

Recent and future developments of GNU MPC

Andreas Enge

LFANT project-team
INRIA Bordeaux-Sud-Ouest
andreas.enge@inria.fr
<http://www.math.u-bordeaux.fr/~aenge>

iRRAM/MPFR/MPC Workshop, Dagstuhl, April 18–20, 2018



MPC in one slide

- **URL:** <http://mpc.multiprecision.org/>
- **Authors:** AE, Paul Zimmermann; Mickaël Gastineau, Philippe Théveny
- **Institutions:** INRIA LFANT and CARAMBA, CNRS
- **Licence:** LGPL 3+
- **Current version:** 1.1.0 “Gladiolus palustris”
- **Language:** C99
- **Dependencies:** GNU MP \geq 5.0.0, GNU MPFR \geq 3.0.0 (both of 2010)
- **Installation**
 - ▶ `./configure; make; make install`



Milestones

- 2002 Version 0.1, AE and PZ
- 2005 Inclusion into Magma
- 2007 Inclusion into Trips, Windows support (MG)
- 2008 Debian packages for squeeze (Laurent Fousse)
- 2007–2009 PT:
 - Trigonometric functions
 - Testing framework
- 2009 Summer school on GNU MPFR and GNU MPC in Nancy
- 2010 **Mandatory for GCC 4.5**
- 2011 Version 0.9, complete with all functions of the C standard
- 2011 First joint GNU MPFR and GNU MPC developers' meeting
- 2011 **GNU project, license change**
- 2012 Version 1.0
- 2018 Version 1.1.0



```
typedef struct {  
    mpfr_t re;  
    mpfr_t im;  
} __mpc_struct;  
typedef __mpc_struct mpc_t[1];
```

- Each coordinate carries its own precision.
- Coordinate access
 - ▶ `mpc_realref (z)`
 - ▶ `mpc_imagref (z)`

- Initialisation

- ▶ `mpc_init2 (z, prec)`
- ▶ `mpc_init3 (z, prec_re, prec_im)`

- Assignment

- ▶ `mpc_set`
- ▶ `mpc_set_fr, mpc_set_ui, ...`
- ▶ `mpc_set_fr_fr, mpc_set_ui_ui, ...`

- Computation

- Freeing

- ▶ `mpc_clear (z)`

- Rounding modes
 - ▶ Separate for each coordinate
 - ▶ $\text{MPC_RNDNU} = \text{MPFR_RNDN} + i \text{MPFR_RNDU}$
- Correct rounding of each coordinate for atomic operation
- Nonary return value
 - ▶ Result exact/rounded down/rounded up by coordinate
 - ▶ $\text{MPC_INEX_RE}(\text{ret}) \in \{0, -1, 1\}$
 - ▶ $\text{MPC_INEX_IM}(\text{ret}) \in \{0, -1, 1\}$
- Comparison
 - ▶ `mpc_cmp (z1, z2)`
 - ▶ Returns nonary value
 - ▶ Value == 0 iff $z1 == z2$

All functions in the C99 standard since version 0.9

- 1.1.0 “Gladiolus palustris”
- Minimally required library versions:
GNU MP 5.0.0 and GNU MPFR 3.0.0
(of 2010)
- Fixed issues with GNU MPFR 4.0.0
- Rewrite of the testing framework
- New `mpcbench` tool, used with `make bench`
- New functions:
 - ▶ `mpc_rootofunity`
 - ▶ `mpc_cmp_abs`
(paving the way for `mpc_agm`)



Small things to do

See also TODO in the distribution or git.

- Simplify code for $\text{GNU MPFR} \geq 3$ (?)
- Improve test coverage (used to be 100%, now is 99.4%...)
- Move hard-coded tests to data files
(for instance, `tests/tcos.c`)

Engineering type things to do

- Publish benchmarks:
scripts and web page comparing timings with different systems,
as done for GNU MPFR at
<http://www.mpfr.org/mpfr-3.1.2/timings.html>
- Support the MPFR_RNDA rounding mode

Algorithmic things to do

- Karatsuba multiplication:
with 3 multiplications at precision around p ,
instead of 2 at $2p$ and 1 at p
- Alternative formulæ for (inverse) trigonometric functions,
depending on (relative) argument sizes
- Correct handling of NaN and infinities in the case of intermediate
overflows while the result may fit

New functions to implement

- Functions with reserved names in C99 and C11
 - ▶ `mpc_erf`, `mpc_erfc`
 - ▶ `mpc_exp2`, `mpc_expm1`
 - ▶ `mpc_log1p`, `mpc_log2`
 - ▶ `mpc_lgamma`, `mpc_tgamma`
- Number theoretic functions (see ARB)
 - ▶ Hurwitz zeta

New number type to implement

Complex ball arithmetic

- Used internally to replace hand calculations of error bounds
- As a tool to implement Taylor and Laurent series
- As a building block for polynomials, class polynomials, etc.
- Only four basic arithmetic operations and square root?
- Need for real ball arithmetic? interface with MPFI?
- Representation as rectangles? as balls?