Hardest-to-Round Cases

Vincent LEFÈVRE

Arénaire, INRIA Grenoble – Rhône-Alpes / LIP, ENS-Lyon

Journées TaMaDi, Lyon, 2010-10-28

Outline

Hardest-to-round cases in:

- binary64 (Double Precision)
- x87 Extended Precision
- decimal64

Hardest-to-Round Cases in binary64 (Double Precision)

- Floating-point system in radix 2.
- Double precision ($p = 53$).
- **No subnormals.**
  In input, the exponent range will be extended to include subnormals.
- Exact cases are regarded as hard-to-round cases (stored in the database). Exactness is checked by `readres` with GNU MPFR and these cases are not output.
- Algorithm used: L-algorithm (first step).

Hardest-to-Round Cases in binary64 (Double Precision) [2]

After $5,537,656$ hours (631.7 years) for the first step, in summary:

- e^x, 2^x, 10^x, sinh, cosh, $\sin(2\pi x)$, $\cos(2\pi x)$, $\tan(2\pi x)$;
- x^n for $3 \leq n \leq 2767$ and $-180 \leq n \leq -2$;
- sin, cos, tan between $-\pi/2$ and $\pi/2$;
- the corresponding inverse functions.

exp and log

- Function exp tested between 2^{-1} and $x_p = (1 + 25405/2^{16}) \cdot 2^9$.
  Note: $\log_2(\exp(x_p)) > 1025$ (instead of 1024, in order to obtain the HR cases of cosh and sinh).
- Function exp tested between -2^{-1} and $x_m = -(1 + 29753/2^{16}) \cdot 2^9$.
  Note: $\log_2(\exp(x_m)) < -1074$.
- Function log tested between 2^{-1} and 2^1 [exponents -1 and 0].

We have $\log(2^{-1}) < -2^{-1}$ and $\log(2^1) > 2^{-1}$, therefore all the HR cases in the domain of the double-precision format.

Function exp

```
$ ./readres -x 57 -f hr -s dwim archres/results.exm.*
$ ./readres -y 57 -f hr -s dwim archres/results.log.-1.54.{1..9}
Domain: ( -1.0111010000111001e9 , -1e-36 ] => [ log(1e-1074) , -1e-36 ]
exp-1.12D31A20FB38BP5=1.5B0BF3244820AP-50:01[58]0010
exp-1.A2FEFEFD580DFP-13=1.FFE5D0BB7EABFP-1:00[57]1100
exp-1.ED318EFB627EAP-27=1.FFFFFF84B39C4P-1:11[59]0001
exp-1.3475AC05CEAD7P-29=1.FFFFFFECB8A54P-1:00[57]1001
```

```
$ ./readres -y 57 -f hr -s dwim archres/results.log.0.54.{1..8}
$ ./readres -x 57 -f hr -s dwim archres/results.exp.*
Domain: [ 1e-31 , 1.0110001100111101e9 ) => [ 1e-31 , log(1e1025) )
exp 1.9E9CBBFD6080BP-31=1.000000033D397P0:10[57]1010
exp 1.83D4BCDEBB3F4P2=1.AC50B409C8AEEP8:00[57]1000
```

Special HR cases:

```
$ ./readres -y 100 -f hr -s dwim archres/results.log.-1.54.*
Domain: ( log(1e-1074) , -1e-54 ]
exp-1.0000000000001P-51=1.FFFFFFFFFFFFCP-1:00[100]1010
```

```
$ ./readres -y 102 -f hr -s dwim archres/results.log.0.54.*
Domain: [ 1e-53 , log(1e1025) )
exp 1.FFFFFFFFFFFFFP-53=1.0000000000000P0:11[104]0101
```

Function log

```
$ ./readres -y 60 -f hr -s dwim archres/results.exm.*
Domain: [ 1e-1074 , 1e-1 )
log 1.EA71D85CEE020P-509=-1.60296A66B42FFP8:11[60]0000
log 1.9476E304CD7C7P-384=-1.09B60CAF47B35P8:10[60]1010
log 1.26E9C4D327960P-232=-1.4156584BCD084P7:00[60]1001
log 1.613955DC802F8P-35=-1.7F02F9BAF6035P4:01[60]0011

$ ./readres -x 54 -y 200 -f hr -s dwim archres/results.log.*
Domain: [ 1e-1 , 1e1 )
log 1.BADED30CBF1C4P-1=-1.290EA09E36478P-3:11[54]0110

$ ./readres -y 63 -f hr -s dwim archres/results.exp.*
Domain: [ 1e1 , 1e1025 )
log 1.C90810D354618P245=1.54CD1FEA76639P7:11[63]0101
log 1.62A88613629B6P678=1.D6479EBA7C971P8:00[64]1110
```

expm1 and log1p

- Function expm1 tested between 2^{-51} and $(1 + 1/16) \cdot 2^6$.
- Function expm1 tested between -2^{-51} and -2^1 [exponents -51 to 0].
- Function log1p tested between -2^{-1} and -2^0 [exponent -1].

Using the results of exp and log, we have all the HR cases in the domain of the double-precision format.

Function expm1

```
$ ./readres -x 58 -f hr -s dwim archres/results.em1.{{0..6},-{1..35}}.54
Domain: [ 1e-35 , 1.0001e6 ) => [ 1e-35 , log(1e1024) )
em1 1.274BBF1EFB1A2P-10=1.2776572C25129P-10:10[58]1000
```

```
$ ./readres -x 56 -f hr -s dwim archres/results.emm.{0,-{1..34}}.54
$ ./readres -y 56 -f hr -s dwim archres/results.l1m.-1.54.*
Domain: ( -inf , -1e-34 ]
em1-1.19E53FCD490D0P-23=-1.19E53E96DFFA8P-23:10[56]1110
```

These results do *not* include the cases that round to -1.

Special HR cases:

```
$ ./readres -x 95 -f hr -s dwim archres/results.em1.*
Domain: [ 1e-51 , 1.0001e6 ) => [ 1e-51 , log(1e1024) )
em1 1.7FFFFFFFFFFFDP-49=1.8000000000005P-49:11[96]0110
```

```
$ ./readres -x 95 -f hr -s dwim archres/results.emm.*
Domain: ( -1e1 , -1e-51 ] => ( -inf , -1e-51 ]
em1-1.8000000000003P-49=-1.7FFFFFFFFFFFAP-49:00[96]1000
```

Function log1p

```
$ ./readres -y 59 -f hr -s dwim archres/results.em1.{{0..6},-{1..36}}.54
Domain: [ 1e-35 , 1e98 ]
l1p 1.AB50B409C8AEEP8=1.83D4BCDEBB3F3P2:11[60]0101
l1p 1.8AA92BC84FF91P54=1.2EE70220FB1C4P5:11[60]0011
l1p 1.0410C95B580B9P71=1.89D56A0C38E6FP5:00[62]1011
```

Note: HR cases for $x > 2^{98}$ are obtained from those of function `log`.
To take the error into account, we subtract 1 from k. Hence...

```
$ for i in archres/results.exp.[78].54; do \
  ./readres -y 62 -f hr -s dwim <(perl -p -e 's/exp$/em1/' $i); done
Domain: [ 1e-35 , 1e1024 )
l1p 1.C90810D354618P245=1.54CD1FEA76639P7:11[63]0101
l1p 1.62A88613629B6P678=1.D6479EBA7C971P8:00[64]1110
```

after suppressing the HR case with $k = 62$.

Function log1p [2]

```
$ ./readres -y 58 -f hr -s dwim archres/results.emm.{0,-{1..35}}.54
$ ./readres -y 58 -f hr -s dwim archres/results.l1m.-1.54.*
Domain: ( -1 , -1e-35 ]
l1p-1.7FFFF3FCFFD03P-30=-1.7FFFF4017FCFEP-30:10[58]1001
```

Special HR cases:

```
$ ./readres -y 98 -f hr -s dwim archres/results.em1.*
Domain: ( 1e-51 , 1e98 ] => ( 1e-51 , 1e1024 )
l1p 1.8000000000003P-50=1.7FFFFFFFFFFFEP-50:10[99]1000
```

```
$ ./readres -y 98 -f hr -s dwim archres/results.emm.*
Domain: [ -1e-1 , -1e-51 ] => ( -1 , -1e-51 ]
l1p-1.7FFFFFFFFFFFDP-50=-1.8000000000001P-50:01[99]0110
```

sinh and asinh

Function sinh tested between 2^{-25} and $x_s = (1 + 25317/2^{16}) \cdot 2^9$.

Since $\log_2(\sinh(x_s)) > 1024$, we have all the HR cases in the domain of the double-precision format. Note that for x large enough, we could have used the results of exp.

```
$ ./readres -x 56 -f hr -s dwim archres/results.sh.*
Domain: [ 1e-25 , 1.0110001011100101e9 ) => [ 1e-25 , asinh(1e1024) )
sh 1.DFFFFFFFFE3EP-20=1.E00000000FD1P-20:11[72]0001
sh 1.DFFFFFFFF8F8P-19=1.E00000003F47P-19:11[66]0001
sh 1.DFFFFFFFE3E0P-18=1.E00000000FD1FP-18:11[60]0001
sh 1.67FFFFFFFD08AP-17=1.680000001AB25P-17:11[57]0000
sh 1.897374D74DE2AP-13=1.897374FE073E1P-13:10[56]1011

$ ./readres -y 62 -f hr -s dwim archres/results.sh.*
Domain: [ 1e-25 , 1e1024 )
ash 1.E00000000FD2P-20=1.DFFFFFFFFE3EP-20:00[72]1110
ash 1.E00000003F48P-19=1.DFFFFFFFF8F8P-19:00[66]1110
ash 1.C90810D354618P244=1.54CD1FEA76639P7:11[63]0101
ash 1.8670DE0B68CADP655=1.C7206C1B753E4P8:00[62]1111
ash 1.62A88613629B6P677=1.D6479EBA7C971P8:00[64]1110
```

cosh and acosh

Function cosh tested between 2^{-25} and 2^6 [exponents -25 to 5].

As for $x \geq 2^6$, we can use the results for exp or sinh, we have all the HR cases in the domain of the double-precision format.

Function cosh

```
$ ./readres -x 57 -f hr -s dwim archres/results.ch.*
Domain: [ 1e-25 , 1e6 )
ch 1.465655F122FF5P-24=1.000000000000CP0:11[61]0001
ch 1.7FFFFFFFFFF7P-23=1.0000000000047P0:11[89]0010
ch 1.7FFFFFFFFFDCP-22=1.000000000011FP0:11[83]0010
ch 1.7FFFFFFFFF70P-21=1.000000000047FP0:11[77]0010
ch 1.7FFFFFFFFDC0P-20=1.00000000011FFP0:11[71]0010
ch 1.1FFFFFFFFF0DP-20=1.0000000000A1FP0:11[73]0110
ch 1.DFFFFFFFFB9BP-20=1.0000000001C1FP0:11[69]0010
ch 1.1FFFFFFFFC34P-19=1.000000000287FP0:11[67]0110
ch 1.7FFFFFFFF700P-19=1.00000000047FFP0:11[65]0010
ch 1.DFFFFFFFEE6CP-19=1.000000000707FP0:11[63]0010
ch 1.1FFFFFFFF0D0P-18=1.000000000A1FFP0:11[61]0110
ch 1.4FFFFFFFE7E2P-18=1.000000000DC7FP0:11[60]0011
ch 1.7FFFFFFFDC00P-18=1.000000011FFFP0:11[59]0010
ch 1.AFFFFFFFCCBEP-18=1.0000000016C7FP0:11[58]0010
ch 1.DFFFFFFFB9B0P-18=1.000000001C1FFP0:11[57]0010
ch 1.EA5F2F2E4B0C5P1=1.710DB0CD0FED5P4:10[57]1110
```

Note: HR cases for $x \geq 2^6$ are obtained from those of function `exp` (or `sinh`).
There are none for `sinh` on `-x 56`, thus none for `cosh` on `-x 57`.

Function acosh

```
$ ./readres -y 59 -f hr -s dwim archres/results.ch.*
$ ./readres -x 59 -y 99 -f hr -s dwim archres/results.ach.0.54.0
Domain: [ 1 , 1e91 ]
ach 1.297DE35D02E90P13=1.3B562D2651A5DP3:01[61]0001
ach 1.91EC4412C344FP85=1.E07E71BFCF06EP5:11[61]0101
```

Note: HR cases for $x > 2^{91}$ are obtained from those of function `log`.
For instance, the second HR case above corresponds to:

```
log 1.91EC4412C344FP86=1.E07E71BFCF06EP5:11[61]0101
```

Thus,

```
Domain: [ 1 , 1e1024 )
ach 1.C90810D354618P244=1.54CD1FEA76639P7:11[63]0101
ach 1.62A88613629B6P677=1.D6479EBA7C971P8:00[64]1110
```

tanh and atanh

Not tested yet.

sin and asin

Function sin tested between 2^{-25} and $(1 + 4675/2^{13}) \cdot 2^1$ $(< \pi)$.

```
$ ./readres -x 59 -f hr -s dwim archres/results.sin.*
Domain: [ 1e-25 , u ) { u = 1.1001001000011e1 }
sin 1.E000000001C2P-20=1.DFFFFFFFF02EP-20:00[72]1110
sin 1.E00000000708P-19=1.DFFFFFFFC0B8P-19:00[66]1110
sin 1.E00000001C20P-18=1.DFFFFFFF02E0P-18:00[60]1110
sin 1.598BAE9E632F6P-7=1.598A0AEA48996P-7:01[59]0000
sin 1.FE767739D0F6DP-2=1.E9950730C4695P-2:11[65]0000

$ ./readres -y 57 -f hr -s dwim archres/results.sin.*
Domain: [ 1e-25 , 1 ]
asn 1.DFFFFFFFFF02EP-20=1.E000000001C1P-20:11[72]0001
asn 1.DFFFFFFFC0B8P-19=1.E00000000707P-19:11[66]0001
asn 1.DFFFFFFF02E0P-18=1.E00000001C1FP-18:11[60]0001
asn 1.67FFFFFFE54DAP-17=1.6800000002F75P-17:11[57]0000
asn 1.C373FF4AAD79BP-14=1.C373FF594D65AP-14:10[57]1010
asn 1.E9950730C4696P-2=1.FE767739D0F6DP-2:00[64]1000
```

cos and acos

- Function cos tested between 2^{-25} and $(1 + 4675/2^{13}) \cdot 2^0$ $(< \pi/2)$ and between $(1 + 4676/2^{13}) \cdot 2^0$ $(> \pi/2)$ and $2^2 = 4$ $(> \pi)$.
- Function acos tested between 2^{-26} and 2^{-4} [exponents -26 to -5].
- Function acos tested between -2^{-27} and -2^{-4} [exponents -27 to -5].

Note: $0 < \cos((1 + 4675/2^{13}) \cdot 2^0) < 2^{-4}$ and $-2^{-4} < \cos((1 + 4676/2^{13}) \cdot 2^0) < 0$.

Function cos

```
$ ./readres -x 56 -f hr -s dwim archres/results.cos.{0,1,-{1..17}}.*
$ ./readres -y 56 -f hr -s dwim archres/results.acs.*
$ ./readres -y 56 -f hr -s dwim archres/results.acm.*
Domain: [ 1e-17 , acos(1e-26) ) U [ acos(-1e-27) , 1e2 )
cos 1.06B505550E6B2P-9=1.FFFFBC9A3FBFEP-1:00[58]1100
cos 1.34EC2F9FC9C00P1=-1.7E2A5C30E1D6DP-1:01[58]0110
```

Special HR cases:

```
$ ./readres -x 83 -f hr -s dwim archres/results.cos.*
$ ./readres -y 83 -f hr -s dwim archres/results.acs.*
Domain: [ 0 , acos(1e-26) ) U [ acos(-1e-27) , 1e2 )
cos 1.8000000000009P-23=1.FFFFFFFFFF70P-1:00[88]1101
```

Function acos

```
$ ./readres -y 57 -f hr -s dwim archres/results.cos.{-*,0.54.1}
$ ./readres -x 57 -y 99 -f hr -s dwim archres/results.acs.*
Domain: [ 1e-26 , 1 ]
acs 1.7283D529A146EP-19=1.921F86F3C82C5P0:00[58]1000
acs 1.FD737BE914578P-11=1.91E006D41D8D8P0:11[62]0010
acs 1.1CDCD1EA2AD3BP-9=1.919146D3F492EP0:11[57]0010
acs 1.60CB9769D9218P-8=1.90BEE93D2D09CP0:00[57]1011
acs 1.53EA6C7255E88P-4=1.7CDACB6BBE707P0:01[57]0101

$ ./readres -y 56 -f hr -s dwim archres/results.cos.{0.54.2,1.54}
$ ./readres -x 56 -f hr -s dwim archres/results.acm.*
Domain: [ -1 , -1e-27 ]
acs-1.52F06359672CDP-2=1.E87CCC94BA418P0:10[56]1101
acs-1.124411A0EC32EP-5=1.9AB23ECDD436AP0:10[56]1101
```

[tamadi2010.tex 40124 2010-10-27 23:50:05Z vinc17/xvii]

tan and atan

- Function tan tested between 2^{-25} and 2^{-3} [exponents -25 to -4].
- Function tan tested between $1 + 2570638124657920/2^{52}$ and $1 + 2570638124657945/2^{52}$.
- Function atan tested between 2^{-4} and 2^{48} [exponents -4 to 47].

Note: $\tan(2^{-3}) > 2^{-4}$ and $\tan(1 + 2570638124657920/2^{52}) < 2^{48}$. Therefore we have all the HR cases of atan in the domain of the double-precision format and all the HR cases of tan between $-\pi/2$ and $\pi/2$.

Functions tan and atan

```
$ ./readres -x 56 -y 99 -f hr -s dwim archres/results.tan.*
$ ./readres -y 56 -f hr -s dwim archres/results.atn.*
Domain: [ 1e-25 , pi/2 )
tan 1.DFFFFFFFFF1FP-22=1.E00000000151P-22:01[78]0100
tan 1.DFFFFFFFFC7CP-21=1.E00000000545P-21:11[72]0100
tan 1.DFFFFFFFF1F0P-20=1.E00000001517P-20:11[66]0100
tan 1.67FFFFFFE845P-19=1.6800000002398P-19:01[63]0100
tan 1.DFFFFFFFC7C0P-19=1.E0000000545FP-19:11[60]0100
tan 1.67FFFFFFA114P-18=1.680000008E61P-18:11[57]0100
tan 1.50486B2F87014P-5=1.5078CEBFF9C72P-5:10[57]1001

$ ./readres -x 57 -f hr -s dwim archres/results.atn.*
Domain: ( 1e-25 , +inf )
atn 1.E00000000546P-21=1.DFFFFFFFFC7CP-21:00[72]1011
atn 1.E00000001518P-20=1.DFFFFFFFF1F0P-20:00[66]1011
atn 1.E00000005460P-19=1.DFFFFFFFC7C0P-19:00[60]1011
atn 1.680000008E62P-18=1.67FFFFFFA114P-18:00[57]1011
atn 1.22E8D75E2BC7FP-11=1.22E8D5694AD2BP-11:10[59]1101
atn 1.6298B5896ED3CP1=1.3970E827504C6P0:10[63]1101
atn 1.C721FD48F4418P19=1.921FA3447AF55P0:10[58]1011
atn 1.EB19A7B5C3292P29=1.921FB540173D6P0:11[59]0011
atn 1.CCDA26AD0CD1CP47=1.921FB54442D06P0:01[57]0111
```

$\sin(2\pi x)$ and its Inverse Function

Function $\sin(2\pi x)$ tested between 2^{-58} and 2^{-1} [exponents -58 to -2].

HR cases for exponents below -58 have the same significand as those for exponent -58 (the number of identical bits after the rounding bit of the result can differ by at most 1). In fact, it was not need to completely test the lowest exponents, but as these tests did not require much time, they were performed.

We have all the HR cases (note: testing exponent -2 was redundant).

$\cos(2\pi x)$ and its Inverse Function

- Function $\cos(2\pi x)$ tested between 2^{-25} and 2^{-1} [exponents -25 to -2].
- Inverse function tested between $1 - 2^{-45}$ and 1.

Since $\cos(2\pi \cdot 2^{-25}) > 1 - 2^{-45}$, we have all the HR cases (again, testing exponent -2 was redundant).

$\tan(2\pi x)$ and its Inverse Function

- Function $\tan(2\pi x)$ tested between 2^{-58} and 2^{-6} [exponents -58 to -7].
- Function $\tan(2\pi x)$ tested between $2^{-2} \cdot (1 - 2^{-45})$ and 2^{-2}.
- Function $\arctan(x)/(2\pi)$ tested between 2^{-4} and 2^{48} [exponents -4 to 47].

Note: $\tan(2\pi \cdot 2^{-6}) > 2^{-4}$ and $\tan(2\pi \cdot 2^{-2}.(1 - 2^{-45})) < 2^{48}$. Therefore we have all the HR cases of $\arctan(x)/(2\pi)$ in the domain of the double-precision format and all the HR cases of $\tan 2\pi x$ between $-1/4$ and $1/4$.

exp2 and log2

- Function exp2 tested between 2^{-1} and 2^1 [exponents -1 and 0].
- Results generated by a script for the exponents from 1 to 4.
- Function exp2 tested between $2^5 = 32$ and 33.
- Results generated by a script for the exponents from 6 to 9.
- Function log2 tested between 2^{-1} and 2^1 [exponents -1 and 0].

We have all the HR cases in the domain of the double-precision format. Note: due to the results of log2, the exponent -1 of exp2 was tested just for a consistency check.

Functions exp2 and log2

```
$ ./readres -y 58 -f hr -s dwim archres/results.lg2.*
Domain: ( -inf , +inf )
ex2 1.12B14A318F904P-27=1.00000017CCE02P0:00[58]1101
ex2 1.BFBBDE44EDFC5P-25=1.0000009B2C385P0:00[59]1011
ex2 1.E4596526BF94DP-10=1.0053FC2EC2B53P0:01[59]0100
```

```
$ ./readres -y 53 -f hr -s dwim archres/results.ex2.*
Domain: [ 1e-1 , 1e1024 )
lg2 1.B4EBE40C95A01P0=1.8ADEAC981E00DP-1:10[53]1011
lg2 1.1BA39FF28E3EAP2=1.12EECF76D63CDP1:00[53]1001
lg2 1.1BA39FF28E3EAP4=1.097767BB6B1E6P2:10[54]1001
lg2 1.61555F75885B4P128=1.00EE05A07A6E7P7:11[53]0011
lg2 1.D30A43773DD1BP256=1.00DE0E189B724P8:10[53]1100
lg2 1.61555F75885B4P256=1.007702D03D373P8:11[54]0011
lg2 1.61555F75885B4P512=1.003B81681E9B9P9:11[55]0011
```

Note: if $x \geq 4$, only HR cases whose exponent is a power of 2 are given.

exp10 and log10

- Function `exp10` tested between 2^{-2} and $x_{p10} = (1 + 13378/2^{16}) \cdot 2^8$.
  Note: $\log_2(\exp_{10}(x_{p10})) > 1024$.
- Function `exp10` tested between -2^{-2} and $x_{m10} = -(1 + 17231/2^{16}) \cdot 2^8$.
  Note: $\log_2(\exp_{10}(x_{m10})) < -1074$.
- Function `log10` tested between 2^{-1} and 2^1 [exponents -1 and 0].

We have $\log_{10}(2^{-1}) < -2^{-2}$ and $\log_{10}(2^1) > 2^{-2}$, therefore all the HR cases in the domain of the double-precision format.

Function exp10

```
$ ./readres -x 58 -f hr -s dwim archres/results.e10.*
$ ./readres -y 58 -f hr -s dwim archres/results.l10.0.54.*
Domain: ( 0 , 1.001101000100001e8 ) => ( 0 , log10(1e1024) )
e10 1.DF760B2CDEED3P-49=1.0000000000022P0:10[58]1110
e10 1.A83B1CF779890P-26=1.000000F434FAAP0:01[60]0101
e10 1.7C3DDD23AC8CAP-10=1.00DB40291E4F5P0:01[58]0010
e10 1.AA6E0810A7C29P-2=1.4DEC173D50B3EP1:01[58]0001
e10 1.D7D271AB4EEB4P-2=1.71CE472EB84C7P1:11[64]0111
e10 1.75F49C6AD3BADP0=1.CE41D8FA665F9P4:11[64]0101

$ ./readres -x 58 -f hr -s dwim archres/results.e01.*
$ ./readres -y 58 -f hr -s dwim archres/results.l10.-1.54.*
Domain: ( -1.010000110100111e8 , 0 ) => ( log10(1e-1074) , 0 )
e10-1.1416C72A588A6P-1=1.27D838F22D09FP-2:11[65]0010
e10-1.F28E0E25574A5P-32=1.FFFFFF70811EP-1:00[59]1011
```

Function log10

```
$ ./readres -y 61 -f hr -s dwim archres/results.e01.*
Domain: [ 1e-1074 , 1e-1 )
l10 1.365116686B078P-765=-1.CC68A4AEE240DP7:01[61]0110
l10 1.83E55C0285C96P-762=-1.CA68A4AEE240DP7:01[61]0110
l10 1.A8639E89F5E46P-625=-1.77D933C1A88E0P7:11[61]0101
l10 1.ED8C87C3BF5CFP-49=-1.CEE46399392D6P3:01[62]0000
l10 1.27D838F22D0A0P-2=-1.1416C72A588A5P-1:11[65]0101

$ ./readres -x 54 -y 99 -f hr -s dwim archres/results.l10.*
Domain: [ 1e-1 , 1e1 )
l10 1.B0CF736F1AE1DP-1=-1.2AE5057CD8C44P-4:01[54]0110
l10 1.89825F74AA6B7P0=1.7E646F3FAB0D0P-3:10[57]1001

$ ./readres -y 62 -f hr -s dwim archres/results.e10.*
Domain: [ 1e1 , 1e1024 )
l10 1.71CE472EB84C8P1=1.D7D271AB4EEB4P-2:00[64]1010
l10 1.CE41D8FA665FAP4=1.75F49C6AD3BADP0:00[66]1010
l10 1.E12D66744FF81P429=1.02D4F53729E44P7:10[68]1001
```

$1/x^2$ and $1/\sqrt{x}$

Function $1/x^2$ tested between 2^{-1} and 2^0 [exponent -1].

This is sufficient to obtain all the HR cases of $1/x^2$ and $1/\sqrt{x}$.

```
$ ./readres -x 52 -f hr -s dwim archres/results.isq.-1.54
Domain: [ 1e-1 , 1 )
isq 1.35DABB9F8EE46P-1=1.5D7D6279EE6A4P1:10[52]1000
```

```
$ ./readres -y 52 -f hr -s dwim archres/results.isq.-1.54
Domain: [ 1 , 1e2 )
isr 1.FFFFFFFFFFFFEP1=1.0000000000000P-1:10[52]1100
isr 1.C562B857453DDP1=1.100B926DF6E72P-1:10[53]1001
isr 1.752BA6B1AB340P1=1.2BDCA562D725CP-1:11[53]0001
isr 1.A3D11B87E09A8P0=1.8FD0DC83C987BP-1:11[52]0110
isr 1.A6A9CC15ABCCEP0=1.8E77A118A3095P-1:01[57]0100
```

x^n and $x^{1/n}$, With n Integer

Function x^n tested between 2^0 and 2^1 [exponent 0] for $3 \le n \le 2767$ (and $-180 \le n \le -2$).

This is sufficient to obtain all the HR cases of x^n and $x^{1/n}$ for these values of n.

Total number of HR cases:

Type $(0, *)$	1416034
Type $(*, 0)$	1415680
Expected	1415680

```
p81 1.EFE513B583325P0=1.33A5036CB10FEP77:00[62]1111
p458 1.0F38CFAACB71AP0=1.1F0B0876BA025P38:10[61]1110
p878 1.B01F3FBE19C40P0=1.1BDF0591D5E42P663:01[62]0101
p952 1.5C69202D46821P0=1.3B993E08AAD26P423:10[63]1001
p1030 1.EA2248BD951F6P0=1.1AB491BDA9399P965:10[62]1001
p1776 1.C72CE7406B3CEP0=1.7D646B1EE4F67P1474:01[64]0011
p2309 1.90DC35E30BD1AP0=1.C3EC89C7763F1P1493:00[61]1001

r1039 1.DCBA0C48B3F29P253=1.2F4027B25ACDFP0:01[73]0100
r1856 1.C20BDF1F91DF1P317=1.2043069BA0526P0:01[72]0001
r1907 1.AC171E04B83E0P137=1.0D24A15B3F0AFP0:10[73]1101
r2309 1.C3EC89C7763F1P1493=1.90DC35E30BD19P0:11[73]0001
r2414 1.3381E4A54361FP736=1.3C4444751CD50P0:01[72]0110
r2510 1.EB2C9F5026338P2169=1.D21B70F594C52P0:10[72]1011
r2600 1.F6E80DB95028FP1299=1.6A0942DF30B18P0:10[72]1101
```

Hardest-to-Round Cases in x87 Extended Precision

Joint work with Damien Stehlé and Paul Zimmermann.

Search:

- 2^x in binary between $1/2$ and 1 with a 64-bit significand size.
- All the HR cases with at least 56 identical bits after the rounding bit.
- Algo: SLZ from 2004-03-21 to 2005-01-10 (69379 hours, using 5 clusters).

Results: up to 63 identical bits after the rounding bit.

- 1 HR case in rounding to nearest;
- 1 HR case in directed rounding.

Details on: `http://www.loria.fr/equipes/spaces/slz.html`

Hardest-to-Round Cases in decimal64

Joint work with Damien Stehlé and Paul Zimmermann.

Search:

- Exponential function in the IEEE-754 decimal64 format (16 digits).
- All HR cases whose distance from a breakpoint (for all rounding modes) is less than 10^{-15} ulp.
- Algo: SLZ from 2006-03-12 (?) to 2006-08-28.

The HR case for $|x| \geq 3 \times 10^{-11}$:

- $x = 9.407822313572878 \times 10^{-2}$
- $\exp(x) = 1.098645682066338\,5\,0000000000000000\,278\ldots$

V. Lefèvre, D. Stehlé, and P. Zimmermann. Worst cases for the exponential function in the IEEE 754r decimal64 format. In *Reliable Implementation of Real Number Algorithms: Theory and Practice*, vol. 5045 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 114–126, 2008.

http://dx.doi.org/10.1007/978-3-540-85521-7_7

http://hal.inria.fr/inria-00068731