Accurate Multiple-Precision Gauss-Legendre Quadrature

Laurent Fousse[1,2]

[1]Université Henri-Poincaré Nancy 1

[2]INRIA Rocquencourt

18th IEEE International Symposium on Computer Arithmetic

Outline

Quadrature is ubiquitous

- In physics (crystallography [Jézéquel Chesneaux 2004]):

$$g(a) = \int_0^{+\infty} [(\exp(x) + \exp(-x))^a - \exp(ax) - \exp(-ax)] \, \mathrm{d}x$$

where $0 < a < 2$ and $g(a)$ is the crystal energy.

- probabilities [Trefethen, Kern 2002]

$$2 = \frac{1}{\pi} \int_{-\pi}^{\pi} \frac{\mathrm{d}\phi}{\sqrt{3 - 4\cos\phi + \cos^2\phi + 16\epsilon^2}}$$

(parameterized random walk in $\mathbb{Z}^2$).

Quadrature is hard

At least you can fail spectacularly.
One example with Pari/GP 2.3.0: $\int_0^{10} x^2 \sin(x^3)\mathrm{d}x$

```
?default(realprecision,115);intnum(x=0,10,x^2*sin(x^3))
%2 = 16.28927300548418203118991830[...]
```

Quadrature is hard

At least you can fail spectacularly.

One example with Pari/GP 2.3.0: $\int_0^{10} x^2 \sin(x^3) \mathrm{d}x$

```
?default(realprecision,115);intnum(x=0,10,x^2*sin(x^3))
%2 = 16.2892730054841820311899183O[...]
```

```
?default(realprecision,116);intnum(x=0,10,x^2*sin(x^3))
%2 = 0.145873643233630725025779820[...]
```

Which result do you trust?

Quadrature is hard

At least you can fail spectacularly.
One example with Pari/GP 2.3.0: $\int_0^{10} x^2 \sin(x^3) \mathrm{d}x$

```
?default(realprecision,115);intnum(x=0,10,x^2*sin(x^3))
%2 = 16.28927300548418203118991830[...]
```

```
?default(realprecision,116);intnum(x=0,10,x^2*sin(x^3))
%2 = 0.1458736432336307250257798 20[...]
```

Which result do you trust?

$$\int_0^{10} x^2 \sin(x^3) \mathrm{d}x = \frac{1}{3}(1 - \cos(1000)) \approx 0,1458736432336$$

Quadrature is hard

A famous computer algebra software:

```
    |\^/|     Maple 10 (IBM INTEL LINUX)
._|\|   |/|_. Copyright (c) Maplesoft, a division of Waterloo Maple Inc. 2005
 \  MAPLE  /  All rights reserved. Maple is a trademark of
 <____ ____>  Waterloo Maple Inc.
      |       Type ? for help.
> evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                           -126
                     0.2604007480 10
```

Quadrature is hard

A famous computer algebra software:

```
    |\^/|     Maple 10 (IBM INTEL LINUX)
._|\|   |/|_. Copyright (c) Maplesoft, a division of Waterloo Maple Inc. 2005
 \  MAPLE  /  All rights reserved. Maple is a trademark of
 <____ ____>  Waterloo Maple Inc.
      |       Type ? for help.
> evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                           -126
                0.2604007480 10
> Digits:=20: evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                                     -126
                0.34288028340847034512 10
```

Quadrature is hard

A famous computer algebra software:

```
    |\^/|     Maple 10 (IBM INTEL LINUX)
._|\|   |/|_. Copyright (c) Maplesoft, a division of Waterloo Maple Inc. 2005
 \  MAPLE  /  All rights reserved. Maple is a trademark of
 <____ ____>  Waterloo Maple Inc.
      |       Type ? for help.
> evalf(Int(exp(-x^2)*ln(x), x=17..42));
                             -126
           0.2604007480 10
> Digits:=20: evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                       -126
           0.34288028340847034512 10
> Digits:=50: evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                                                -128
           0.49076783443012876473973482836733778547443399549250 10
```

Quadrature is hard

A famous computer algebra software:

```
    |\^/|     Maple 10 (IBM INTEL LINUX)
._|\|   |/|_. Copyright (c) Maplesoft, a division of Waterloo Maple Inc. 2005
 \  MAPLE  /  All rights reserved. Maple is a trademark of
 <____ ____>  Waterloo Maple Inc.
      |       Type ? for help.
> evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                                -126
                     0.2604007480 10
> Digits:=20: evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                                          -126
                     0.34288028340847034512 10
> Digits:=50: evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                                                                     -128
                     0.49076783443012876473973482836733778547443399549250 10
> Digits:=100: evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                                                                          -128
                     0.49076783443012876473973482836733778547443399549250[...] 10
```

Quadrature is hard

$$\int_{17}^{42} \exp(-x^2) \log x \mathrm{d}x \approx 0,2565728500 \times 10^{-126}$$

Precision	Maple answer	Correct digits
10	$0,2604007480 \times 10^{-126}$	$1,8$
20	$0,3428802834 \times 10^{-126}$	$0,6$
50	$0,4907678344 \times 10^{-128}$	$-1,7$
100	$0,4907678344 \times 10^{-128}$	$-1,7$

Miscomputations are bad

- Ariane 5

Miscomputations are bad

- Ariane 5
- Patriot anti-missile

Miscomputations are bad

- Ariane 5
- Patriot anti-missile
- Vancouver Stock Exchange

Miscomputations are bad

Every time you miscompute

a kitten gets killed.

Please, think of the kitten!

$\Longrightarrow$ mimic the IEEE 754 standard as closely as possible for numerical integration.

Gauss-Legendre Quadrature in short

- rule of interpolatory type:

$$\int_a^b f(x)\mathrm{d}x \approx \int_a^b \sum_{i=0}^{n-1} f(x_i) \frac{\prod x - x_j}{\prod x_i - x_j} = \sum_{i=0}^{n-1} w_i \cdot f(x_i)$$

- the x_i's are the roots of the n-th Legendre polynomial P_n
- the weights w_i are easily computed

$$w_i = \frac{2}{(1 - x_i^2) P_n'^2(x_i)}.$$

- mathematical error bounded by

$$\frac{(b-a)^{2n+1}(n!)^4}{(2n+1)[(2n)!]^3} ||f^{(2n)}||_\infty.$$

Gauss-Legendre Quadrature in CRQ

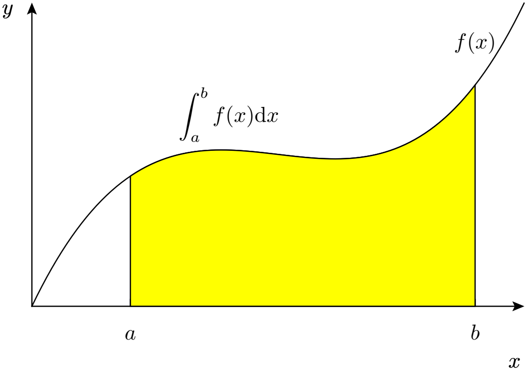
Correctly **R**ounded **Q**uadrature

- Compute P_n via

$$(n+1)P_{n+1}(X) = (2n+1)XP_n(X) - nP_{n-1}(X)$$

- Root isolation (Uspensky) and refinement (interval Newton)
- Forward error analysis to derive bounds and their proof

Hypothesis



- f is a C^∞ black-box function in arbitrary precision.
- bounds M_k on $|f^{(k)}|$ are known.

Quadrature Algorithm

INPUTS: $\widehat{a}, \widehat{b-a}, (\widehat{w}_i), f, (\widehat{v}_i), n$ where $\widehat{v}_i = \circ(\frac{1+x_i}{2})$.

OUTPUT: $\widehat{I} \approx \int_a^b f(x)\mathrm{d}x$

for $i \leftarrow 0 \text{ to } n-1$ **do**

  $t \leftarrow \circ((\widehat{b-a}) \cdot \widehat{v}_i)$

  $\widehat{x}_i \leftarrow \circ(t + \widehat{a})$

  $\widehat{f}_i \leftarrow \circ(f(\widehat{x}_i))$

  $\widehat{y}_i \leftarrow \circ(\widehat{f}_i \cdot \widehat{w}_i)$

end for

$\widehat{S} \leftarrow \mathrm{sum}(\widehat{y}_i, i = 0 \ldots n-1)$

$\widehat{D} \leftarrow \circ(\widehat{b-a})/2$

return $\circ(\widehat{D}\widehat{S}) = \widehat{I}$

Theorem

Theorem

Let $\delta_{\widehat{y}_i} = \frac{11}{4}\mathrm{ulp}(\widehat{y}_i) + 6M_1\widehat{w}_i\mathrm{ulp}(\widehat{x}_i)$, $p \geqslant 2$ *the working precision. With n integration points the error* $|\widehat{I} - I|$ *is bounded by:*

$$B_{total} = \frac{21}{4}\mathrm{ulp}(\widehat{I}) + \frac{5n}{4}\widehat{D} \cdot \max(\delta_{\widehat{y}_i}) + \frac{(b-a)^{2n+1}(n!)^4}{(2n+1)[(2n)!]^3}M_{2n}.$$

A glimpse of the proof

Let $v_i = \frac{1+x_i}{2}$

$$\begin{aligned} |\widehat{v_i} - v_i| &\leq \frac{1}{2}\mathrm{ulp}(\widehat{v_i}) \\ \widehat{x_i} &= \circ(\circ(\widehat{v_i} \cdot \widehat{(b-a)}) + \widehat{a}). \end{aligned}$$

Then

$$| \circ (\widehat{v_i} \cdot \widehat{b-a}) - v_i \cdot (b-a)| \leq \frac{5}{2}\mathrm{ulp}(\circ(\widehat{v_i} \cdot \widehat{b-a}))$$

A glimpse of the proof

Let $v_i = \frac{1+x_i}{2}$

$$\begin{aligned} |\widehat{v_i} - v_i| &\leq \frac{1}{2}\mathrm{ulp}(\widehat{v_i}) \\ \widehat{x_i} &= \circ(\circ(\widehat{v_i} \cdot (\widehat{b-a})) + \widehat{a}). \end{aligned}$$

Then

$$\begin{aligned} |\widehat{x_i} - x_i| &\leq \frac{1}{2}\mathrm{ulp}(\widehat{x_i}) + \frac{5}{2}\mathrm{ulp}(\circ(\widehat{v_i} \cdot \widehat{b-a})) + \frac{1}{2}\mathrm{ulp}(\widehat{a}) \\ &\leq \frac{17}{4}\mathrm{ulp}(\widehat{x_i}). \end{aligned}$$

A glimpse of the proof

Let $v_i = \frac{1+x_i}{2}$

$$
\begin{aligned}
|\widehat{v_i} - v_i| &\leq \frac{1}{2}\mathrm{ulp}(\widehat{v_i}) \\
\widehat{x_i} &= \circ(\circ(\widehat{v_i} \cdot (\widehat{b - a})) + \widehat{a}).
\end{aligned}
$$

Then

$$
\begin{aligned}
|\widehat{x_i} - x_i| &\leq \frac{1}{2}\mathrm{ulp}(\widehat{x_i}) + \frac{5}{2}\mathrm{ulp}(\circ(\widehat{v_i} \cdot \widehat{b - a})) + \frac{1}{2}\mathrm{ulp}(\widehat{a}) \\
&\leq \frac{17}{4}\mathrm{ulp}(\widehat{x_i}).
\end{aligned}
$$

$$
|\circ(f(\hat{x}_i)) - f(x_i)| \leq |\circ(f(\hat{x}_i)) - f(\hat{x}_i)| + |f(\hat{x}_i) - f(x_i)|
$$

A glimpse of the proof

Let $v_i = \frac{1+x_i}{2}$

$$\begin{aligned} |\widehat{x_i} - x_i| &\leq \frac{1}{2}\mathrm{ulp}(\widehat{x_i}) + \frac{5}{2}\mathrm{ulp}(\circ(\widehat{v_i} \cdot \widehat{b-a})) + \frac{1}{2}\mathrm{ulp}(\widehat{a}) \\ &\leq \frac{17}{4}\mathrm{ulp}(\widehat{x_i}). \end{aligned}$$

$$\begin{aligned} |\circ(f(\hat{x}_i)) - f(x_i)| &\leq |\circ(f(\hat{x}_i)) - f(\hat{x}_i)| + |f(\hat{x}_i) - f(x_i)| \\ &\leq \mathrm{ulp}(\circ(f(\hat{x}_i))) + M_1|\hat{x}_i - x_i| \end{aligned}$$

A glimpse of the proof

Let $v_i = \frac{1+x_i}{2}$

$$
\begin{aligned}
|\widehat{x_i} - x_i| &\leq \frac{1}{2}\mathrm{ulp}(\widehat{x_i}) + \frac{5}{2}\mathrm{ulp}(\circ(\widehat{v_i} \cdot \widehat{b - a})) + \frac{1}{2}\mathrm{ulp}(\widehat{a}) \\
&\leq \frac{17}{4}\mathrm{ulp}(\widehat{x_i}).
\end{aligned}
$$

$$
\begin{aligned}
|\circ(f(\hat{x}_i)) - f(x_i)| &\leq |\circ(f(\hat{x}_i)) - f(\hat{x}_i)| + |f(\hat{x}_i) - f(x_i)| \\
&\leq \mathrm{ulp}(\circ(f(\hat{x}_i))) + M_1|\hat{x}_i - x_i| \\
&\leq \mathrm{ulp}(\circ(f(\hat{x}_i))) + \frac{17}{4}M_1\mathrm{ulp}(\hat{x}_i)
\end{aligned}
$$

Our favorite example

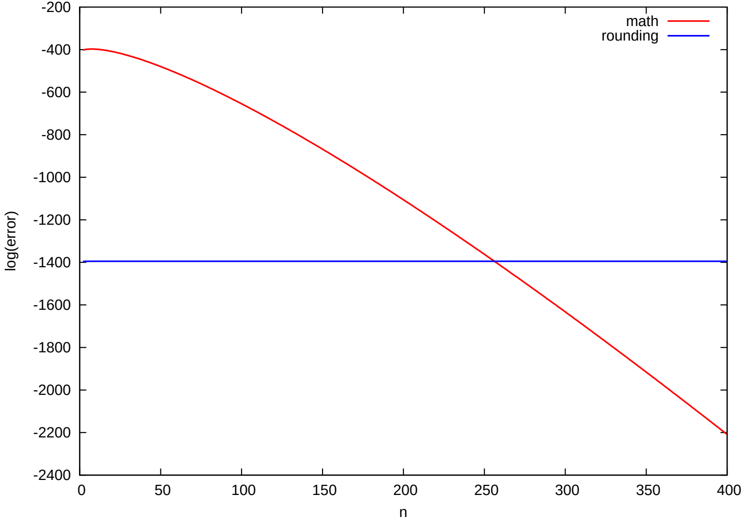
$$I = \int_{17}^{42} e^{-x^2} \log x \mathrm{d}x$$

$$\begin{aligned} g(x) &= e^{-x^2} \\ h(x) &= \log x \\ f^{(n)}(x) &= \sum_{i=0}^{n} \binom{n}{i} \frac{\mathrm{d}^i}{\mathrm{d}x^i} g(x) \frac{\mathrm{d}^{n-i}}{\mathrm{d}x^{n-i}} h(x) \end{aligned}$$

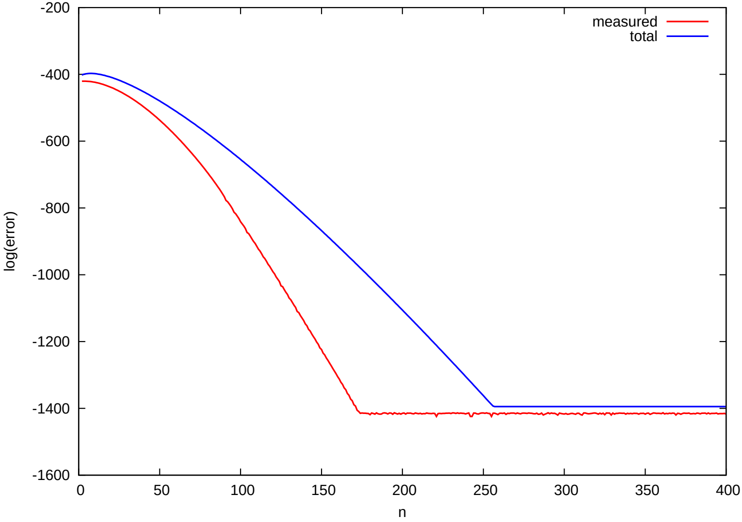
After some computations:

$$|f^{(n)}| \leq n \cdot n! e^{-17^2} \left((n+1) 42^n \log\ 42 + (n-1) 42^{n-2} \right).$$

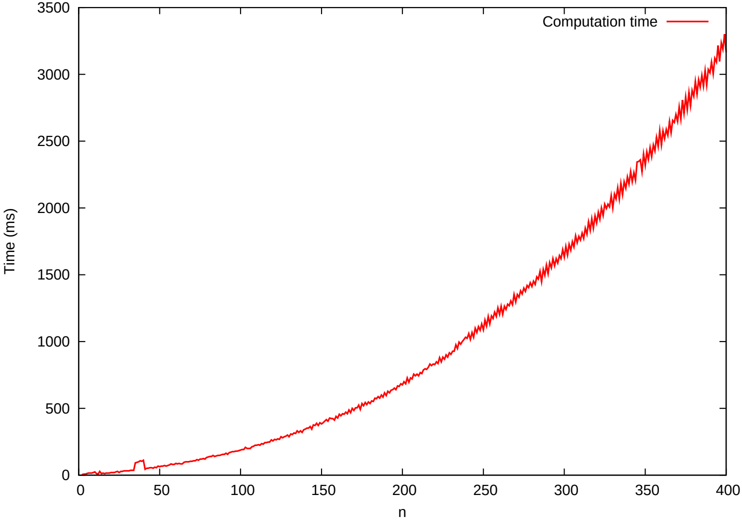
B_{math} vs. B_{rounding} ($p = 1000$)



Measured error vs. computed error bound ($p = 1000$)



Quadrature coefficients computation time



CRQ vs. Rest of the World

$$I = \int_0^1 \max(\sin(x), \cos(x)) \mathrm{d}x = \sqrt{2} - \cos 1$$

Precision		31	61	151	302	603	1506
CRQ	Time	26ms	110ms	1s	11s	108s	3138s
	Error	1	2.6e-2	9.2e-1	3.6	4.2	3.5
Maple	Time	6s	38s	371s	2618s	–	–
	Error	2.1e-1	1.7e-1	1.7e-1	2.9e-1	–	–
PARI/GP	Time	20ms	80ms	480ms	3s	22s	301s
	Error	3.1e25	2.0e54	2.7e144	1.0e295	1.7e595	9.2e1496

Summary

- Multiple-precision quadrature can be fast. . .

Summary

- Multiple-precision quadrature can be fast. . .
- and certified at the same time.

Summary

- Multiple-precision quadrature can be fast. . .
- and certified at the same time.
- Code available at
  `http://komite.net/laurent/soft/crq/`

Summary

- Multiple-precision quadrature can be fast. . .
- and certified at the same time.
- Code available at `http://komite.net/laurent/soft/crq/`

TODOs:

- Speed-up Gauss-Legendre coefficients computation (parallel Newton iteration?),
- Look at other methods in the Gauss family,
- Automatic computation of bounds for specific functions,
- Optimal composition strategy.