldots + f_{n-1} x^{n-1} \in \mathbb{C}[[x]]/\langle x^n \rangle \]

With \( n = 2 \) (first derivatives):

\[ (f_0 + f_1 x) \times (g_0 + g_1 x) = f_0 g_0 + (f_0 g_1 + f_1 g_0) x \]

\[ \frac{1}{f_0 + f_1 x} = \frac{1}{f_0} - \frac{f_1}{f_0^2} x \]

\[ \sin(f_0 + f_1 x) = \sin(f_0) + \cos(f_1) x \]
Formal and non-formal operations

\[
\frac{1}{1 - f} = 1 + f + f^2 + f^3 + \ldots
\]

With \( f = x + x^2 \):

\[
1 + (x + x^2) + (x^2 + 2x^3 + x^4) + (x^3 + 3x^4 + 3x^5 + x^6) + \ldots
\]

With \( f = 0.5 + x \):

\[
1 + (0.5 + x) + (0.25 + x + x^2) + (0.125 + 0.75x + 1.5x^2 + x^3) + \ldots
\]

With \( f = 2 + x^2 \)?
Functions of formal power series

In general, if $F$ is a function and $f$ is a power series, we define

$$F(f) = F(f_0 + x) \circ (f_1 x + f_2 x^2 + \ldots)$$

where

$$F(c + x) = F(c) + F'(c)x + \frac{1}{2}F''(c)x^2 + \ldots$$

and $\circ$ denotes **formal** composition of two power series.

Example:

$$\frac{1}{1 - f} = \left( \frac{1}{1 - f_0} + \frac{x}{(1 - f_0)^2} + \frac{x^2}{(1 - f_0)^3} \ldots \right) \circ (f_1 x + f_2 x^3 + \ldots)$$
For large $n$, **reduce everything to multiplication**

- Classical polynomial multiplication: $M(n) = O(n^2)$ arithmetic operations, $O^{\sim}(n^2 p)$ bit operations
- FFT-based polynomial multiplication: $M(n) = O^{\sim}(n)$ arithmetic operations, $O^{\sim}(np)$ bit operations

**Caveat**: When working with numerical data, we often need precision $p \sim n$, so effective bit complexity with FFT is often $O^{\sim}(n^2)$.

Feasible to work with $n \approx 10^5$ ($\approx 10^{10}$ bits of data).
Bad: truncate
Bad: cover everything
Better: split into blocks
Improving efficiency: scaling

\[ F(x) \times G(x) = H(x) \]
Improving efficiency: scaling

\[ F(2^c x) \times G(2^c x) = H(2^c x) \]
Fast operations on power series

Many operations on power series of length $n$ can all be done in $O(M(n))$ arithmetic operations:

- Multiplication
- Division
- Square root
- Elementary functions (exp, log, sin, ...)
- Solution of some differential equations
Formal Newton iteration

To solve $F(g) = 0$, apply the Newton step

$$g_{\text{new}} = g_{\text{old}} - \frac{F(g_{\text{old}})}{F'(g_{\text{old}})}$$

If $F(g_{\text{old}}) = O(x^n)$, then $F(g_{\text{new}}) = O(x^{2n})$

**Example**: to compute the reciprocal $1/f$ of a power series, solve

$$F(g) = \frac{1}{g} - f$$

which corresponds to the Newton update

$$g_{\text{new}} = 2g_{\text{old}} - fg_{\text{old}}^2.$$
Fast mass computation of Bernoulli numbers

Using the generating function:

\[
\frac{x}{e^x - 1} = \frac{1}{1 + \frac{1}{2}x + \frac{1}{6}x^2 + \ldots} = \sum_{k=0}^{\infty} \frac{B_k}{k!} x^k
\]

Cost to compute \( n \) Bernoulli numbers exactly (using \( \mathbb{Q}[[x]] \) arithmetic):

- \( O(M(n)) = O^\sim(n) \) arithmetic operations
- \( O^\sim(n^2) \) bit operations
A slower (but faster) algorithm for Bernoulli numbers

How to generate $B_0, B_1, \ldots, B_{10000}$ in two seconds!

$$B_{2n} = (-1)^{n+1} \frac{2(2n)!}{(2\pi)^{2n}} \zeta(2n)$$

$\zeta(2) = 1 + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \ldots$

$\zeta(4) = 1 + \frac{1}{2^4} + \frac{1}{3^4} + \frac{1}{4^4} + \ldots$

$\zeta(6) = 1 + \frac{1}{2^6} + \frac{1}{3^6} + \frac{1}{4^6} + \ldots$

Recycle the powers in each column (multiplication by $2^2, 3^2, 4^2, \ldots$)

Complexity is $O^\sim(n^3)$, but in practice an order of magnitude faster than the $O^\sim(n^2)$ power series algorithm. I first saw this algorithm in a blog by Remco Bloemen [http://2pi.com/09/11/even-faster-zeta-calculation](http://2pi.com/09/11/even-faster-zeta-calculation)
Elementary functions of power series

Functional equation + formal composition

\[
\exp(f) = \exp(f_0) \left( \exp(x) \circ (f_1 x + f_2 x^2 + \ldots) \right)
\]

\[
\log(f) = \log(f_0) + \log(1 + x) \circ \left( \frac{xf_1 + x^2f_2 + \ldots}{f_0} \right)
\]

Fast formal composition (\(O(M(n))\) arithmetic operations):

\[
\log(1 + f) = \int \frac{f'}{1 + f}
\]

\[
\tan(f) = \int \frac{f'}{1 + f^2}
\]

Newton iteration gives \(\log \rightarrow \exp, \tan \rightarrow \tan\), etc.
Fast composition of arbitrary formal power series

Horner’s rule

\[ f(g(x)) = f_0 + g(f_1 + g(f_2 + \cdots + g(f_{n-2} + f_{n-1}x) \cdots )) \]

Arithmetic complexity: \( O(nM(n)) \)

Brent-Kung 2.1: baby-step, giant-step version of Horner’s rule
Arithmetic complexity: \( O(n^{1/2}M(n) + n^{1/2}MM(n^{1/2})) \)

Brent-Kung 2.2: divide-and-conquer Taylor expansion
Arithmetic complexity: \( O((n \log n)^{1/2}M(n)) \)
Example with $n = 9$, $m = \lceil \sqrt{n} \rceil = 3$

Computing $f(g)$ where $f = f_0 + f_1 x + \ldots + f_8 x^8$

$$(f_0 + f_1 g + f_2 g^2) + (f_3 + f_4 g + f_5 g^2)g^3 + (f_6 + f_7 g + f_8 g^2)g^6$$

$(m \times m) \times (m \times m^2)$ matrix multiplication:

$$\begin{pmatrix} f_0 & f_1 & f_2 \\ f_3 & f_4 & f_5 \\ f_6 & f_7 & f_8 \end{pmatrix} \times \begin{pmatrix} 1 \\ g \\ g^2 \end{pmatrix}$$

Finally apply Horner’s rule to block polynomials.
General functions of power series

If $f, g$ are holomorphic with

$$f(z) \approx g(z)$$

for all $z$ inside some disk, then also

$$f^{(n)}(z) \approx g^{(n)}(z)$$

$$\zeta(s) \approx \sum_{k=1}^{N} \frac{1}{k^s} \Rightarrow \zeta(s + x) \approx \sum_{k=1}^{N} \frac{1}{k^{s+x}}$$

where $\zeta(s + x) = \zeta(s) + \zeta'(s)x + \frac{1}{2}\zeta''(s)x \ldots$

To compute derivatives of $\zeta(s)$, take your favorite approximate formula and evaluate it at $s + x \in \mathbb{C}[[x]]$ instead of $s$. 
Recall: the Euler-Maclaurin formula

\[ \sum_{k=N}^{U} f(k) = I + T + R \]

\[ I = \int_{N}^{U} f(t) \, dt \]

\[ T = \frac{1}{2} (f(N) + f(U)) + \sum_{k=1}^{M} \frac{B_{2k}}{(2k)!} \left( f^{(2k-1)}(U) - f^{(2k-1)}(N) \right) \]

\[ R = -\int_{N}^{U} \frac{\tilde{B}_{2M}(t)}{(2M)!} f^{(2M)}(t) \, dt \]
Computing $\zeta(s, a)$ using Euler-Maclaurin

$$\zeta(s, a) = \sum_{k=0}^{N-1} f(k) + \sum_{k=N}^{\infty} f(k), \quad f(k) = \frac{1}{(a + k)^s}$$

$S$ $I + T + R$

For derivatives, substitute $s \rightarrow s + x \in \mathbb{C}[[x]]$:

$$f(k) = \frac{1}{(a + k)^{s+x}} = \sum_{i=0}^{\infty} \frac{(-1)^i \log^i(a + k)}{i!(a + k)^s} x^i \in \mathbb{C}[[x]]$$
The terms

\[ f(1) = \left[ \frac{1}{(a+1)^s} \right] + \left[ -\frac{\log(a + 1)}{(a+1)^s} \right] x + \left[ \frac{\log^2(a + 1)}{2(a+1)^s} \right] x^2 + \ldots \]

\[ f(2) = \left[ \frac{1}{(a+2)^s} \right] + \left[ -\frac{\log(a + 2)}{(a+2)^s} \right] x + \left[ \frac{\log^2(a + 2)}{2(a+2)^s} \right] x^2 + \ldots \]

\[ f(3) = \left[ \frac{1}{(a+3)^s} \right] + \left[ -\frac{\log(a + 3)}{(a+3)^s} \right] x + \left[ \frac{\log^2(a + 3)}{2(a+3)^s} \right] x^2 + \ldots \]
Parts to evaluate

\[ S = \sum_{k=0}^{N-1} \frac{1}{(a + k)^{s+x}} \]

\[ I = \int_N^\infty \frac{1}{(a + t)^{s+x}} \, dt = \frac{(a + N)^{1-(s+x)}}{(s + x) - 1} \]

\[ T = \frac{1}{(a + N)^{s+x}} \left( \frac{1}{2} + \sum_{k=1}^{M} \frac{B_{2k}}{(2k)!} \frac{(s + x)_{2k-1}}{(a + N)^{2k-1}} \right) \]

\[ R = -\int_N^\infty \frac{\tilde{B}_{2M}(t)}{(2M)!} \frac{(s + x)_{2M}}{(a + t)^{(s+x)+2M}} \, dt \quad \text{(bound)} \]
Bounding the remainder

Define $|f| = |f_0| + |f_1|x + |f_2|x^2 + \ldots$

Then $|f + g| \leq |f| + |g|$ and $|fg| \leq |f||g|$ (coefficient-wise)

\[
|R| = \left| \int_N^\infty \tilde{B}_{2M}(t) \frac{(s + x)_{2M}}{(2M)!} \frac{(s + x)_{2M}}{(a + t)^{s+x+2M}} \, dt \right|
\]

\[
\leq \int_N^\infty \left| \tilde{B}_{2M}(t) \frac{(s + x)_{2M}}{(2M)!} \frac{(a + t)^{s+x+2M}}{(a + t)^{s+x+2M}} \right| \, dt
\]

\[
\leq 4 \left| (s + x)_{2M} \right| \frac{4}{(2\pi)^{2M}} \int_N^\infty \frac{dt}{(a + t)^{s+x+2M}} \in \mathbb{R}[[x]]
\]

\[
\int_N^\infty \frac{dt}{(a + t)^{s+x+2M}} = \sum_{k=0}^\infty \left( \int_N^\infty \frac{dt}{k!} \left| \log(a + t)^k \right| \frac{(a + t)^{s+2M}}{(a + t)^{s+2M}} \right) x^k
\]
A sequence of integrals

For $k \in \mathbb{N}$, $A > 0$, $B > 1$, $C \geq 0$,

$$J_k(A, B, C) \equiv \int_{A}^{\infty} t^{-B}(C + \log t)^k dt$$

$$= \frac{L_k}{(B - 1)^{k+1}A^{B-1}}$$

where

$$L_0 = 1, \quad L_k = kL_{k-1} + D^k$$

$$D = (B - 1)(C + \log A)$$
Error bound

Given complex numbers \( s = \sigma + \tau i, \ a = \alpha + \beta i \) and positive integers \( N, M \) such that \( \alpha + N > 1 \) and \( \sigma + 2M > 1 \), the error term in the Euler-Maclaurin summation formula applied to \( \zeta(s + x, a) \in \mathbb{C}[[x]] \) satisfies

\[
|R(s + x)| \leq \frac{4 |(s + x)_{2M}|}{(2\pi)^{2M}} \left| \sum_{k=0}^{\infty} R_k x^k \right| \in \mathbb{R}[[x]]
\]

where \( R_k \leq (K/k!) J_k(N + \alpha, \sigma + 2M, C) \), with

\[
C = \frac{1}{2} \log \left( 1 + \frac{\beta^2}{(\alpha + N)^2} \right) + \tan \left( \frac{|\beta|}{\alpha + N} \right)
\]

and

\[
K = \exp \left( \max \left( 0, \tau \tan \left( \frac{\beta}{\alpha + N} \right) \right) \right).
\]
Evaluation steps

To evaluate $\zeta(s + x, a)$ with an error of $2^{-p}$:

1. Choose $N, M = O(p)$, bound the error term $R$
2. Compute the power sum $S$
3. Compute the integral $I$
4. Compute the Bernoulli numbers
5. Compute the tail $T$
Some computational results
Stieltjes constants

The (generalized) Stieltjes constants are the coefficients $\gamma_n(a)$ in the Laurent series

$$\zeta(s, a) = \frac{1}{s-1} + \sum_{n=0}^{\infty} \frac{(-1)^n}{n!} \gamma_n(a) (s-1)^n.$$ 

The (classical) Stieltjes constants $\gamma_n = \gamma_n(1)$ have numerical values

$$\begin{align*}
\gamma_0 & \approx +0.577216 & \gamma_{10} & \approx +0.000205 \\
\gamma_1 & \approx -0.072816 & \gamma_{100} & \approx -4.25340 \times 10^{17} \\
\gamma_2 & \approx -0.009690 & \gamma_{1000} & \approx -1.57095 \times 10^{486}
\end{align*}$$
Asymptotics of Stieltjes constants

One of the best available bounds for $\gamma_n$ is [Matsuoka, 1985]:

$$|\gamma_n| < 0.0001e^{n\log\log n}, \quad n \geq 10$$

But this is not very accurate.

Actual value: $\gamma_{1000} \approx -1.57095 \times 10^{486}$

Matsuoka: $|\gamma_{1000}| < 2.17242 \times 10^{835}$
Knessl-Coffey approximation


\[ \gamma_n \sim \frac{B}{\sqrt{n}} e^{nA} \cos(an + b) \]

\[ A = \frac{1}{2} \log(u^2 + v^2) - \frac{u}{u^2 + v^2}, \quad B = \frac{2\sqrt{2\pi}\sqrt{u^2 + v^2}}{[u + 1]^2 + v^2]^{1/4}} \]

\[ a = \tan^{-1}\left(\frac{v}{u}\right) + \frac{v}{u^2 + v^2}, \quad b = \tan^{-1}\left(\frac{v}{u}\right) - \frac{1}{2} \left(\frac{v}{u + 1}\right) \]

where \( u = v \tan v \), and \( v \) is the unique solution of

\[ 2\pi \exp(v \tan v) = \left(\frac{n}{v}\right) \cos(v), \quad 0 < v < \pi/2. \]

Similar formula for \( \gamma_n(a), \ a \neq 1. \)

- Predicts sign oscillations (correct except for \( n = 137? \))
- Accurate even for small \( n \)
- No explicit error bound
Computations of Stieltjes constants

[Kreminski, *Newton-Cotes integration for approximating Stieltjes (generalized Euler) constants*, 2003]:

- Values up to $n = 8300$ with 200 significant digits
- Heuristic error estimates

Later heuristic computations up to about $n = 35000$

[FJ, 2014]:

- All $\gamma_n$ up to $n = 100000$ with more than 10000 digits each
- Rigorous error bounds
- To replicate, simply call the Hurwitz zeta function of a power series in Arb (also for $\gamma_n(a)$)

Data available from:
http://fredrikj.net/math/hurwitz_zeta.html
Numerical values

Computed value of $\gamma_{100000}$:

$$1.99192730631254109565 \ldots \times 10^{83432}$$

Knessl-Coffey approximation:

$$1.9919333 \times 10^{83432}$$

Matsuoka bound: $3.71 \times 10^{106114}$

Computed value of $\lambda_{50000}(1 + i)$:

$$(1.032502087431 \ldots - 1.441962552840 \ldots i) \times 10^{39732}$$

Knessl-Coffey approximation:

$$(1.0324943 - 1.4419586i) \times 10^{39732}$$
Relative error of Knessl-Coffey formula
Relative error of Knessl-Coffey formula
The Keiper-Li coefficients

Define \( \{\lambda_n\}_{n=1}^{\infty} \) by

\[
\log \xi \left( \frac{1}{1-x} \right) = \log \xi \left( \frac{x}{x-1} \right) = -\log 2 + \sum_{n=1}^{\infty} \lambda_n x^n
\]

where \( \xi(s) = \frac{1}{2} s(s-1) \pi^{-s/2} \Gamma(s/2) \zeta(s) \).

Keiper (1992): Riemann hypothesis \( \Rightarrow \forall n : \lambda_n > 0 \)
Li (1997): Riemann hypothesis \( \Leftarrow \forall n : \lambda_n > 0 \)

Keiper conjectured \( 2\lambda_n \approx (\log n - \log(2\pi) + \gamma - 1) \)
Evaluating the Keiper-Li coefficients

Evaluate

\[ \log \xi(s) = \log(-\zeta(s)) + \log \Gamma(1 + s/2) + \log(1 - s) - s \log(\pi)/2 \]

at \( s = x \in \mathbb{R}[[x]] \)

Ingredients:
1. The series expansion \( \zeta(s + x) \) at \( s = 0 \)
2. The logarithm of a power series: \( \log f(x) = \int \frac{f'(x)}{f(x)} dx \)
3. The series \( \log \Gamma(1 + x) \), essentially \( \gamma, \zeta(2), \zeta(3), \zeta(4), \ldots \)
4. Right-composing by \( x/(x - 1) \)

A working precision of \( \approx n \) bits is needed to get an accurate value for \( \lambda_n \).
Fast composition

The binomial transform of \( f = \sum_{k=0}^{\infty} a_k x^k \) is

\[
T[f] = \frac{1}{1-x} f \left( \frac{x}{x-1} \right) = \sum_{n=0}^{\infty} \left( \sum_{k=0}^{n} (-1)^k \binom{n}{k} a_k \right) x^n
\]

and the Borel transform is

\[
B[f] = \sum_{k=0}^{\infty} \frac{a_k}{k!} x^k.
\]

\[
T[f(x)] = B^{-1}[e^x B[f(-x)]], \text{ so we get } f \left( \frac{x}{x-1} \right) \text{ by a single power series multiplication!}
\]
Values of Keiper-Li coefficients

I have computed rigorous values of all $\lambda_n$ up to $n = 100000$ (using Arb, and 110000 bits of working precision). In particular,

$$\lambda_{100000} = 4.62580782406902231409416038 \ldots$$

plus about 2900 more accurate digits.

Keiper’s approximation suggests $\lambda_{100000} \approx 4.626132$.

See examples/keiper_li.c in Arb and http://fredrikj.net/math/hurwitz_zeta.html for data
Comparison with approximation formula

Plot of $\lambda_n$ and $\lambda_n - (\log n - \log(2\pi) + \gamma - 1) / 2$
Comparison with approximation formula

Plot of \( n \left( \lambda_n - \left( \log n - \log(2\pi) + \gamma - 1 \right) / 2 \right) \).
Time to compute Keiper-Li coefficients

(In seconds)

<table>
<thead>
<tr>
<th></th>
<th>(n = 1000)</th>
<th>(n = 10000)</th>
<th>(n = 100000)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error bound (R)</td>
<td>0.017</td>
<td>1.0</td>
<td>97</td>
</tr>
<tr>
<td>Power sum (S)</td>
<td>0.048</td>
<td>47</td>
<td>65402</td>
</tr>
<tr>
<td>(CPU time)</td>
<td>(0.65)</td>
<td>(693)</td>
<td>(1042210)</td>
</tr>
<tr>
<td>Bernoulli numbers</td>
<td>0.0020</td>
<td>0.19</td>
<td>59</td>
</tr>
<tr>
<td>Tail (T)</td>
<td>0.058</td>
<td>11</td>
<td>1972</td>
</tr>
<tr>
<td>Logarithm of power series</td>
<td>0.047</td>
<td>8.5</td>
<td>1126</td>
</tr>
<tr>
<td>(\log \Gamma(1 + x))</td>
<td>0.019</td>
<td>3.0</td>
<td>1610</td>
</tr>
<tr>
<td>Composition by (x/(x - 1))</td>
<td>0.022</td>
<td>4.1</td>
<td>593</td>
</tr>
<tr>
<td>Total wall time</td>
<td>0.23</td>
<td>84</td>
<td>71051</td>
</tr>
<tr>
<td>RAM (MiB)</td>
<td>8</td>
<td>730</td>
<td>48700</td>
</tr>
</tbody>
</table>

Bit complexity is \(O^\sim(n^2)\) except for the power sum which is \(O^\sim(n^3)\) (\(O^\sim(n^2)\) is possible in theory, but not implemented)