Automatic Differentiation – Lecture No 2

Warwick Tucker

The CAPA group
Department of Mathematics
Uppsala University, Sweden

eScience Winter School, Geilo

UPPSALA
UNIVERSITET

Are floating point computations reliable?

Computing with the `C/C++` `single` format

UPPSALA
UNIVERSITET

Are floating point computations reliable?

Computing with the `C/C++` `single` format

Example 1: Repeated addition

$$\sum_{i=1}^{10^3} \langle 10^{-3} \rangle = 0.999990701675415,$$

$$\sum_{i=1}^{10^4} \langle 10^{-4} \rangle = 1.000053524971008.$$

Are floating point computations reliable?

Computing with the `C/C++` `single` format

Example 1: Repeated addition

$$\sum_{i=1}^{10^3} \langle 10^{-3} \rangle = 0.999990701675415,$$

$$\sum_{i=1}^{10^4} \langle 10^{-4} \rangle = 1.000053524971008.$$

Example 2: Order of summation

$$1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{10^6} = 14.357357,$$

$$\frac{1}{10^6} + \cdots + \frac{1}{3} + \frac{1}{2} + 1 = 14.392651.$$

Are floating point computations reliable?

Given the point $(x, y) = (77617, 33096)$, evaluate the function

$$f(x, y) = 333.75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5.5y^8 + x/(2y)$$

Are floating point computations reliable?

Given the point $(x, y) = (77617, 33096)$, evaluate the function

$$f(x, y) = 333.75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5.5y^8 + x/(2y)$$

IBM S/370 ($\beta = 16$) with FORTRAN:

type	p	$f(x, y)$
REAL*4	24	$1.172603\ldots$
REAL*8	53	$1.1726039400531\ldots$
REAL*10	64	$1.172603940053178\ldots$

Are floating point computations reliable?

Given the point $(x, y) = (77617, 33096)$, evaluate the function

$$f(x,y) = 333.75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5.5y^8 + x/(2y)$$

IBM S/370 ($\beta = 16$) with FORTRAN:

type	p	$f(x,y)$
REAL*4	24	$1.172603\ldots$
REAL*8	53	$1.1726039400531\ldots$
REAL*10	64	$1.172603940053178\ldots$

Pentium III ($\beta = 2$) with C/C++ (gcc/g++):

type	p	$f(x,y)$
float	24	178702833214061281280
double	53	178702833214061281280
long double	64	178702833214061281280

UPPSALA UNIVERSITET

Are floating point computations reliable?

Given the point $(x, y) = (77617, 33096)$, evaluate the function

$$f(x, y) = 333.75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5.5y^8 + x/(2y)$$

IBM S/370 ($\beta = 16$) with FORTRAN:

type	p	$f(x, y)$
`REAL*4`	24	$1.172603\ldots$
`REAL*8`	53	$1.1726039400531\ldots$
`REAL*10`	64	$1.172603940053178\ldots$

Pentium III ($\beta = 2$) with C/C++ (gcc/g++):

type	p	$f(x, y)$
`float`	24	178702833214061281280
`double`	53	178702833214061281280
`long double`	64	178702833214061281280

Correct answer: $-0.8273960599\ldots$

Are integer computations reliable?

Example 3: The harmonic series

If we define

$$S_N = \sum_{k=1}^{N} \frac{1}{k},$$

then $\lim_{N\to\infty} S_N = +\infty$. The computer also gets this result, but for the entirely wrong reason (integer wrapping).

$$I_{max} + 1 = -I_{max} = I_{min}$$

Are integer computations reliable?

Example 3: The harmonic series

If we define

$$S_N = \sum_{k=1}^{N} \frac{1}{k},$$

then $\lim_{N\to\infty} S_N = +\infty$. The computer also gets this result, but for the entirely wrong reason (integer wrapping).

$$I_{max} + 1 = -I_{max} = I_{min}$$

Example 4: Elementary Taylor series

Now define

$$S_N = \sum_{k=0}^{N} \frac{1}{k!},$$

then $\lim_{N\to\infty} S_N = e \approx 2.7182818$. The integer wrapping produces a sequence that is *not* strictly increasing:

Are integer computations reliable?

```
S_0  = 1.000000000000000          S_18 = 2.718281826004540  S
S_1  = 2.000000000000000          S_19 = 2.718281835125155
S_2  = 2.500000000000000          S_20 = 2.718281834649448  S
S_3  = 2.666666666666667          S_21 = 2.718281833812708  S
S_4  = 2.708333333333333          S_22 = 2.718281831899620  S
S_5  = 2.716666666666666          S_23 = 2.718281833059102
S_6  = 2.718055555555555          S_24 = 2.718281831770353  S
S_7  = 2.718253968253968          S_25 = 2.718281832252007
S_8  = 2.718278769841270          S_26 = 2.718281831712599  S
S_9  = 2.718281525573192          S_27 = 2.718281832386097
S_10 = 2.718281801146385          S_28 = 2.718281831659211  S
S_11 = 2.718281826198493          S_29 = 2.718281830853743  S
S_12 = 2.718281828286169          S_30 = 2.718281831563322
S_13 = 2.718281828803753          S_31 = 2.718281832917973
S_14 = 2.718281829585647          S_32 = 2.718281832452312  S
S_15 = 2.718281830084572          S_33 = 2.718281831986650  S
S_16 = 2.718281830583527          S_34 =        inf
S_17 = 2.718281827117590  S
```

UPPSALA
UNIVERSITET

How do we control rounding errors?

Round each partial result both ways

If $x, y \in \mathbb{F}$ and $\star \in \{+, -, \times, \div\}$, we can enclose the exact result in an *interval*:

$$x \star y \in [\bigtriangledown(x \star y), \bigtriangleup(x \star y)].$$

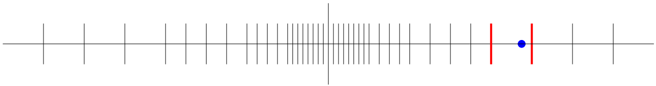
How do we control rounding errors?

Round each partial result both ways

If $x, y \in \mathbb{F}$ and $\star \in \{+, -, \times, \div\}$, we can enclose the exact result in an *interval*:

$$x \star y \in [\bigtriangledown(x \star y), \bigtriangleup(x \star y)].$$

Since all (modern) computers round with *maximal quality*, the interval is the smallest one that contains the exact result.



UPPSALA UNIVERSITET

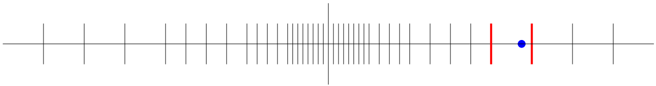
How do we control rounding errors?

Round each partial result both ways

If $x, y \in \mathbb{F}$ and $\star \in \{+, -, \times, \div\}$, we can enclose the exact result in an *interval*:

$$x \star y \in [\bigtriangledown(x \star y), \bigtriangleup(x \star y)].$$

Since all (modern) computers round with *maximal quality*, the interval is the smallest one that contains the exact result.



Question

How do we compute with intervals? And why, really?

Arithmetic over $\mathbb{IR}$

Definition

If $\star$ is one of the operators $+, -, \times, \div$, and if $\mathbb{A}, \mathbb{B} \in \mathbb{IR}$, then

$$\mathbb{A} \star \mathbb{B} = \{a \star b \colon a \in \mathbb{A}, b \in \mathbb{B}\},$$

except that $\mathbb{A} \div \mathbb{B}$ is undefined if $0 \in \mathbb{B}$.

Arithmetic over $\mathbb{IR}$

Definition

If $\star$ is one of the operators $+, -, \times, \div$, and if $\mathbb{A}, \mathbb{B} \in \mathbb{IR}$, then

$$\mathbb{A} \star \mathbb{B} = \{a \star b \colon a \in \mathbb{A}, b \in \mathbb{B}\},$$

except that $\mathbb{A} \div \mathbb{B}$ is undefined if $0 \in \mathbb{B}$.

Simple arithmetic

$$\begin{aligned}
\mathbb{A} + \mathbb{B} &= [\underline{\mathbb{A}} + \underline{\mathbb{B}}, \overline{\mathbb{A}} + \overline{\mathbb{B}}] \\
\mathbb{A} - \mathbb{B} &= [\underline{\mathbb{A}} - \overline{\mathbb{B}}, \overline{\mathbb{A}} - \underline{\mathbb{B}}] \\
\mathbb{A} \times \mathbb{B} &= [\min\{\underline{\mathbb{A}}\underline{\mathbb{B}}, \underline{\mathbb{A}}\overline{\mathbb{B}}, \overline{\mathbb{A}}\underline{\mathbb{B}}, \overline{\mathbb{A}}\overline{\mathbb{B}}\}, \max\{\underline{\mathbb{A}}\underline{\mathbb{B}}, \underline{\mathbb{A}}\overline{\mathbb{B}}, \overline{\mathbb{A}}\underline{\mathbb{B}}, \overline{\mathbb{A}}\overline{\mathbb{B}}\}] \\
\mathbb{A} \div \mathbb{B} &= \mathbb{A} \times [1/\overline{\mathbb{B}}, 1/\underline{\mathbb{B}}], \quad \text{if } 0 \notin \mathbb{B}.
\end{aligned}$$

Arithmetic over $\mathbb{IR}$

Definition

If $\star$ is one of the operators $+, -, \times, \div$, and if $\mathbb{A}, \mathbb{B} \in \mathbb{IR}$, then

$$\mathbb{A} \star \mathbb{B} = \{a \star b \colon a \in \mathbb{A}, b \in \mathbb{B}\},$$

except that $\mathbb{A} \div \mathbb{B}$ is undefined if $0 \in \mathbb{B}$.

Simple arithmetic

$$\begin{aligned}
\mathbb{A} + \mathbb{B} &= [\underline{\mathbb{A}} + \underline{\mathbb{B}}, \overline{\mathbb{A}} + \overline{\mathbb{B}}] \\
\mathbb{A} - \mathbb{B} &= [\underline{\mathbb{A}} - \overline{\mathbb{B}}, \overline{\mathbb{A}} - \underline{\mathbb{B}}] \\
\mathbb{A} \times \mathbb{B} &= [\min\{\underline{\mathbb{A}\mathbb{B}}, \underline{\mathbb{A}}\overline{\mathbb{B}}, \overline{\mathbb{A}}\underline{\mathbb{B}}, \overline{\mathbb{A}\mathbb{B}}\}, \max\{\underline{\mathbb{A}\mathbb{B}}, \underline{\mathbb{A}}\overline{\mathbb{B}}, \overline{\mathbb{A}}\underline{\mathbb{B}}, \overline{\mathbb{A}\mathbb{B}}\}] \\
\mathbb{A} \div \mathbb{B} &= \mathbb{A} \times [1/\overline{\mathbb{B}}, 1/\underline{\mathbb{B}}], \quad \text{if } 0 \notin \mathbb{B}.
\end{aligned}$$

On a computer we use *directed rounding*, e.g.

$$\mathbb{A} + \mathbb{B} = [\bigtriangledown(\underline{\mathbb{A}} \oplus \underline{\mathbb{B}}), \bigtriangleup(\overline{\mathbb{A}} \oplus \overline{\mathbb{B}})].$$

UPPSALA UNIVERSITET

Interval arithmetic - implementations

It is natural to implement the `interval` class as of two numbers – the endpoints of the interval:

Interval arithmetic - implementations

It is natural to implement the `interval` class as of two numbers – the endpoints of the interval:

```
function iv = interval(lo, hi)
% A naive interval class constructor.
if nargin == 1
    hi = lo;
elseif ( hi < lo )
    error('The endpoints do not define an interval.');
end
iv.lo = lo; iv.hi = hi;
iv = class(iv,'interval');
```

Interval arithmetic - implementations

It is natural to implement the `interval` class as of two numbers – the endpoints of the interval:

```
01 function iv = interval(lo, hi)
02 % A naive interval class constructor.
03 if nargin == 1
04     hi = lo;
05 elseif ( hi < lo )
06     error('The endpoints do not define an interval.');
07 end
08 iv.lo = lo; iv.hi = hi;
09 iv = class(iv,'interval');
```

By including lines 03 and 04, we allow the constuctor to automatically cast a single number x to a thin interval.

We must also inform `MATLAB` how to display `interval` objects. This is achieved via the m-file `display.m`:

Interval arithmetic - implementations

We must also inform MATLAB how to display interval objects. This is achieved via the m-file display.m:

```
function display(iv)
% A simple output formatter for the interval class.
disp([inputname(1), ' = ']);
fprintf('  [%17.17f, %17.17f]\n', iv.lo, iv.hi);
```

Interval arithmetic - implementations

We must also inform MATLAB how to display interval objects. This is achieved via the m-file display.m:

```
function display(iv)
% A simple output formatter for the interval class.
disp([inputname(1), ' = ']);
fprintf('  [%17.17f, %17.17f]\n', iv.lo, iv.hi);
```

We can now input/output intervals within the MATLAB environment:

```
>> a = interval(3, 4), b = interval(2, 5), c = interval(1)
a =
  [3.00000000000000000, 4.00000000000000000]
b =
  [2.00000000000000000, 5.00000000000000000]
c =
  [1.00000000000000000, 1.00000000000000000]
```

Interval arithmetic - implementations

Implementing the arithmetic operations is straight-forward:

Interval arithmetic - implementations

Implementing the arithmetic operations is straight-forward:

```
function result = plus(a, b)
% Overloading the '+' operator for intervals.
[a, b] = cast(a, b);
setround(-inf);
lo = a.lo + b.lo;
setround(+inf);
hi = a.hi + b.hi;
setround(0.5);
result = interval(lo, hi);
```

UPPSALA UNIVERSITET

Interval arithmetic - implementations

Implementing the arithmetic operations is straight-forward:

```
01 function result = plus(a, b)
02 % Overloading the '+' operator for intervals.
03 [a, b] = cast(a, b);
04 setround(-inf);
05 lo = a.lo + b.lo;
06 setround(+inf);
07 hi = a.hi + b.hi;
08 setround(0.5);
09 result = interval(lo, hi);
```

Note the call to setround on lines 04, 06, and 08:

```
function setround(rnd)
% A switch for changing rounding mode. The arguments
% {+inf, -inf, 0.5, 0} correspond to the roundings
% {upward, downward, to nearest, to zero}, respectively.
system_dependent('setround',rnd);
```

Interval arithmetic - implementations

Implementing the arithmetic operations is straight-forward:

```
01 function result = plus(a, b)
02 % Overloading the '+' operator for intervals.
03 [a, b] = cast(a, b);
04 setround(-inf);
05 lo = a.lo + b.lo;
06 setround(+inf);
07 hi = a.hi + b.hi;
08 setround(0.5);
09 result = interval(lo, hi);
```

Note the call to `setround` on lines 04, 06, and 08:

```
function setround(rnd)
% A switch for changing rounding mode. The arguments
% {+inf, -inf, 0.5, 0} correspond to the roundings
% {upward, downward, to nearest, to zero}, respectively.
system_dependent('setround',rnd);
```

This is a hidden (undocumented) feature of `MATLAB`.

Interval arithmetic - implementations

Performing some simple interval calculations, we have:

```
>> a+b, a-b, a*b, a/b
ans =
  [5.0000000000000000, 9.0000000000000000]
ans =
  [-2.0000000000000000, 2.0000000000000000]
ans =
  [6.0000000000000000, 20.0000000000000000]
ans =
  [0.5999999999999998, 2.0000000000000000]
```

Interval arithmetic - implementations

Performing some simple interval calculations, we have:

```
>> a+b, a-b, a*b, a/b
ans =
  [5.0000000000000000, 9.0000000000000000]
ans =
  [-2.0000000000000000, 2.0000000000000000]
ans =
  [6.0000000000000000, 20.0000000000000000]
ans =
  [0.59999999999999998, 2.0000000000000000]
```

Note the outward rounding in the left endpoint of the last result.

Interval arithmetic - implementations

Performing some simple interval calculations, we have:

```
>> a+b, a-b, a*b, a/b
ans =
  [5.0000000000000000, 9.0000000000000000]
ans =
  [-2.0000000000000000, 2.0000000000000000]
ans =
  [6.0000000000000000, 20.0000000000000000]
ans =
  [0.5999999999999998, 2.0000000000000000]
```

Note the outward rounding in the left endpoint of the last result.
Now let's try to find the smallest interval containing $1/10$.

```
>> interval(1/10)
ans =
  [0.1000000000000001, 0.1000000000000001]
>> interval(1)/10
ans =
  [0.0999999999999999, 0.1000000000000001]
```

UPPSALA UNIVERSITET

Interval-valued functions

Range enclosure

Extend a real-valued function f to an interval-valued F:

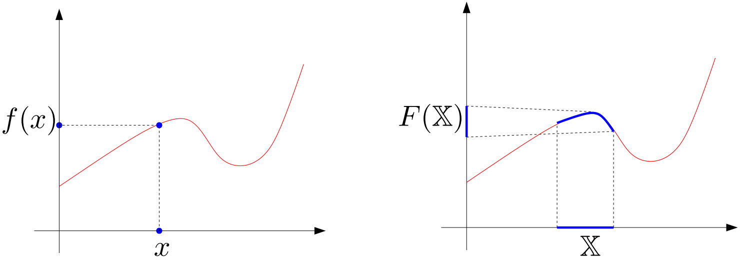
$$R(f;\mathbb{X}) = \{f(x) \colon x \in \mathbb{X}\} \subseteq F(\mathbb{X})$$

Interval-valued functions

Range enclosure

Extend a real-valued function f to an interval-valued F:

$$R(f;\mathbb{X}) = \{f(x) \colon x \in \mathbb{X}\} \subseteq F(\mathbb{X})$$

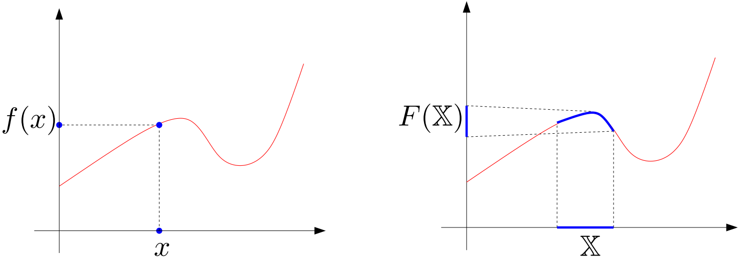


Interval-valued functions

Range enclosure

Extend a real-valued function f to an interval-valued F:

$$R(f;\mathbb{X}) = \{f(x) \colon x \in \mathbb{X}\} \subseteq F(\mathbb{X})$$



$y \notin F(\mathbb{X})$ implies that $f(x) \neq y$ for all $x \in \mathbb{X}$.

Interval-valued functions

Some explicit formulas are given below:

Interval-valued functions

Some explicit formulas are given below:

$$
\begin{array}{lll}
e^{\mathbb{X}} & = & [e^{\underline{\mathbb{X}}}, e^{\overline{\mathbb{X}}}] \\
\sqrt{\mathbb{X}} & = & [\sqrt{\underline{\mathbb{X}}}, \sqrt{\overline{\mathbb{X}}}] \qquad \text{if } 0 \leq \underline{\mathbb{X}} \\
\log \mathbb{X} & = & [\log \underline{\mathbb{X}}, \log \overline{\mathbb{X}}] \qquad \text{if } 0 < \underline{\mathbb{X}} \\
\arctan \mathbb{X} & = & [\arctan \underline{\mathbb{X}}, \arctan \overline{\mathbb{X}}] \quad .
\end{array}
$$

UPPSALA UNIVERSITET

Some explicit formulas are given below:

$$\begin{array}{lcll} e^{\mathbb{X}} & = & [e^{\underline{\mathbb{X}}}, e^{\overline{\mathbb{X}}}] & \\ \sqrt{\mathbb{X}} & = & [\sqrt{\underline{\mathbb{X}}}, \sqrt{\overline{\mathbb{X}}}] & \text{if } 0 \leq \underline{\mathbb{X}} \\ \log \mathbb{X} & = & [\log \underline{\mathbb{X}}, \log \overline{\mathbb{X}}] & \text{if } 0 < \underline{\mathbb{X}} \\ \arctan \mathbb{X} & = & [\arctan \underline{\mathbb{X}}, \arctan \overline{\mathbb{X}}] & . \end{array}$$

Set $S^+ = \{2k\pi + \pi/2 \colon k \in \mathbb{Z}\}$ and $S^- = \{2k\pi - \pi/2 \colon k \in \mathbb{Z}\}$.
Then $\sin \mathbb{X}$ is given by

$$\begin{cases} [-1, 1] & : \text{ if } \mathbb{X} \cap S^- \neq \emptyset \text{ and } \mathbb{X} \cap S^+ \neq \emptyset, \\ [-1, \max\{\sin \underline{\mathbb{X}}, \sin \overline{\mathbb{X}}\}] & : \text{ if } \mathbb{X} \cap S^- \neq \emptyset \text{ and } \mathbb{X} \cap S^+ = \emptyset, \\ [\min\{\sin \underline{\mathbb{X}}, \sin \overline{\mathbb{X}}\}, 1] & : \text{ if } \mathbb{X} \cap S^- = \emptyset \text{ and } \mathbb{X} \cap S^+ \neq \emptyset, \\ [\min\{\sin \underline{\mathbb{X}}, \sin \overline{\mathbb{X}}\}, \max\{\sin \underline{\mathbb{X}}, \sin \overline{\mathbb{X}}\}] & : \text{ if } \mathbb{X} \cap S^- = \emptyset \text{ and } \mathbb{X} \cap S^+ = \emptyset. \end{cases}$$

Interval-valued functions

A simple (and incorrect) implementation of $\sin$ is the following:

```
function result = sin(x)
% Overloading the 'sin' operator.
Sp = (2*floor(x.hi/(2*pi) - 1/4)*pi + pi/2 <= x);
Sm = (2*floor(x.hi/(2*pi) + 1/4)*pi - pi/2 <= x);
system_dependent('setround',-inf);
min_sin = min(sin(x.lo),sin(x.hi));
system_dependent('setround',+inf);
max_sin = max(sin(x.lo),sin(x.hi));
system_dependent('setround',0.5);
if ( Sm && Sp )
    result = interval(-1,+1);
elseif Sm
    result = interval(-1, max_sin);
elseif Sp
    result = interval(min_sin, +1);
else
    result = interval(min_sin, max_sin);
end
```

Exercise

Draw an accurate graph of the function $f(x) = \cos^3 x + \sin x$ over the domain $[-5, 5]$.

Graph Enclosures

Exercise

Draw an accurate graph of the function $f(x) = \cos^3 x + \sin x$ over the domain $[-5, 5]$.

Solution

Define $F(\mathbb{X}) = \cos^3 \mathbb{X} + \sin \mathbb{X}$, and adaptively bisect the domain $\mathbb{X}_0 = [-5, 5]$ into smaller pieces $\mathbb{X}_0 = \cup_{i=1}^{N} \mathbb{X}_i$ until we arrive at some desired accuracy, e.g. $\max_i \text{width}\big(F(\mathbb{X}_i)\big) \leq \texttt{TOL}$.

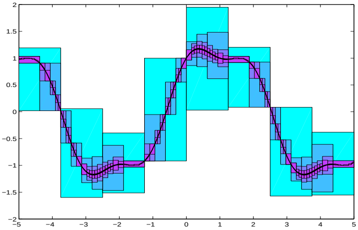
Graph Enclosures

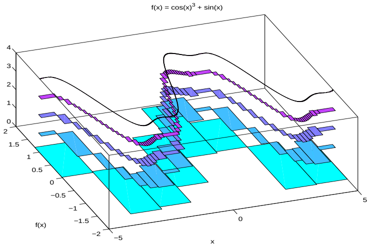
Exercise

Draw an accurate graph of the function $f(x) = \cos^3 x + \sin x$ over the domain $[-5, 5]$.

Solution

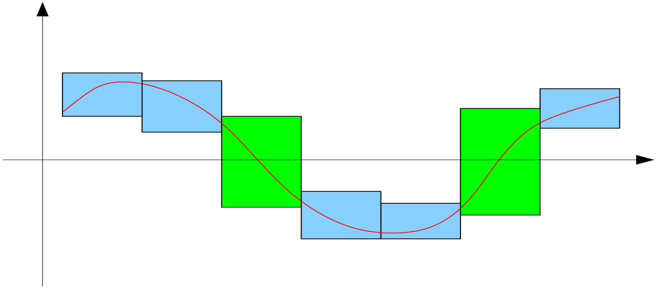
Define $F(\mathbb{X}) = \cos^3 \mathbb{X} + \sin \mathbb{X}$, and adaptively bisect the domain $\mathbb{X}_0 = [-5, 5]$ into smaller pieces $\mathbb{X}_0 = \cup_{i=1}^{N} \mathbb{X}_i$ until we arrive at some desired accuracy, e.g. $\max_i \text{width}\big(F(\mathbb{X}_i)\big) \leq \texttt{TOL}$.



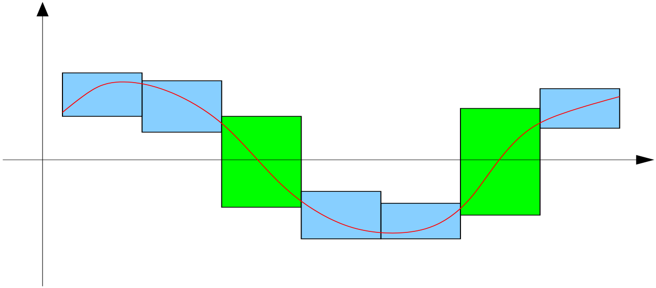


UPPSALA UNIVERSITET

Solving non-linear equations



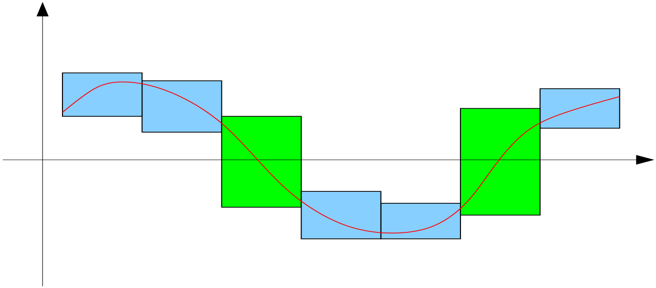
Solving non-linear equations



Consider everything. Keep what is good.
Avoid evil whenever you recognize it.
St. Paul, ca. 50 A.D. (The Bible, 1 Thess. 5:21-22)

UPPSALA
UNIVERSITET

Solving non-linear equations



Consider everything. Keep what is good.
Avoid evil whenever you recognize it.
St. Paul, ca. 50 A.D. (The Bible, 1 Thess. 5:21-22)

No solutions can be missed!

UPPSALA UNIVERSITET

Solving non-linear equation

The code is transparent and natural

```
function bisect(fcnName, X, tol)
f = inline(fcnName);
if ( 0 <= f(X) )         % If f(X) contains zero...
    if Diam(X) < tol     % and the tolerance is met...
        X                % print the interval X.
    else                 % Otherwise, divide and conquer.
        bisect(fcnName, interval(Inf(X), Mid(X)), tol);
        bisect(fcnName, interval(Mid(X), Sup(X)), tol);
    end
end
```

Solving non-linear equation

The code is transparent and natural

```
function bisect(fcnName, X, tol)
f = inline(fcnName);
if ( 0 <= f(X) )         % If f(X) contains zero...
    if Diam(X) < tol     % and the tolerance is met...
        X                % print the interval X.
    else                 % Otherwise, divide and conquer.
        bisect(fcnName, interval(Inf(X), Mid(X)), tol);
        bisect(fcnName, interval(Mid(X), Sup(X)), tol);
    end
end
```

Nice property

If F is well-defined on the domain, the algorithm produces an enclosure of *all* zeros of f. [No existence is established, however.]

Existence and uniqueness

Existence and uniqueness require *fixed point* theorems.

Existence and uniqueness

Existence and uniqueness require *fixed point* theorems.

Brouwer's fixed point theorem

Let B be homeomorhpic to the closed unit ball in $\mathbb{R}^n$. Then given any continuous mapping $f: B \to B$ there exists $x \in B$ such that $f(x) = x$.

Existence and uniqueness

Existence and uniqueness require *fixed point* theorems.

Brouwer's fixed point theorem

Let B be homeomorhpic to the closed unit ball in $\mathbb{R}^n$. Then given any continuous mapping $f\colon B \to B$ there exists $x \in B$ such that $f(x) = x$.

Schauder's fixed point theorem

Let X be a normed vector space, and let $K \subset X$ be a non-empty, compact, and convex set. Then given any continuous mapping $f\colon K \to K$ there exists $x \in K$ such that $f(x) = x$.

Existence and uniqueness

Existence and uniqueness require *fixed point* theorems.

Brouwer's fixed point theorem

Let B be homeomorhpic to the closed unit ball in $\mathbb{R}^n$. Then given any continuous mapping $f\colon B \to B$ there exists $x \in B$ such that $f(x) = x$.

Schauder's fixed point theorem

Let X be a normed vector space, and let $K \subset X$ be a non-empty, compact, and convex set. Then given any continuous mapping $f\colon K \to K$ there exists $x \in K$ such that $f(x) = x$.

Banach's fixed point theorem

If f is a contraction defined on a complete metric space X, then there exists a unique $x \in X$ such that $f(x) = x$.

Newton's method in $\mathbb{IR}$

Theorem

Let $f \in C^1(\mathbb{R}, \mathbb{R})$, *and set* $\check{x} = \mathrm{mid}(\mathbb{X})$. *We define*

$$N_f(\mathbb{X}) \stackrel{\text{def}}{=} N_f(\mathbb{X}, \check{x}) = \check{x} - f(\check{x})/F'(\mathbb{X}).$$

If $N_f(\mathbb{X})$ *is well-defined, then the following statements hold:*

Newton's method in $\mathbb{IR}$

Theorem

Let $f \in C^1(\mathbb{R}, \mathbb{R})$*, and set* $\check{x} = \mathrm{mid}(\mathbb{X})$*. We define*

$$N_f(\mathbb{X}) \stackrel{def}{=} N_f(\mathbb{X}, \check{x}) = \check{x} - f(\check{x})/F'(\mathbb{X}).$$

If $N_f(\mathbb{X})$ *is well-defined, then the following statements hold:*

(1) *if* $\mathbb{X}$ *contains a zero* x^* *of* f*, then so does* $N_f(\mathbb{X}) \cap \mathbb{X}$*;*

Newton's method in $\mathbb{IR}$

Theorem

Let $f \in C^1(\mathbb{R}, \mathbb{R})$, *and set* $\check{x} = \mathrm{mid}(\mathbb{X})$. *We define*

$$N_f(\mathbb{X}) \stackrel{\text{def}}{=} N_f(\mathbb{X}, \check{x}) = \check{x} - f(\check{x})/F'(\mathbb{X}).$$

If $N_f(\mathbb{X})$ *is well-defined, then the following statements hold:*

(1) *if* $\mathbb{X}$ *contains a zero* x^* *of* f, *then so does* $N_f(\mathbb{X}) \cap \mathbb{X}$;

(2) *if* $N_f(\mathbb{X}) \cap \mathbb{X} = \emptyset$, *then* $\mathbb{X}$ *contains no zeros of* f;

Newton's method in $\mathbb{IR}$

Theorem

Let $f \in C^1(\mathbb{R}, \mathbb{R})$*, and set* $\check{x} = \text{mid}(\mathbb{X})$*. We define*

$$N_f(\mathbb{X}) \stackrel{def}{=} N_f(\mathbb{X}, \check{x}) = \check{x} - f(\check{x})/F'(\mathbb{X}).$$

If $N_f(\mathbb{X})$ *is well-defined, then the following statements hold:*

(1) *if* $\mathbb{X}$ *contains a zero* x^* *of* f*, then so does* $N_f(\mathbb{X}) \cap \mathbb{X}$*;*

(2) *if* $N_f(\mathbb{X}) \cap \mathbb{X} = \emptyset$*, then* $\mathbb{X}$ *contains no zeros of* f*;*

(3) *if* $N_f(\mathbb{X}) \subseteq \mathbb{X}$*, then* $\mathbb{X}$ *contains a unique zero of* f*.*

Newton's method in $\mathbb{IR}$

Theorem

Let $f \in C^1(\mathbb{R}, \mathbb{R})$*, and set* $\check{x} = \mathrm{mid}(\mathbb{X})$*. We define*

$$N_f(\mathbb{X}) \stackrel{def}{=} N_f(\mathbb{X}, \check{x}) = \check{x} - f(\check{x})/F'(\mathbb{X}).$$

If $N_f(\mathbb{X})$ *is well-defined, then the following statements hold:*

(1) *if* $\mathbb{X}$ *contains a zero* x^* *of* f*, then so does* $N_f(\mathbb{X}) \cap \mathbb{X}$*;*

(2) *if* $N_f(\mathbb{X}) \cap \mathbb{X} = \emptyset$*, then* $\mathbb{X}$ *contains no zeros of* f*;*

(3) *if* $N_f(\mathbb{X}) \subseteq \mathbb{X}$*, then* $\mathbb{X}$ *contains a unique zero of* f*.*

Proof.

(1) Follows from the MVT;

(2) The contra-positive statement of (1);

(3) Existence from Brouwer's fixed point theorem;
Uniqueness from non-vanishing f'.

□

Newton's method in $\mathbb{IR}$

Algorithm

Starting from an initial search region $\mathbb{X}_0$, we form the sequence

$$\mathbb{X}_{i+1} = N_f(\mathbb{X}_i) \cap \mathbb{X}_i \qquad i = 0, 1, \ldots .$$

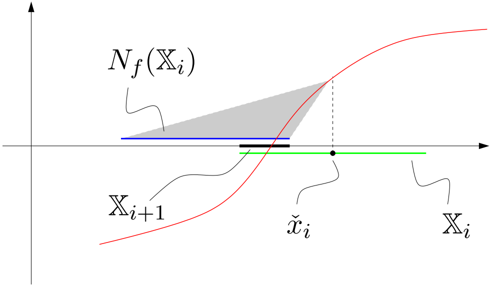
UPPSALA UNIVERSITET

Newton's method in $\mathbb{IR}$

Algorithm

Starting from an initial search region $\mathbb{X}_0$, we form the sequence

$$\mathbb{X}_{i+1} = N_f(\mathbb{X}_i) \cap \mathbb{X}_i \qquad i = 0, 1, \ldots.$$

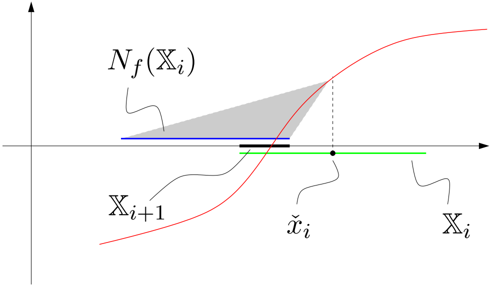


Newton's method in $\mathbb{R}$

Algorithm

Starting from an initial search region $\mathbb{X}_0$, we form the sequence

$$\mathbb{X}_{i+1} = N_f(\mathbb{X}_i) \cap \mathbb{X}_i \qquad i = 0, 1, \ldots .$$



Performance

If well-defined, this method is never worse than bisection, and it converges quadratically fast under mild conditions.

Newton's method in $\mathbb{R}$

Example

Let $f(x) = -2.001 + 3x - x^3$, and $\mathbb{X}_0 = [-3, -3/2]$. Then $F'(\mathbb{X}_0) = [-24, -15/4]$, so $N_f(\mathbb{X}_0)$ is well-defined, and the above theorem holds.

Newton's method in $\mathbb{IR}$

Example

Let $f(x) = -2.001 + 3x - x^3$, and $\mathbb{X}_0 = [-3, -3/2]$. Then $F'(\mathbb{X}_0) = [-24, -15/4]$, so $N_f(\mathbb{X}_0)$ is well-defined, and the above theorem holds.

```
X(0) = [-3.000000000000000,-1.500000000000000]; rad = 7.50000e-01
X(1) = [-2.140015625000001,-1.546099999999996]; rad = 2.96958e-01
X(2) = [-2.140015625000001,-1.961277398284108]; rad = 8.93691e-02
X(3) = [-2.006849239640351,-1.995570580247208]; rad = 5.63933e-03
X(4) = [-2.000120104486270,-2.000103608530276]; rad = 8.24798e-06
X(5) = [-2.000111102890393,-2.000111102873815]; rad = 8.28893e-12
X(6) = [-2.000111102881727,-2.000111102881724]; rad = 1.55431e-15
X(7) = [-2.000111102881727,-2.000111102881724]; rad = 1.55431e-15
Finite convergence!
Unique root in -2.00011110288172 +- 1.555e-15
```

UPPSALA UNIVERSITET

Newton's method in $\mathbb{R}$

Example

Let $f(x) = -2.001 + 3x - x^3$, and $\mathbb{X}_0 = [-3, -3/2]$. Then $F'(\mathbb{X}_0) = [-24, -15/4]$, so $N_f(\mathbb{X}_0)$ is well-defined, and the above theorem holds.

```
X(0) = [-3.000000000000000,-1.500000000000000]; rad = 7.50000e-01
X(1) = [-2.140015625000001,-1.546099999999996]; rad = 2.96958e-01
X(2) = [-2.140015625000001,-1.961277398284108]; rad = 8.93691e-02
X(3) = [-2.006849239640351,-1.995570580247208]; rad = 5.63933e-03
X(4) = [-2.000120104486270,-2.000103608530276]; rad = 8.24798e-06
X(5) = [-2.000111102890393,-2.000111102873815]; rad = 8.28893e-12
X(6) = [-2.000111102881727,-2.000111102881724]; rad = 1.55431e-15
X(7) = [-2.000111102881727,-2.000111102881724]; rad = 1.55431e-15
Finite convergence!
Unique root in -2.00011110288172 +- 1.555e-15
```

Question:

What do we do when $\mathbb{X}_0$ contains several zeros?

The Krawczyk method

If f has a zero x^* in $\mathbb{X}$, then for any $x \in \mathbb{X}$, we can enclose the zero via

$$x^* \in x - Cf(x) - \big(1 - CF'(\mathbb{X})\big)(x - \mathbb{X}) \stackrel{\text{def}}{=} K_f(\mathbb{X}, x, C).$$

The Krawczyk method

If f has a zero x^* in $\mathbb{X}$, then for any $x \in \mathbb{X}$, we can enclose the zero via

$$x^* \in x - Cf(x) - \big(1 - CF'(\mathbb{X})\big)(x - \mathbb{X}) \stackrel{\text{def}}{=} K_f(\mathbb{X}, x, C).$$

Good choices are $x = \check{x}$ and $C = 1/f'(\check{x})$, yielding the *Krawczyk operator*

$$K_f(\mathbb{X}) \stackrel{\text{def}}{=} \check{x} - \frac{f(\check{x})}{f'(\check{x})} - \left(1 - \frac{F'(\mathbb{X})}{f'(\check{x})}\right)[-r, r],$$

where we use the notation $r = \mathrm{rad}(\mathbb{X})$.

UPPSALA UNIVERSITET

The Krawczyk method

If f has a zero x^* in $\mathbb{X}$, then for any $x \in \mathbb{X}$, we can enclose the zero via

$$x^* \in x - Cf(x) - \big(1 - CF'(\mathbb{X})\big)(x - \mathbb{X}) \stackrel{\text{def}}{=} K_f(\mathbb{X}, x, C).$$

Good choices are $x = \check{x}$ and $C = 1/f'(\check{x})$, yielding the *Krawczyk operator*

$$K_f(\mathbb{X}) \stackrel{\text{def}}{=} \check{x} - \frac{f(\check{x})}{f'(\check{x})} - \left(1 - \frac{F'(\mathbb{X})}{f'(\check{x})}\right)[-r, r],$$

where we use the notation $r = \text{rad}(\mathbb{X})$.

Theorem

Assume that $K_f(\mathbb{X})$ is well-defined. Then the following statements hold:

(1) if $\mathbb{X}$ contains a zero x^* of f, then so does $K_f(\mathbb{X}) \cap \mathbb{X}$;

(2) if $K_f(\mathbb{X}) \cap \mathbb{X} = \emptyset$, then $\mathbb{X}$ contains no zeros of f;

(3) if $K_f(\mathbb{X}) \subseteq \text{int}(\mathbb{X})$, then $\mathbb{X}$ contains a unique zero of f.

The Krawczyk method with bisection

Example

Let $f(x) = \sin x(x - \cos x)$, and $\mathbb{X}_0 = [1, 15]$.

The Krawczyk method with bisection

Example

Let $f(x) = \sin x(x - \cos x)$, and $\mathbb{X}_0 = [1, 15]$.

```
Domain            : [1, 15]
Tolerance         : 1e-13
Function calls    : 59
 Unique zero in the interval 3.14159265358979[08,76]
 Unique zero in the interval 6.28318530717958[44,89]
 Unique zero in the interval 9.4247779607693[757,865]
 Unique zero in the interval 12.5663706143591[689,796]
```

Example

Let $f(x) = \sin x(x - \cos x)$, and $\mathbb{X}_0 = [1, 15]$.

```
Domain            : [1, 15]
Tolerance         : 1e-13
Function calls    : 59
 Unique zero in the interval 3.14159265358979[08,76]
 Unique zero in the interval 6.28318530717958[44,89]
 Unique zero in the interval 9.4247779607693[757,865]
 Unique zero in the interval 12.5663706143591[689,796]
```

Of course, all derivaties are computed using AD-techniques, overloaded with intervals.

Quadrature

Task:

Compute (approximate/enclose) the definite integral

$$I = \int_a^b f(x)dx.$$

Quadrature

Task:

Compute (approximate/enclose) the definite integral

$$I = \int_a^b f(x)dx.$$

We assume that I is well-defined, and that f is nice...

UPPSALA UNIVERSITET

Quadrature

Task:

Compute (approximate/enclose) the definite integral

$$I = \int_a^b f(x)dx.$$

We assume that I is well-defined, and that f is nice...

Naive approach:

Split the domain of integration into N equally wide subintervals: we set $h = (b-a)/N$ and $x_i = a + ih$, $i = 0, \dots, N$, and enclose the integrand via IA.

Quadrature

Task:

Compute (approximate/enclose) the definite integral

$$I = \int_a^b f(x)dx.$$

We assume that I is well-defined, and that f is nice...

Naive approach:

Split the domain of integration into N equally wide subintervals: we set $h = (b-a)/N$ and $x_i = a + ih$, $i = 0, \dots, N$, and enclose the integrand via IA.

This produces the enclosure

$$\int_a^b f(x)dx \in I_a^b(f, N) \stackrel{\text{def}}{=} h \sum_{i=1}^{N} F([x_{i-1}, x_i]),$$

which satisfies $w(I_a^b(f, N)) = \mathcal{O}(1/N)$.

Example

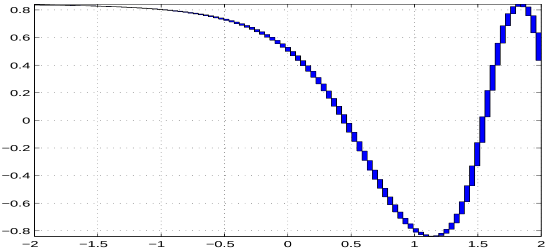
Enclose the definite integral $\int_{-2}^{2} \sin\left(\cos\left(e^x\right)\right) dx$.

Quadrature

Example

Enclose the definite integral $\int_{-2}^{2} \sin(\cos(e^x))dx$.

N	$I_{-2}^{2}(f,N)$	$w(I_{-2}^{2}(f,N))$
10^0	$[-3.36588, 3.36588]$	$6.73177 \cdot 10^0$
10^2	$[1.26250, 1.41323]$	$1.50729 \cdot 10^{-1}$
10^4	$[1.33791, 1.33942]$	$1.50756 \cdot 10^{-3}$
10^6	$[1.33866, 1.33868]$	$1.50758 \cdot 10^{-5}$



Quadrature

Taylor series approach:

Generate tighter bounds on the integrand via AD.

Quadrature

Taylor series approach:

Generate tighter bounds on the integrand via AD.

For all $x \in \mathbb{X}$, we have

$$\begin{aligned} f(x) &= \sum_{k=0}^{n-1} f_k(\check{x})(x-\check{x})^k + f_n(\zeta_x)(x-\check{x})^n \\ &\in \sum_{k=0}^{n} f_k(\check{x})(x-\check{x})^k + [-\varepsilon_n, \varepsilon_n]|x-\check{x}|^n, \end{aligned}$$

where $\varepsilon_n = \mathrm{mag}\big(F_n(\mathbb{X}) - f_n(\check{x})\big)$.

Quadrature

Taylor series approach:

Generate tighter bounds on the integrand via AD.

For all $x \in \mathbb{X}$, we have

$$\begin{aligned} f(x) &= \sum_{k=0}^{n-1} f_k(\check{x})(x-\check{x})^k + f_n(\zeta_x)(x-\check{x})^n \\ &\in \sum_{k=0}^{n} f_k(\check{x})(x-\check{x})^k + [-\varepsilon_n, \varepsilon_n]|x-\check{x}|^n, \end{aligned}$$

where $\varepsilon_n = \mathrm{mag}\big(F_n(\mathbb{X}) - f_n(\check{x})\big)$.
We are now prepared to compute the integral itself:

$$\begin{aligned} \int_{\check{x}-r}^{\check{x}+r} f(x)dx &\in \int_{\check{x}-r}^{\check{x}+r} \left(\sum_{k=0}^{n} f_k(\check{x})(x-\check{x})^k + [-\varepsilon_n, \varepsilon_n]|x-\check{x}|^n\right) dx \\ &= \sum_{k=0}^{n} f_k(\check{x}) \int_{-r}^{r} x^k dx + [-\varepsilon_n, \varepsilon_n] \int_{-r}^{r} |x|^n dx. \end{aligned}$$

Continuing the calculation, we see a lot of cancellation:

$$\begin{aligned}
\int_{\check{x}-r}^{\check{x}+r} f(x)dx &\in \sum_{k=0}^{n} f_k(\check{x}) \int_{-r}^{r} x^k dx + [-\varepsilon_n, \varepsilon_n] \int_{-r}^{r} |x|^n dx \\
&= \sum_{k=0}^{\lfloor n/2 \rfloor} f_{2k}(\check{x}) \int_{-r}^{r} x^{2k} dx + [-\varepsilon_n, \varepsilon_n] \int_{-r}^{r} |x|^n dx \\
&= 2 \left(\sum_{k=0}^{\lfloor n/2 \rfloor} f_{2k}(\check{x}) \frac{r^{2k+1}}{2k+1} + [-\varepsilon_n, \varepsilon_n] \frac{r^{n+1}}{n+1} \right).
\end{aligned}$$

Quadrature

Continuing the calculation, we see a lot of cancellation:

$$\begin{aligned}\int_{\check{x}-r}^{\check{x}+r} f(x)dx &\in \sum_{k=0}^{n} f_k(\check{x}) \int_{-r}^{r} x^k dx + [-\varepsilon_n, \varepsilon_n] \int_{-r}^{r} |x|^n dx \\ &= \sum_{k=0}^{\lfloor n/2 \rfloor} f_{2k}(\check{x}) \int_{-r}^{r} x^{2k} dx + [-\varepsilon_n, \varepsilon_n] \int_{-r}^{r} |x|^n dx \\ &= 2\left(\sum_{k=0}^{\lfloor n/2 \rfloor} f_{2k}(\check{x}) \frac{r^{2k+1}}{2k+1} + [-\varepsilon_n, \varepsilon_n] \frac{r^{n+1}}{n+1} \right).\end{aligned}$$

Partition the domain of integration: $a = x_0 < x_1 < \cdots < x_N = b$.

$$\begin{aligned}\int_a^b f(x)dx &= \sum_{i=1}^{N} \int_{x_{i-1}}^{x_i} f(x)dx = \sum_{i=1}^{N} \int_{\check{x}_i - r_i}^{\check{x}_i + r_i} f(x)dx \\ &\in 2\sum_{i=1}^{N} \left(\sum_{k=0}^{\lfloor n/2 \rfloor} f_{2k}(\check{x}_i) \frac{r_i^{2k+1}}{2k+1} + [-\varepsilon_{n,i}, \varepsilon_{n,i}] \frac{r_i^{n+1}}{n+1} \right).\end{aligned}$$

UPPSALA UNIVERSITET

Quadrature

Uniform partition:

N	$E^2_{-2}(f,6,N)$	$w(E^2_{-2}(f,6,N))$
9	$[0.86325178469, 1.81128961988]$	$9.4804 \cdot 10^{-1}$
12	$[1.28416304745, 1.39316025451]$	$1.0900 \cdot 10^{-1}$
21	$[1.33783795371, 1.33950680633]$	$1.6689 \cdot 10^{-3}$
75	$[1.33866863493, 1.33866878008]$	$1.4514 \cdot 10^{-7}$

Uniform partition:

N	$E^2_{-2}(f, 6, N)$	$w(E^2_{-2}(f, 6, N))$
9	$[0.86325178469, 1.81128961988]$	$9.4804 \cdot 10^{-1}$
12	$[1.28416304745, 1.39316025451]$	$1.0900 \cdot 10^{-1}$
21	$[1.33783795371, 1.33950680633]$	$1.6689 \cdot 10^{-3}$
75	$[1.33866863493, 1.33866878008]$	$1.4514 \cdot 10^{-7}$

Adaptive partition:

TOL	$A^2_{-2}(f, 6, \mathrm{TOL})$	$w(A^2_{-2}(f, 6, \mathrm{TOL}))$	N_{TOL}
10^{-1}	$[1.33229594606, 1.34500942603]$	$1.2713 \cdot 10^{-2}$	9
10^{-2}	$[1.33822575109, 1.33911045235]$	$8.8470 \cdot 10^{-4}$	12
10^{-4}	$[1.33866170207, 1.33867571626]$	$1.4014 \cdot 10^{-5}$	21
10^{-8}	$[1.33866870618, 1.33866870862]$	$2.4304 \cdot 10^{-9}$	75

Example (A bonus problem)

Compute the integral $\int_0^8 \sin\left(x + e^x\right) dx$.

Example (A bonus problem)

Compute the integral $\int_0^8 \sin\left(x + e^x\right) dx$.

A regular MATLAB session:

```
% Define the integrand and domain.
>> f = vectorize(inline('sin(x + exp(x))'));
>> a = 0; b = 8;
% Compute the integral using MATLAB's 'quad'.
>> q = quad(f,a,b)
q =
    0.251102722027180
```

Example (A bonus problem)

Compute the integral $\int_0^8 \sin\left(x + e^x\right) dx$.

A regular MATLAB session:

```
% Define the integrand and domain.
>> f = vectorize(inline('sin(x + exp(x))'));
>> a = 0; b = 8;
% Compute the integral using MATLAB's 'quad'.
>> q = quad(f,a,b)
q =
    0.251102722027180
```

Using the adaptive validated integrator:

```
$$ ./adQuad 0 8 20 1e-10
Partitions: 874
CPU time   : 0.45 seconds
Integral   : 0.3474001726[492276,652638]
```

Further study...

Interval Computations Web Page

http://www.cs.utep.edu/interval-comp

Further study...

Interval Computations Web Page

http://www.cs.utep.edu/interval-comp

INTLAB – INTerval LABoratory

http://www.ti3.tu-harburg.de/~rump/intlab/

Further study...

Interval Computations Web Page

http://www.cs.utep.edu/interval-comp

INTLAB – INTerval LABoratory

http://www.ti3.tu-harburg.de/~rump/intlab/

CXSC – C eXtensions for Scientific Computation

http://www.xsc.de/

Further study...

Interval Computations Web Page

http://www.cs.utep.edu/interval-comp

INTLAB – INTerval LABoratory

http://www.ti3.tu-harburg.de/~rump/intlab/

CXSC – C eXtensions for Scientific Computation

http://www.xsc.de/

PROFIL/BIAS

http://www.ti3.tu-harburg.de/Software/PROFILEnglisch.html