Motivations for an arbitrary precision interval arithmetic and the MPFI library

Nathalie Revol Project INRIA Arénaire, LIP, ENS-Lyon
and Lab. ANO, Univ. Lille, France
`Nathalie.Revol@ens-lyon.fr`

Fabrice Rouillier Project INRIA Spaces, Loria, Nancy
and LIP6, Paris, France
`Fabrice.Rouillier@loria.fr`

Validated Computing, Toronto, Ontario, Canada, 23-25 May 2002

Problems with roundings
theoretical limits

Accuracy and Stability of Numerical Algorithms, N. Higham, 1996

Theoretical study of the stability of the factorization of a dense matrix: stability ensured if

$$n^3 u < 1$$

where 1^+ is the smallest fp number (machine representable) > 1
and $u = 1^+ - 1$,
in other words OK if $n < 2 \times 10^5$ in IEEE double precision.
This size is almost reached in nowadays computations.

More generally: stability ensured if $nb\_op \times u < 1$
i.e. nb_op must be less than 10^{16}.

Studying the numerical quality of a code

- **“manual” study**

CEA (French Nuclear Agency)
Aérospatiale

- **controlling rounding errors**

Cadna: stochastic arithmetic
helps in finding where problems occur

- **correcting rounding errors**

CENA: computes rounding errors and corrects the results
first-order estimations + bounds on the rounding errors (OK for “linear algorithms”)

Tool to study and control the numerical quality

- **Interval arithmetic**

(Moore 1966, Kulisch 1983, Neumaier 1990, Rump 1994, Alefeld and Mayer 2000. . .)

principle: every number is replaced by a number containing it

ex: $\pi \in [3.14159; 3.14160]$, data $\in$ measured value $+[-\tau, \tau]$

- **Advantages**

$\oplus$ certified (or guaranteed) computations

$\oplus$ rounding errors estimation (?)

$\oplus$ global computations informations

- **Drawbacks**

$\ominus$ enclosures **too large**

Interval arithmetic: global information

Range of a (continuous) function over a whole set

enclosure obtained by enclosing this set in an interval I
and by evaluating an expression for this function on I.

Use in global optimization:

if $F(I) > \bar{f}$ then I cannot contain the global minimizer.

Brouwer theorem :

if $F(I) \subset I$ then f has a fixed point in I.
Under some assumptions, if the inclusion is proper then this fixed point is unique.

Outline

- **Bisection: need for more precision**
  - problem in interval arithmetic: overestimation
  - solution: bisection (theorem and limits)
  - arbitrary precision is needed
- **The MPFI library**
  - general view of MPFI
  - closer view of MPFI
- **Applications**
  - enclosures of polynomial real roots
  - approximations of real roots

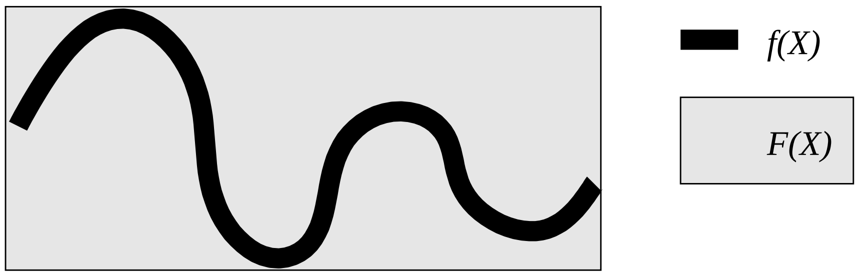
Interval arithmetic: sources of troubles

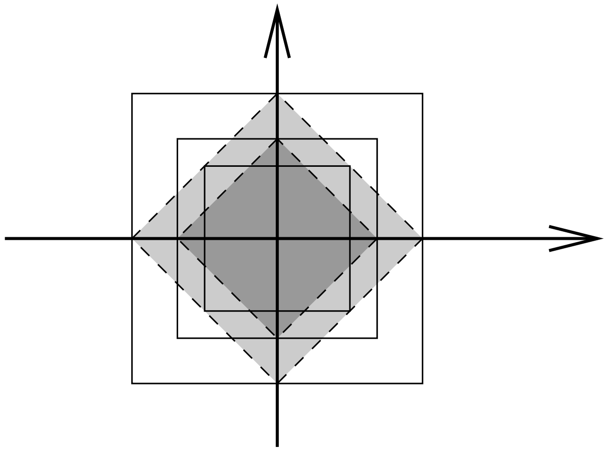
Variable dependency

$$I * I = \{x * y \,/\, x \in I,\ y \in I\} \text{ whereas } I^2 = \{x^2 \,/\, x \in I\} :$$

the correlation between the first x and the second one is lost.
Ex: $[-3, 4] \times [-3, 4] = [-12, 16]$ whereas $[-3, 4]^2 = [0, 16]$.

Wrapping effect





Solution = bisection

Bisect intervals:

if $X = X_1 \bigcup X_2$, $F(X_1) \bigcup F(X_2) \subset F(X)$

and the left interval is usually tighter than the right one.

Interval Newton algorithm:

apply one step of Newton to the current interval X

if the interval is not enough reduced then

  bisect X into X_1 and X_2, recursive application of Newton

Hansen's algorithm:

test if the current interval X can contain the optimizer

if yes, reduce X

if X is not reduced enough, then

  bisect X into X_1 and X_2, recursive application of Hansen

Theorem: bisection does work

• Under mild assumptions on f, if $f(X)$ is evaluated by replacing x by X in an expression of f, then this enclosure $F(x)$ satisfies:

$$\text{dist}(f(X), F(X)) \leq \mathcal{O}(w(X)).$$

• **With 1st order Taylor expansion,** quadratic approximation.

• **Moral**: in order to refine the precision on the results, the precision on the inputs must be refined. . . and thus it should be possible to refine as much as required.

Limits of bisection

• Under mild assumptions on f, if $f(X)$ is evaluated by replacing x by X in an expression of f, and using floating-point arithmetic with precision ε, then this enclosure $F(x)$ satisfies:

$$\text{dist}(f(X), F(X)) \leq \mathcal{O}(w(X)) + \mathcal{O}(\varepsilon).$$

• **Moral**: the computing precision can limit the bisection and the achieved accuracy.

• **Moral**: arbitrary computing precision must be available.

MP and interval arithmetic
Mathematica and the intervals

(Keiper 1993, PolKA project 1999)

Some existing solutions are not suitable...

```
e = 15-39 Sin [EulerGamma] + 2 pi;
N[Interval[{e,e}], 1]
-> Interval[{0.001, 0.001}]

N[Interval[{e,e}], 2]
-> Interval[{0.0012, 0.0012}]
```

When the precision changes, the results are disjoint intervals.

Remark: idem with Maple **(Connell and Corless 1993)**.

Outline

- **Bisection: need for more precision**
  - problem in interval arithmetic: overestimation
  - solution: bisection (theorem and limits)
  - arbitrary precision is needed
- **The MPFI library**
  - general view of MPFI
  - closer view of MPFI
- **Applications**
  - enclosures of polynomial real roots
  - approximations of real roots

MPFI: Multiple Precision Floating-point Interval arithmetic library

- **what:** C library for arbitrary precision interval arithmetic
- **why:** to compute reliably and accurately
- **where:** freely available at **http://www.ens-lyon.fr/˜nrevol**, including source code and documentation (last release: April 2002)
- **who:** N. Revol and F. Rouillier
- **how:** based on MPFR for arbitrary precision floating-point arithmetic

MPFI: Multiple Precision Floating-point Interval arithmetic library

- **Requirements:** directed roundings are required, exact is better. Even for elementary functions!
- **Based on MPFR:** for arbitrary precision floating-point arithmetic and compliance with the IEEE-754 philosophy.
- **Based on GMP:** for efficiency and portability.
- **Implementation choices**
  Language: C.
  Representation: by endpoints, as every operation is software and based on integers.

MPFI: functionalities

- arbitrary precision interval data type
- arithmetic operations: $+$, $-$, $\times$, $/$, $\sqrt{}$
- constants: π, $\log 2$, Euler constant
- elementary functions: exp, log, atan, cosh, asinh. . .
- IEEE-754 special values: ∞, signed zeros, NaN
- conversions to and from integer, double, exact naturals, exact integers, rational, "reals" (MPFR numbers)
- Input/Output

MPFI: planned functionalities

- **In the very near future:**

- more efficiency
- complete compliance with Hickey, Ju and Van Emden, ACM 2001.

- **C++ interface**

- **interface à la Profil/BIAS**

with data types and basic operations for linear algebra.

- **automatic differentiation**

even if quite elementary.

Outline

- **Bisection: need for more precision**
  - problem in interval arithmetic: overestimation
  - solution: bisection (theorem and limits)
  - arbitrary precision is needed
- **The MPFI library**
  - general view of MPFI
  - closer view of MPFI
- **Applications**
  - isolation of polynomial real roots
  - approximations of real roots

Isolation of polynomial real roots (Rouillier and Zimmermann 2001)

Principle: apply Descartes rule of sign to the polynomial.

Problem: exact computations, $i.e.$ costly in term of time.

Revised principle: apply Descartes rule of sign to the polynomial with interval coefficients (containing the exact coefficients)

Algorithm:

while computing precision $\leq$ `MaxPrec`
    convert the input polynomial into an interval polynomial
    if computed intervals have constant signs then
        apply Descartes rule of sign
    else   increase the computing precision
if `MaxPrec` is reached then   exact computations

Interval Newton algorithm

Revol 2001

Input: F, F', X_0 // X_0 initial search interval
Initialization: $\mathcal{L} = \{X_0\}$
Loop: while $\mathcal{L} \neq \emptyset$
Suppress $(X, \mathcal{L})$
Insert X in $\mathcal{R}es$ when X is small enough and $F(X)$ is small enough
Increase the working precision if needed
Compute one step of Newton's algorithm
if X is not reduced enough
$(X_1, X_2) :=$ bisect(X)
Insert X_1 and X_2 in $\mathcal{L}$

Interval Newton algorithm: experiments

Chebychev polynomials: $C_n(\cos(\theta)) = \cos(n\theta)$.

Results: very precise roots for degrees up to 40, with proof of existence and uniqueness.

Wilkinson polynomial: $\prod_{i=1}^{20}(X - i)$.

With enough precision to be able to store exactly the coefficients: roots found with a precision 5.10^{-2} and a proof of existence (but not uniqueness). A lot of intervals are not eliminated: $[0.96, 1.02]$ et $[1.62, 20.984]$.

With enough precision and a perturbation $\pm 2^{-19}$ on the coefficient of X^{19}: roots (with proof of existence but not uniqueness): 1 ± 4.10^{-2}, 2 ± 5.10^{-2}, 3 ± 4.10^{-2}, 4 ± 4.10^{-2}, 5 ± 4.10^{-2}, 6 ± 5.10^{-2}, 7 ± 6.10^{-2} and $[7.91, 22.11]$. A lot of intervals are not eliminated: $[0.96, 22.64]$.

Conclusion and future work

Computing "in the large" is the ultimate goal. . .
but it is NP-hard.

Usual solution = bisection (and subpavings):
it must be possible to bisect as long as desired.
With arbitrary precision arithmetic: no limits to bisection.

MPFI library:
allows arbitrary precision interval computations
users/testers welcome!
more functionalities to be added
more algorithms to be added