CHAPTER

FIVE

COMPLEX NUMBERS

The functions described in this chapter provide support for complex numbers. The algorithms take care to avoid unnecessary intermediate underflows and overflows, allowing the functions to be evaluated over as much of the complex plane as possible.

For multiple-valued functions the branch cuts have been chosen to follow the conventions of Abramowitz and Stegun. The functions return principal values which are the same as those in GNU Calc, which in turn are the same as those in "Common Lisp, The Language (Second Edition)"[1] and the HP-28/48 series of calculators.

The complex types are defined in the header file `gsl_complex.h`, while the corresponding complex functions and arithmetic operations are defined in `gsl_complex_math.h`.

5.1 Representation of complex numbers

Complex numbers are represented using the type `gsl_complex`. The internal representation of this type may vary across platforms and should not be accessed directly. The functions and macros described below allow complex numbers to be manipulated in a portable way.

For reference, the default form of the `gsl_complex` type is given by the following struct:

```
typedef struct
{
  double dat[2];
} gsl_complex;
```

The real and imaginary part are stored in contiguous elements of a two element array. This eliminates any padding between the real and imaginary parts, `dat[0]` and `dat[1]`, allowing the struct to be mapped correctly onto packed complex arrays.

gsl_complex **gsl_complex_rect** (double *x*, double *y*)

This function uses the rectangular Cartesian components (x, y) to return the complex number $z = x + iy$. An inline version of this function is used when `HAVE_INLINE` is defined.

gsl_complex **gsl_complex_polar** (double *r*, double *theta*)

This function returns the complex number $z = r \exp(i\theta) = r(\cos(\theta) + i \sin(\theta))$ from the polar representation (`r`, `theta`).

GSL_REAL (z)
GSL_IMAG (z)

These macros return the real and imaginary parts of the complex number `z`.

[1] Note that the first edition uses different definitions.

GSL_SET_COMPLEX (zp, x, y)
This macro uses the Cartesian components (x, y) to set the real and imaginary parts of the complex number pointed to by zp. For example:

```
GSL_SET_COMPLEX(&z, 3, 4)
```

sets z to be $3 + 4i$.

GSL_SET_REAL (zp, x)
GSL_SET_IMAG (zp, y)
These macros allow the real and imaginary parts of the complex number pointed to by zp to be set independently.

5.2 Properties of complex numbers

double **gsl_complex_arg** (gsl_complex *z*)
This function returns the argument of the complex number z, $\arg(z)$, where $-\pi < \arg(z) <= \pi$.

double **gsl_complex_abs** (gsl_complex *z*)
This function returns the magnitude of the complex number z, $|z|$.

double **gsl_complex_abs2** (gsl_complex *z*)
This function returns the squared magnitude of the complex number z, $|z|^2$.

double **gsl_complex_logabs** (gsl_complex *z*)
This function returns the natural logarithm of the magnitude of the complex number z, $\log|z|$. It allows an accurate evaluation of $\log|z|$ when $|z|$ is close to one. The direct evaluation of `log(gsl_complex_abs(z))` would lead to a loss of precision in this case.

5.3 Complex arithmetic operators

gsl_complex **gsl_complex_add** (gsl_complex *a*, gsl_complex *b*)
This function returns the sum of the complex numbers a and b, $z = a + b$.

gsl_complex **gsl_complex_sub** (gsl_complex *a*, gsl_complex *b*)
This function returns the difference of the complex numbers a and b, $z = a - b$.

gsl_complex **gsl_complex_mul** (gsl_complex *a*, gsl_complex *b*)
This function returns the product of the complex numbers a and b, $z = ab$.

gsl_complex **gsl_complex_div** (gsl_complex *a*, gsl_complex *b*)
This function returns the quotient of the complex numbers a and b, $z = a/b$.

gsl_complex **gsl_complex_add_real** (gsl_complex *a*, double *x*)
This function returns the sum of the complex number a and the real number x, $z = a + x$.

gsl_complex **gsl_complex_sub_real** (gsl_complex *a*, double *x*)
This function returns the difference of the complex number a and the real number x, $z = a - x$.

gsl_complex **gsl_complex_mul_real** (gsl_complex *a*, double *x*)
This function returns the product of the complex number a and the real number x, $z = ax$.

gsl_complex **gsl_complex_div_real** (gsl_complex *a*, double *x*)
This function returns the quotient of the complex number a and the real number x, $z = a/x$.

gsl_complex **gsl_complex_add_imag** (gsl_complex *a*, double *y*)
This function returns the sum of the complex number a and the imaginary number iy, $z = a + iy$.

gsl_complex **gsl_complex_sub_imag** (gsl_complex *a*, double *y*)
: This function returns the difference of the complex number `a` and the imaginary number iy, $z = a - iy$.

gsl_complex **gsl_complex_mul_imag** (gsl_complex *a*, double *y*)
: This function returns the product of the complex number `a` and the imaginary number iy, $z = a * (iy)$.

gsl_complex **gsl_complex_div_imag** (gsl_complex *a*, double *y*)
: This function returns the quotient of the complex number `a` and the imaginary number iy, $z = a/(iy)$.

gsl_complex **gsl_complex_conjugate** (gsl_complex *z*)
: This function returns the complex conjugate of the complex number `z`, $z^* = x - iy$.

gsl_complex **gsl_complex_inverse** (gsl_complex *z*)
: This function returns the inverse, or reciprocal, of the complex number `z`, $1/z = (x - iy)/(x^2 + y^2)$.

gsl_complex **gsl_complex_negative** (gsl_complex *z*)
: This function returns the negative of the complex number `z`, $-z = (-x) + i(-y)$.

5.4 Elementary Complex Functions

gsl_complex **gsl_complex_sqrt** (gsl_complex *z*)
: This function returns the square root of the complex number `z`, $\sqrt{z}$. The branch cut is the negative real axis. The result always lies in the right half of the complex plane.

gsl_complex **gsl_complex_sqrt_real** (double *x*)
: This function returns the complex square root of the real number `x`, where `x` may be negative.

gsl_complex **gsl_complex_pow** (gsl_complex *z*, gsl_complex *a*)
: The function returns the complex number `z` raised to the complex power `a`, z^a. This is computed as $\exp(\log(z) * a)$ using complex logarithms and complex exponentials.

gsl_complex **gsl_complex_pow_real** (gsl_complex *z*, double *x*)
: This function returns the complex number `z` raised to the real power `x`, z^x.

gsl_complex **gsl_complex_exp** (gsl_complex *z*)
: This function returns the complex exponential of the complex number `z`, $\exp(z)$.

gsl_complex **gsl_complex_log** (gsl_complex *z*)
: This function returns the complex natural logarithm (base e) of the complex number `z`, $\log(z)$. The branch cut is the negative real axis.

gsl_complex **gsl_complex_log10** (gsl_complex *z*)
: This function returns the complex base-10 logarithm of the complex number `z`, $\log_{10}(z)$.

gsl_complex **gsl_complex_log_b** (gsl_complex *z*, gsl_complex *b*)
: This function returns the complex base-`b` logarithm of the complex number `z`, $\log_b(z)$. This quantity is computed as the ratio $\log(z)/\log(b)$.

5.5 Complex Trigonometric Functions

gsl_complex **gsl_complex_sin** (gsl_complex *z*)
: This function returns the complex sine of the complex number `z`, $\sin(z) = (\exp(iz) - \exp(-iz))/(2i)$.

gsl_complex **gsl_complex_cos** (gsl_complex *z*)
: This function returns the complex cosine of the complex number `z`, $\cos(z) = (\exp(iz) + \exp(-iz))/2$.

gsl_complex **gsl_complex_tan** (gsl_complex *z*)
: This function returns the complex tangent of the complex number `z`, $\tan(z) = \sin(z)/\cos(z)$.

gsl_complex **gsl_complex_sec** (gsl_complex z)
This function returns the complex secant of the complex number z, $\sec(z) = 1/\cos(z)$.

gsl_complex **gsl_complex_csc** (gsl_complex z)
This function returns the complex cosecant of the complex number z, $\csc(z) = 1/\sin(z)$.

gsl_complex **gsl_complex_cot** (gsl_complex z)
This function returns the complex cotangent of the complex number z, $\cot(z) = 1/\tan(z)$.

5.6 Inverse Complex Trigonometric Functions

gsl_complex **gsl_complex_arcsin** (gsl_complex z)
This function returns the complex arcsine of the complex number z, $\arcsin(z)$. The branch cuts are on the real axis, less than -1 and greater than 1.

gsl_complex **gsl_complex_arcsin_real** (double z)
This function returns the complex arcsine of the real number z, $\arcsin(z)$. For z between -1 and 1, the function returns a real value in the range $[-\pi/2, \pi/2]$. For z less than -1 the result has a real part of $-\pi/2$ and a positive imaginary part. For z greater than 1 the result has a real part of $\pi/2$ and a negative imaginary part.

gsl_complex **gsl_complex_arccos** (gsl_complex z)
This function returns the complex arccosine of the complex number z, $\arccos(z)$. The branch cuts are on the real axis, less than -1 and greater than 1.

gsl_complex **gsl_complex_arccos_real** (double z)
This function returns the complex arccosine of the real number z, $\arccos(z)$. For z between -1 and 1, the function returns a real value in the range $[0, \pi]$. For z less than -1 the result has a real part of π and a negative imaginary part. For z greater than 1 the result is purely imaginary and positive.

gsl_complex **gsl_complex_arctan** (gsl_complex z)
This function returns the complex arctangent of the complex number z, $\arctan(z)$. The branch cuts are on the imaginary axis, below $-i$ and above i.

gsl_complex **gsl_complex_arcsec** (gsl_complex z)
This function returns the complex arcsecant of the complex number z, $\mathrm{arcsec}(z) = \arccos(1/z)$.

gsl_complex **gsl_complex_arcsec_real** (double z)
This function returns the complex arcsecant of the real number z, $\mathrm{arcsec}(z) = \arccos(1/z)$.

gsl_complex **gsl_complex_arccsc** (gsl_complex z)
This function returns the complex arccosecant of the complex number z, $\mathrm{arccsc}(z) = \arcsin(1/z)$.

gsl_complex **gsl_complex_arccsc_real** (double z)
This function returns the complex arccosecant of the real number z, $\mathrm{arccsc}(z) = \arcsin(1/z)$.

gsl_complex **gsl_complex_arccot** (gsl_complex z)
This function returns the complex arccotangent of the complex number z, $\mathrm{arccot}(z) = \arctan(1/z)$.

5.7 Complex Hyperbolic Functions

gsl_complex **gsl_complex_sinh** (gsl_complex z)
This function returns the complex hyperbolic sine of the complex number z, $\sinh(z) = (\exp(z) - \exp(-z))/2$.

gsl_complex **gsl_complex_cosh** (gsl_complex z)
This function returns the complex hyperbolic cosine of the complex number z, $\cosh(z) = (\exp(z) + \exp(-z))/2$.

gsl_complex **gsl_complex_tanh** (gsl_complex z)

This function returns the complex hyperbolic tangent of the complex number z, $\tanh(z) = \sinh(z)/\cosh(z)$.

gsl_complex **gsl_complex_sech** (gsl_complex z)

This function returns the complex hyperbolic secant of the complex number z, $\mathrm{sech}(z) = 1/\cosh(z)$.

gsl_complex **gsl_complex_csch** (gsl_complex z)

This function returns the complex hyperbolic cosecant of the complex number z, $\mathrm{csch}(z) = 1/\sinh(z)$.

gsl_complex **gsl_complex_coth** (gsl_complex z)

This function returns the complex hyperbolic cotangent of the complex number z, $\coth(z) = 1/\tanh(z)$.

5.8 Inverse Complex Hyperbolic Functions

gsl_complex **gsl_complex_arcsinh** (gsl_complex z)

This function returns the complex hyperbolic arcsine of the complex number z, $\mathrm{arcsinh}(z)$. The branch cuts are on the imaginary axis, below $-i$ and above i.

gsl_complex **gsl_complex_arccosh** (gsl_complex z)

This function returns the complex hyperbolic arccosine of the complex number z, $\mathrm{arccosh}(z)$. The branch cut is on the real axis, less than 1. Note that in this case we use the negative square root in formula 4.6.21 of Abramowitz & Stegun giving $\mathrm{arccosh}(z) = \log(z - \sqrt{z^2 - 1})$.

gsl_complex **gsl_complex_arccosh_real** (double z)

This function returns the complex hyperbolic arccosine of the real number z, $\mathrm{arccosh}(z)$.

gsl_complex **gsl_complex_arctanh** (gsl_complex z)

This function returns the complex hyperbolic arctangent of the complex number z, $\mathrm{arctanh}(z)$. The branch cuts are on the real axis, less than -1 and greater than 1.

gsl_complex **gsl_complex_arctanh_real** (double z)

This function returns the complex hyperbolic arctangent of the real number z, $\mathrm{arctanh}(z)$.

gsl_complex **gsl_complex_arcsech** (gsl_complex z)

This function returns the complex hyperbolic arcsecant of the complex number z, $\mathrm{arcsech}(z) = \mathrm{arccosh}(1/z)$.

gsl_complex **gsl_complex_arccsch** (gsl_complex z)

This function returns the complex hyperbolic arccosecant of the complex number z, $\mathrm{arccsch}(z) = \mathrm{arcsinh}(1/z)$.

gsl_complex **gsl_complex_arccoth** (gsl_complex z)

This function returns the complex hyperbolic arccotangent of the complex number z, $\mathrm{arccoth}(z) = \mathrm{arctanh}(1/z)$.

5.9 References and Further Reading

The implementations of the elementary and trigonometric functions are based on the following papers,

- T. E. Hull, Thomas F. Fairgrieve, Ping Tak Peter Tang, "Implementing Complex Elementary Functions Using Exception Handling", ACM Transactions on Mathematical Software, Volume 20 (1994), pp 215–244, Corrigenda, p553
- T. E. Hull, Thomas F. Fairgrieve, Ping Tak Peter Tang, "Implementing the complex arcsin and arccosine functions using exception handling", ACM Transactions on Mathematical Software, Volume 23 (1997) pp 299–335

The general formulas and details of branch cuts can be found in the following books,

- Abramowitz and Stegun, Handbook of Mathematical Functions, “Circular Functions in Terms of Real and Imaginary Parts”, Formulas 4.3.55–58, “Inverse Circular Functions in Terms of Real and Imaginary Parts”, Formulas 4.4.37–39, “Hyperbolic Functions in Terms of Real and Imaginary Parts”, Formulas 4.5.49–52, “Inverse Hyperbolic Functions—relation to Inverse Circular Functions”, Formulas 4.6.14–19.
- Dave Gillespie, Calc Manual, Free Software Foundation, ISBN 1-882114-18-3