High-Accurate Computation of One-Loop Integrals by Several Hundred Digits Multiple-Precision Arithmetic

Hiroshi Fujiwara
(藤原 宏志)

Graduate School of Informatics, Kyoto University

CPP2010 Workshop
KEK Japan,   24 September 2010

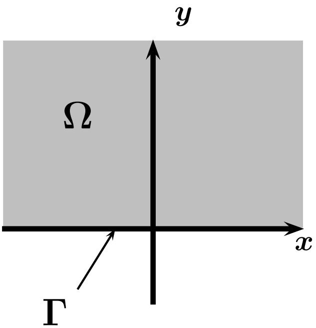
Contents

1. Motivation of Multiple-Precision Arithmetic
2. Multiple-Precision Arithmetic Library "exflib"
3. Application to One-Loop Integrals (Box type)

1. Motivation of Multiple-Precision Arithmetic

Motivated Example (Ill-Posed Problem)

A Cauchy problem of the Laplace equation



$$\Omega = \{y > 0\}$$

$$\Gamma = \{y = 0\}$$

$$u = u(x, y)$$

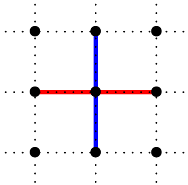
$$\begin{aligned} \triangle u &= 0 \quad \text{in} \quad \Omega \\ u(x, 0) &= x^5 \quad \text{on} \quad \Gamma \\ \frac{\partial u}{\partial y}(x, 0) &= 0 \quad \text{on} \quad \Gamma \end{aligned}$$

Exact Solution:

$$u(x, y) = x^5 - 10x^3y^2 + 5xy^4$$

Finite Difference Method

$$x_i = i\Delta x, \quad y_j = j\Delta y$$



$\Delta x, \Delta y$	:	mesh-size
$u_{i,j}$	$\approx$	$u(x_i, y_j)$

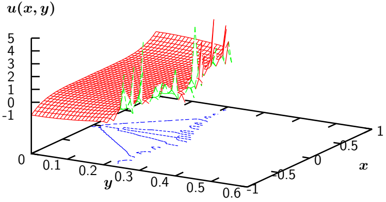
$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2} = 0$$

$$u_{i,0} = (x_i)^5$$

$$\frac{u_{i,1} - u_{i,0}}{\Delta y} = 0$$

Numerical Results ($0 \leq t \leq 0.6$)

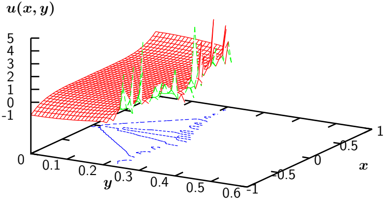
$$\Delta y = \Delta x = 1/100$$



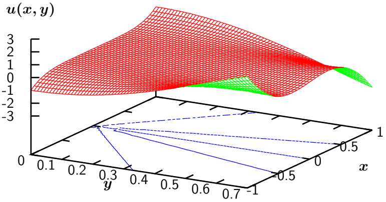
double (16 decimal digits)

Numerical Results ($0 \le t \le 0.6$)

$$\Delta y = \Delta x = 1/100$$



double (16 decimal digits)



100 decimal digits

Numerically Unstable Scheme $\longrightarrow$ Double precision is not enough.

Partial Differential Eq.

$$Pu = f$$

Discretization

Real Number

$$P_h u_h = f_h$$

Partial Differential Eq.

$$Pu = f$$

Discretization

Real Number

$$P_h u_h = f_h$$

Numerical Process

Floating-Point Arithmetic

$$P_{h,p} u_{h,p} = f_{h,p}$$

Numerical Error

$$\|u - u_{h,p}\| \leq \|u - u_h\| + \|u_h - u_{h,p}\|$$

- $\|u - u_h\|$: discretization error $\longleftarrow$ high-order discretization
- $\|u_h - u_{h,p}\|$: rounding error $\longleftarrow$ multiple-precision arithmetic

Floating-Point Arithmetic and Rounding Error

Rounding and Rounding Error

$$\pi \approx (-1)^0 \times 10^0 \times 3.14159$$

- $1.00000 \times 10^0 \oplus 1.00000 \times 10^{-100} = 1.00000 \times 10^0$
- $1.23456 \times 10^0 \ominus 1.23455 \times 10^0 = 1.00000 \times 10^{-6}$ (cancellation)

Rounding and Rounding Error

$$\pi \approx (-1)^0 \times 10^0 \times 3.14159$$

- $1.00000 \times 10^0 \oplus 1.00000 \times 10^{-100} = 1.00000 \times 10^0$
- $1.23456 \times 10^0 \ominus 1.23455 \times 10^0 = 1.00000 \times 10^{-6}$ (cancellation)

IEEE754 double format : 15.95 decimal digits precision

Gap in Arithmetic and Finite Precision Arithmetic

$$a^2 + b^2 \geq 2ab.$$

4 decimal digits computation, and let $a = 1.022, b = 1.038$.

$$a^2 = 1.044484 \longrightarrow 1.044$$
$$b^2 = 1.077444 \longrightarrow 1.077$$

$$a^2 + b^2 \longrightarrow 1.044 + 1.077 = 2.121$$

Gap in Arithmetic and Finite Precision Arithmetic

$$a^2 + b^2 \geq 2ab.$$

4 decimal digits computation, and let $a = 1.022, b = 1.038$.

$$a^2 = 1.044484 \longrightarrow 1.044$$
$$b^2 = 1.077444 \longrightarrow 1.077$$

$$a^2 + b^2 \longrightarrow 1.044 + 1.077 = 2.121$$
$$2ab = 2.044 \times 1.038 = 2.121672 \longrightarrow 2.122$$

For a, b, we have

$$(a \otimes a) \oplus (b \otimes b) < 2 \otimes a \otimes b$$

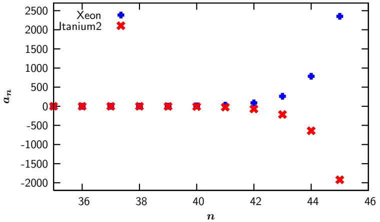
$$\frac{1}{xy + zw - 2ts} \text{ when } x = y = t, z = w = s \text{ ?}$$

Example of Numerical Instability

$$a_{n+2} = \frac{34}{11} a_{n+1} - \frac{3}{11} a_n \qquad a_n = \left(\frac{1}{11}\right)^n$$

$$a_0 = 1, \quad a_1 = \frac{1}{11}$$

Numerical results are different on computer enrivonments.



n	Xeon(Linux)	Itanium2
2	0.00826446	0.00826446
5	6.20921×10^{-6}	6.20921×10^{-6}
10	3.86014×10^{-11}	3.85159×10^{-11}
40	9.68369	-7.91327
50	571812	-467270

$(3^{50} \times 10^{-16} \approx 7.2 \times 10^7)$.

Rounding error grows rapidly, and double precision is not enough.

2. Multiple-Precision Arithmetic Library “exflib”

exflib – extended floating-point arithmetic library

Aimed For Scientific Computing, Computational Mechanics

- Fast computation in 100–1000 decimal digits
  Implementation and optimization in assembly language.
  Suitable arithmetic algorithm
- Large scale computation
  Saving memory
  MPI/OpenMP
- Simple Interface
  Fortran 90, programming language C++
  Seamless interface as built-in types, functions

exflib : extended precision floating-point arithmetic library

http://www-an.acs.i.kyoto-u.ac.jp/~fujiwara/exflib

Key features

- 60 – 20,000 decimal digit arithmetic
- range : $\mathbf{10^{-10^{18}}} \ - \ \mathbf{10^{10^{18}}}$
- basic four rules, built-in functions
- polymorphic interface in C++, FORTRAN90
- instruction level optimization in assembly language

Available on

- Opteron, Athlon64 Xeon, Core, Atom, Pentium (AMD64, Intel64 IA32)
- Solaris, Linux, MacOSX (64bit), Windows (32bit)
- C++, FORTRAN90
- MPI, OpenMP

Data Structure of exfloat

64-bit unsigned integer array

s	e_b	f_1	f_2	f_3	$\cdots$	f_n
1	63	64	64	64		64

$$(-1)^s \times 2^{e_b - \mathrm{BIAS}} \times \left(1 + \sum_{i=1}^{n} \frac{f_i}{2^{64i}}\right)$$

$1 + 64n$ bits $\approx 19.27 \times n$ decimal digits

$n = 6$	115.62 digits
$n = 11$	211.97 digits
$n = 16$	308.32 digits

Arithmetic Benchmark (2007 Jan)

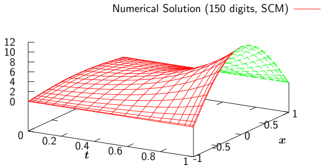
64-bit, Linux, GCC 3 unit : μ sec.

	Opteron150 2.4GHz			Opteron246 2GHz			
	mpfun90	FMLIB	Maple	Mathematica	MPFR	Pari	exflib
digits, op	FORTRAN90				C	C	C
100, mul	3.0	1.1	22	1.5	0.24	0.29	0.14
div	3.5	1.4	26	5.9	0.52	1.2	0.33
1000, mul	60	17	100	17	6.2	15	3.7
div	63	20	93	50	16	23	8.3
10000, mul	–	1105	3400	580	300	1400	280
div	–	1251	3000	1800	730	1500	630

cf. `http://www.medicis.polytechnique.fr/~pphd/mpfr/timings-220.html`

Benchmark : Backward Heat Equation

$$\frac{\partial u}{\partial t} = -\frac{\partial^2 u}{\partial x^2}$$
$$u(0, x) = \cos\frac{\pi}{2}$$
$$u(t, -1) = u(t, 1) = 0.$$



The problem is ill-posed in the sense of Hadamard.

Discretization by the Spectral Method $\longrightarrow$ Linear Equation

env. A : FMLIB, Xeon (2.0GHz) x 1, memory 4GB, 1 procs
env. B : FMLIB, Xeon(2.0GHz) 6 CPU + Xeon(2.4GHz) 2CPU, total memory 16GB, 10 procs Prallel Comp
env. C : exflib, Opteron250 (2.4GHz), 1 procs

(cnt.)

Computational Time (150 decimal digits) unit : sec.

Spectral Order	unknows	error $\|\cdot\|_\infty$	env. A	env. B	exflib env. C	Speed-up Ratio : $\frac{B}{C}$
40	**1681**	$\mathbf{10^{-47}}$	**10581**	**1503**	**428**	**3.51**
50	**2601**	$\mathbf{10^{-58}}$	**40285**	**5529**	**1624**	**3.40**
60	**3721**	$\mathbf{10^{-71}}$	—	**16169**	**4815** (1.0GB)	**3.36**
70	**5041**	$\mathbf{10^{-86}}$	—	**40296**	**11915**	**3.38**
80	**6561**	$\mathbf{10^{-106}}$	—	**89247**	**26900**	**3.32**
90	**8281**	$\mathbf{10^{-118}}$	—	**191380**	**54712** (4.6GB)	**3.50**

cf. T.Takeuchi, H.Imai, Y.Iso, "Infinite-Precision Numerical Simulation of Cauchy problems for the elliptic operator", 52th NCTAM Abstract(2003)

3. Application of the Proposed Multiple-Precision Arithmetic to One-Loop Integrals (Box Type)

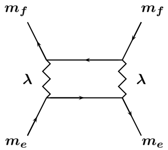
Thanks to Prof. F.Yuasa (KEK) and Prof. T.Ishikawa (KEK)

Infrared Divergent Process (One-Loop, Box Type)

$$I = \lim_{\epsilon \downarrow 0} \int_0^1 dx \int_0^{1-x} dy \int_0^{1-x-y} dz \frac{D^2 - \epsilon^2}{\left(D^2 + \epsilon^2\right)^2},$$

$$D = -xys - tz(1-x-y-z) + (x+y)\lambda^2 + (1-x-y-z)(1-x-y)m_e^2 + z(1-x-y)m_f^2.$$

λ	:	fictious photon mass $\sim 10^{-30}$
m_e	:	electron mass ~ 0.0005
m_f	:	Fermion mass ~ 150
s, t	:	Mandelstam variables
		$s = 500^2, t = -150^2$

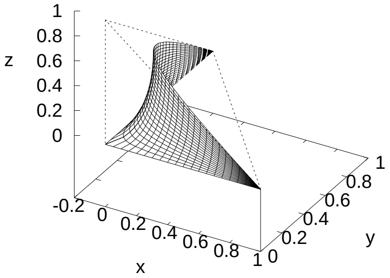


---

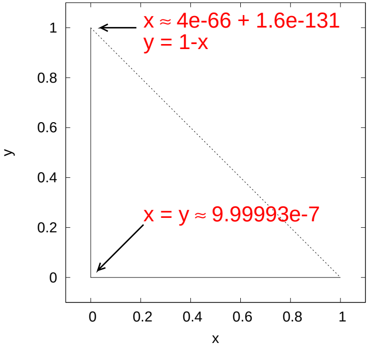
2 decimal digits in 5.1 days (2008 KEK, x, y-quadruple, z-double)

Surface of $D(x, y, z) = 0$

$$D = -xys - tz(1-x-y-z) + (x+y)\lambda^2 + (1-x-y-z)(1-x-y)m_e^2 + z(1-x-y)m_f^2.$$



(a) profile



(b) slice on $z = 0$

Difficluty 1 : $\epsilon \longrightarrow +0$

- $D(x, y, z)$ has zero in the integral domain.
  Exchange of $\lim_{\epsilon \longrightarrow +0}$ and $\int$ is not allowed.
- Difficulties in Extrapolation
  - choice of $\{\epsilon_n\}$
  - validation

Our Strategy

Direct computaion with $\epsilon \ll 1$

High-Accurate Quadrature Rule

$+$

Multiple-Precision Arithmetic by exflib

▸ ϵ-extrapolation ▸ DE rule

Difficulty 2 : Cancellation

$$I^{(0)}(x, y; \epsilon) = \int_0^{1-x-y} \frac{D^2 - \epsilon^2}{(D^2 + \epsilon^2)^2} dz.$$

$$x = 0.5, \epsilon = 10^{-60}$$

y	0.038360058419277516181026179141342735265889813952237803083329586181383 1
$I^{(0)}$	$+4.33 \times 10^{66}$
y	0.0383600584192775161810261791413427352658898139522378030833295861951030
$I^{(0)}$	-4.33×10^{66}
y	0.0383600584192775161810271943721675639816420468162279467522727339786472
$I^{(0)}$	-9.24×10^{87}
y	0.0383600584192775161810271943721675639816420468162279467522727339903181
$I^{(0)}$	$+2.77 \times 10^{87}$

$\Rightarrow$ Cancellations in adding $I^{(0)}(x, y_k; \epsilon)$.

Designing Algorithm

1. Set $\epsilon \ll 1$
2. Decompose the integrand to prevent Cancellation

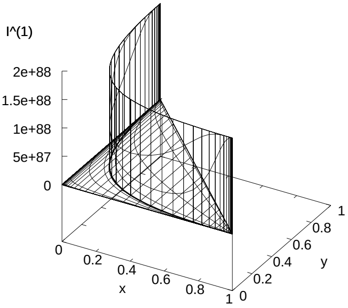
$$\frac{D^2 - \epsilon^2}{\left(D^2 + \epsilon^2\right)^2} = \frac{1}{D^2 + \epsilon^2} - \frac{2\epsilon^2}{\left(D^2 + \epsilon^2\right)^2}$$

3. $\cdots$

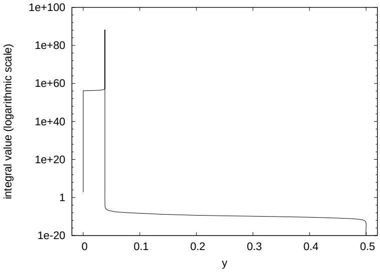
Difficluty 3 : Numerical Singularity

$$I^{(1)}(x, y; \epsilon) = \int_0^{1-x-y} \frac{dz}{D^2 + \epsilon^2}.$$

The integrand is regular (analytic) in the domain.

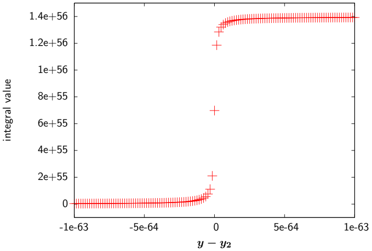


(a) Profile of $I^{(1)}, \epsilon = 10^{-60}$



(b) Silce at $x = 0.5$

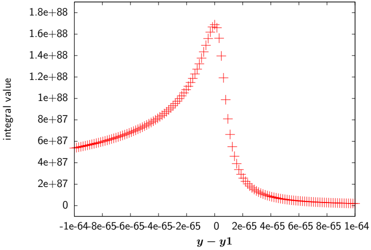
$$I^{(1)}(x, y; \epsilon) = \int_0^{1-x-y} \frac{dz}{D^2 + \epsilon^2}, \quad x = 0.5.$$



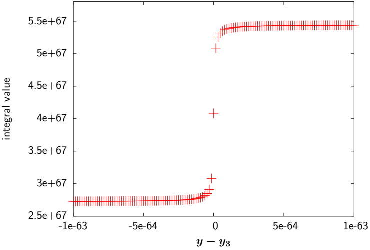
(c) In the interval $y_2 \pm 10^{-63}$

$$y_2 \approx 4.999999999990000000000024999999999930 \times 10^{-13}$$

Numerical Singularities



(d) In the interval $y_1 \pm 10^{-64}$



(e) In the interval $y_3 \pm 10^{-63}$

$$y_1 := \arg\max_{0<y<0.5} I^{(1)}(0.5, y; 10^{-60})$$

$$\approx 0.03836005841927751618102\textcolor{red}{71943721675639816}$$

$$y_3 \approx 0.03836005841927751618102\textcolor{red}{61791413427352659}$$

Proposed Algorithm (2010, F)

1. Set $\epsilon \ll 1$
2. Decompose the integrand to prevent Cancellation

$$\frac{D^2 - \epsilon^2}{\left(D^2 + \epsilon^2\right)^2} = \frac{1}{D^2 + \epsilon^2} - \frac{2\epsilon^2}{\left(D^2 + \epsilon^2\right)^2}$$

3. Find numerical singularities
4. Decompose the integral domain
5. Apply a numerical quadrature (DE rule)

Cancellation in the final subtraction

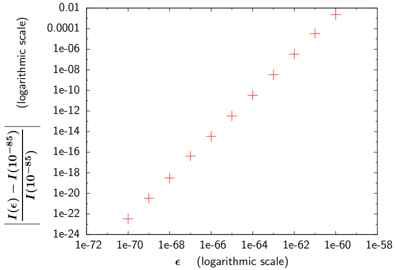
ϵ	$\iiint \frac{1}{D^2+\epsilon^2}$	$\iiint \frac{1}{D^2+\epsilon^2} - \iiint \frac{2\epsilon^2}{(D^2+\epsilon^2)^2}$
10^{-60}	$+2.00601 \times 10^{55}$	-3.5535×10^{-7}
10^{-65}	$+2.00601 \times 10^{60}$	-3.5617×10^{-7}
10^{-70}	$+2.00601 \times 10^{65}$	-3.5617×10^{-7}
10^{-75}	$+2.00601 \times 10^{70}$	-3.5617×10^{-7}
10^{-80}	$+2.00601 \times 10^{75}$	-3.5617×10^{-7}

Numerical Results (2010, F)

Infrared Divergent Process (One-Loop, Box Type)

512 Procs on HX600 (T2K @ Kyoto Univ, Opteron8350 2.3GHz)

ϵ	digits	$I(\epsilon)$	min.	N
10^{-40}	180	$-2.4724863525991758999 \times 10^{-7}$	52	32768
10^{-50}	180	$-3.0171115827587106392 \times 10^{-7}$	114	65536
10^{-60}	200	$-3.5535393863845744033 \times 10^{-7}$	116	65536
10^{-62}	200	$-3.5617356303387124202 \times 10^{-7}$	397	131072
10^{-62}	220	$-3.5617356303387125518 \times 10^{-7}$	135	65536
10^{-64}	220	$-3.5617368127999815129 \times 10^{-7}$	132	65536
10^{-66}	220	$-3.5617368129182335520 \times 10^{-7}$	131	65536
10^{-68}	220	$-3.5617368129182453682 \times 10^{-7}$	465	131072
10^{-68}	240	$-3.5617368129182453773 \times 10^{-7}$	543	131072
10^{-70}	240	$-3.5617368129182453784 \times 10^{-7}$	544	131072
10^{-80}	300	$-3.5617368129182453784347506602215657 \times 10^{-7}$		
10^{-85}	300	$-3.5617368129182453784347506602215657 \times 10^{-7}$		



(f) Relative Errrors

(g) Difference of $I(10\epsilon) - I(\epsilon)$

---

2 digits in 5.1 days (2008 KEK, x, y-quadruple, z-double, Opteron 2.2G

14 digits in 131 min. (2010 F, 220 digits, 512procs) $\approx$ 46.5 days

Numerical Results (2010, F)

Pair of infrared divergent process (Coresponding imaginary part)

$$J = \lim_{\epsilon \downarrow 0} \int_0^1 dx \int_0^{1-x} dy \int_0^{1-x-y} dz \frac{2\epsilon D}{\left(D^2 + \epsilon^2\right)^2}.$$

ϵ	$J(\epsilon)$, 300 digits	min.
10^{-15}	$3.715343935569068417 \times 10^{-9}$	
10^{-25}	$3.715368928426927073 \times 10^{-9}$	
10^{-35}	$3.715368928676695769 \times 10^{-9}$	85
10^{-45}	$3.715368928676700632 \times 10^{-9}$	97
10^{-55}	$3.715392581449913835 \times 10^{-9}$	94

Not Enough Precision ?

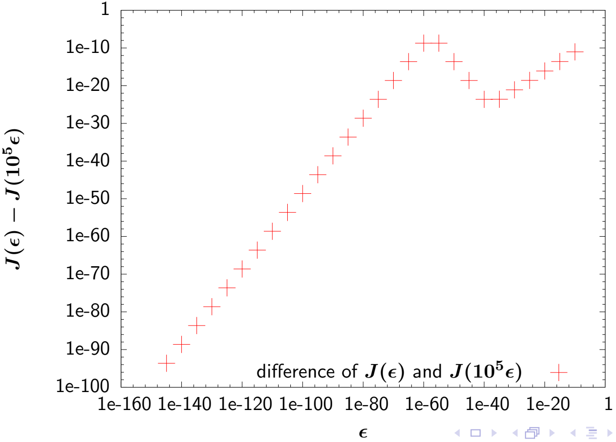
Pair of infrared divergent process (Coresponding imaginary part)

$$J = \lim_{\epsilon \downarrow 0} \int_0^1 dx \int_0^{1-x} dy \int_0^{1-x-y} dz \frac{2\epsilon D}{\left(D^2 + \epsilon^2\right)^2}.$$

ϵ	$J(\epsilon)$, 300 digits	min.
10^{-15}	$3.7153439355690684172 \times 10^{-9}$	
10^{-25}	$3.7153689284269270739 \times 10^{-9}$	
10^{-35}	$3.7153689286766957690 \times 10^{-9}$	85
10^{-45}	$3.7153689286767006320 \times 10^{-9}$	97
10^{-55}	$3.7153925814499138357 \times 10^{-9}$	94
10^{-60}	$5.5730533930150474001 \times 10^{-9}$	266
10^{-65}	$7.4307142045801809644 \times 10^{-9}$	244
10^{-75}	$7.4307378573533941682 \times 10^{-9}$	240
10^{-85}	$7.4307378573533965334 \times 10^{-9}$	262
10^{-100}	$7.43073785735339653344783661 \times 10^{-9}$	

Numerical Results (2010, F)

$$J(\epsilon) = \int_0^1 dx \int_0^{1-x} dy \int_0^{1-x-y} dz \frac{2\epsilon D}{\left(D^2+\epsilon^2\right)^2}.$$



Current Works

- Validating the proposed algorithm and results
- Optimizing the proposed algorithm
  Finding singularity, Numerical quadrature

Appendix

Parallel Efficiency

Parallel Computation by MPI

ϵ	digits	128proc	512proc	Parallel Efficiency
10^{-60}	200	446 min.	116 min.	0.96
10^{-65}	220	508	133	0.95
10^{-70}	240	2133	544	0.98
10^{-75}	300	3213	844	0.95

High parallel efficiency has been achieved.

Example of extrapolation : ϵ-algorithm

$$e_{0,n} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots + \frac{(-1)^n}{2n+1}.$$

n	$4e_{0,n}$
0	4.0000000000000000
2	3.4666666666666663
4	3.3396825396825394
6	3.2837384837384831
8	3.2523659347188754
10	3.2323158094055922
12	3.2184027659273315
14	3.2081856522619416
16	3.2003655154095467
18	3.1941879092319403

Example of extrapolation : ϵ-algorithm

$$e_{0,n} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots + \frac{(-1)^n}{2n+1}.$$

$$e_{i+1,j} = e_{i-1,j+1} + \frac{1}{e_{i,j+1} - e_{i,j}}, \quad e_{-1,n} = 0.$$

n	$4e_{0,n}$
0	4.0000000000000000
2	3.4666666666666663
4	3.3396825396825394
6	3.2837384837384831
8	3.2523659347188754
10	3.2323158094055922
12	3.2184027659273315
14	3.2081856522619416
16	3.2003655154095467
18	3.1941879092319403

Example of extrapolation : ϵ-algorithm

$$e_{0,n} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots + \frac{(-1)^n}{2n+1}.$$

$$e_{i+1,j} = e_{i-1,j+1} + \frac{1}{e_{i,j+1} - e_{i,j}}, \quad e_{-1,n} = 0.$$

n	$4e_{0,n}$	$4e_{2,n}$
0	4.0000000000000000	4.0000000000000000
2	3.4666666666666663	3.1666666666666665
4	3.3396825396825394	3.1423423423423418
6	3.2837384837384831	3.1416149068322978
8	3.2523659347188754	3.1415933118799271
10	3.2323158094055922	3.1415926730303343
12	3.2184027659273315	3.1415926541633659
14	3.2081856522619416	3.1415926536067058
16	3.2003655154095467	3.1415926535902914
18	3.1941879092319403	3.1415926535898073

In this case $\{e_{2,n}\}$ converges more rapidly than $\{e_{0,n}\}$.

Double Exponential Rule by H.Takahashi and M.Mori

$$I = \int_{-1}^{1} f(x)dx = \int_{-\infty}^{\infty} f\left(\phi(t)\right)\phi'(t)\,dt,$$

where

$$\phi(t) = \tanh\left(\frac{\pi}{2}\sinh t\right).$$

Trapesoidal rule to RHS:

$$I_h^{(N)} = h \sum_{k=-N_-}^{N_+} f\left(\phi(kh)\right)\phi'(kh)$$

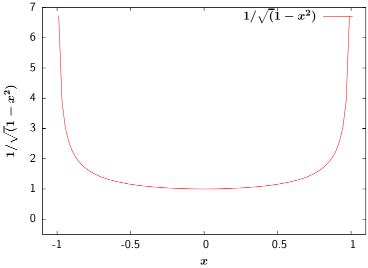
Error Estimate: under some assumptions,

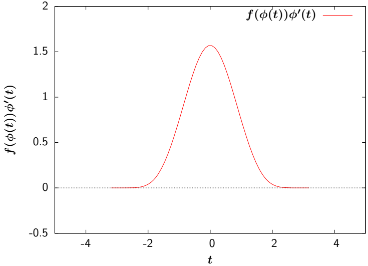
$$\left|I - I_h^{(N)}\right| = O\left(\exp(-cN/\log N)\right), \quad c > 0, N \to \infty.$$

DE transformation

Singularities at $x = \pm 1$

$$f(x) = \frac{1}{\sqrt{1 - x^2}}$$





$f(x)$ $\qquad$ $f(\phi(t))\phi'(t)$