There's no Reliable Computing without Reliable Access to Rounding Modes

Christoph Lauter & Valérie Ménissier-Morain

Université Pierre et Marie Curie - Paris 6
LIP6 - Équipe PEQUAN

SCAN 2012, Novosibirsk, September 24th, 2012

Rounding mode changes are a basic requirement

Interval Arithmetic

Does massive amounts of rounding mode changes.
For example, $[a, b] + [c, d] = [\bigtriangledown(a + b), \bigtriangleup(c + d)]$.

Compensated Arithmetic through Errorfree Transforms

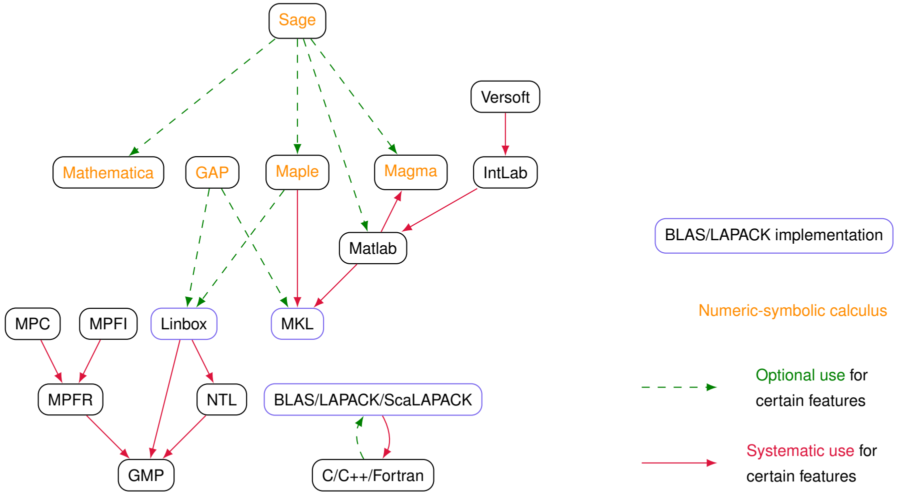
Most algorithms require to be run in Round-To-Nearest mode.

CADNA

The CESTAC method the CADNA software is based on uses random rounding mode changes in order to exhibit numerical instabilities.

Structure of typical Numerical Software Packages

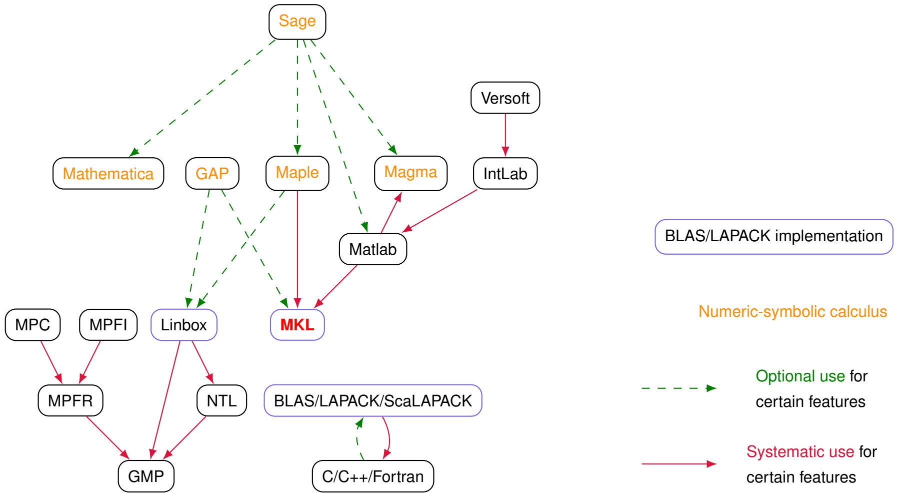
Numerical Software is often based on different other software packages that are often intermingled:



Sage
Versoft
Mathematica
GAP
Maple
Magma
IntLab
Matlab
BLAS/LAPACK implementation
MPC
MPFI
Linbox
MKL
Numeric-symbolic calculus
MPFR
NTL
BLAS/LAPACK/ScaLAPACK
Optional use for certain features
GMP
C/C++/Fortran
Systematic use for certain features

Structure of typical Numerical Software Packages

Numerical Software is often based on different other software packages that are often intermingled:


Sage
Versoft
Mathematica
GAP
Maple
Magma
IntLab
Matlab
BLAS/LAPACK implementation
MPC
MPFI
Linbox
MKL
Numeric-symbolic calculus
MPFR
NTL
BLAS/LAPACK/ScaLAPACK
Optional use for certain features
GMP
C/C++/Fortran
Systematic use for certain features

MKL

Most numerical software packages use the Intel Math Kernel Library (MKL) without really telling their users.

MKL highlights

- One of the best implementations of BLAS, LAPACK, ScaLAPACK1
- Integrates solvers for sparse systems
- FFTs
- Vectorized elementary functions
- Automatic parallelisation for multithreaded or clustered systems
- C and Fortran interfaces
- Redistributable but a black box

From low-level Interval Arithmetic to Matrices

Goal: take advantage of MKL for Interval Arithmetic on Vectors and Matrices

MKL does not enable easy rounding mode changes.

We'd like do things like

```
set roundingmode ...
call MKL
set roundingmode ...
```

and, through high-level reasoning on the MKL code, recover e.g. for some matrix $A = (A_{ij})$ a valid enclosure $([\bigtriangledown(A_{ij}), \bigtriangleup(A_{ij})])$.

Same thing for tricks like $\bigtriangledown(a+b) = -\bigtriangleup\left((-a)+(-b)\right)$, etc.

From low-level Interval Arithmetic to Matrices

Goal: take advantage of MKL for Interval Arithmetic on Vectors and Matrices

MKL does not enable easy rounding mode changes.

We'd like do things like

```
set roundingmode ...
call MKL
set roundingmode ...
```

and, through high-level reasoning on the MKL code, recover e.g. for some matrix $A = (A_{ij})$ a valid enclosure $([\bigtriangledown(A_{ij}), \bigtriangleup(A_{ij})])$.

Same thing for tricks like $\bigtriangledown(a+b) = - \bigtriangleup((-a)+(-b))$, etc.

Our issue: we are reasoning on code we don't know!

Rounding-mode changes: any synchronization?

- Rounding-mode changes are done via `fstcw` on x87, `stmxcsr` on the SSE side: they work on two separate registers!
- C99 `fesetround` guarantees their sychronization but anything can happen when assembly is used.
- In MKL, it's not always the same code that runs: that depends on the processor, its vendor, x87 or SSE capabilities etc.

- What happens in a multi-threaded environment? Do all threads see the same global rounding mode? Their own?
- And on a cluster?

Libraries: why should I care about the rounding-mode?

- GNU/Linux `printf` and `scanf` code never reads the rounding-mode or takes it into account by some other means: decimal vs. binary conversion is always done in round-to-nearest.

- `exp` and other `glibc` elementary functions (see bug report/fix 3976, 29/2/12) can't handle rounding-modes other than round-to-nearest at all. They might even segfault.

Is this Reliable Computing?

Is computing

based on software like that

to be called

Reliable Computing?

Conclusion

We need at least

- that each software component that modifies the rounding-mode says it does in its documentation,
- that most software components integrate easy ways to change the rounding mode,
- and that documentation is available for each component on whether it takes the rounding mode into account or not and under which conditions.

In principle, the whole issue is technical and not scientific problem.
It has become a problem to scientists due to a lack of documentation.

Q/A

Thank your for your attention!

Questions?