Some algebraic properties of floating-point arithmetic

Jean-Michel Muller

Some algebraic properties of floating-point arithmetic

Jean-Michel Muller
CNRS-LIP
Projet CNRS-ENSL-INRIA Arenaire
École Normale Supérieure de Lyon
46 Allée d'Italie, 69364 Lyon Cedex 07
France

Abstract

Thanks to the IEEE-754 standard, floating-point arithmetic is now a well-defined mathematical structure, on which it is possible to build proofs and algorithms. We give some examples of properties (mainly closure properties) that can be proven.

Introduction

For many years, floating-point arithmetic has been a mere set of cooking recipes. The consequences of this have sometimes been disastrous: numerical programs were not reliable nor portable. Without a clear specification of the underlying arithmetic, it was not possible to prove even simple properties of a sequence of operations, and the only way to feel comfortable with an important numerical program was to perform intensive tests.

The IEEE-754 [3, 1] standard for binary floating-point arithmetic (and the radix independent IEEE-854 [2, 5] standard that followed) put an end to this dangerous era. The IEEE-754 standard (we will later refer to it as "the IEEE standard" or "the standard") clearly specifies the formats of the floating-point representations of numbers, and the behaviour of the four arithmetic operations.

Define F_n as the set of exponent-unbounded, n-bit mantissa, binary floating-point numbers (with $n \geq 1$), that is:

$$F_n = \left\{ M \times 2^E, 2^{n-1} \leq M \leq 2^n - 1, M, E \in \mathbb{N} \right\} \cup \{0\}$$

F_n is not the set of the available floating-point numbers on an existing system. It is an "ideal" system, with no overflows or underflows. We will show results in F_n. These results will remain true in an actual systems that implement the IEEE standard, provided that no overflows or underflows occur. The *mantissa* of a nonzero element $M \times 2^E$ of F_n is the number $m(x) = M/2^{n-1}$.

The result of an arithmetic operation whose input values belong to F_n may not belong to F_n (as a matter of fact, in general it does not). Hence that result must be *rounded*. The standard defines 4 different rounding modes:

- rounding towards $+\infty$, or upwards: $\circ_u(x)$ is the smallest element of F_n that is greater than or equal to x;

- rounding towards $-\infty$, or downwards: $\circ_d(x)$ is the largest element of F_n that is less than or equal to x;
- rounding towards 0: $\circ_z(x)$ is equal to $\circ_u(x)$ if $x < 0$, and to $\circ_d(x)$ otherwise;
- rounding to the nearest even: $\circ_n(x)$ is the element of F_n that is closest to x. If x is exactly halfway between two elements of F_n, $\circ_n(x)$ is the one for which M is an even number.

The first three rounding modes are called *directed* rounding modes.

The standard requires that the user should be able to choose one rounding mode among these ones, called the *active rounding mode*. After that, when performing one of the 4 arithmetic operations, or when computing square roots, the obtained result should be equal to the rounding of the exact result.

We will denote these "correctly rounded" operations with a circle and a letter indicating the rounding mode. For instance, when a and b belong to F_n, $a \oplus_u b = \circ_u(a + b)$, whereas $a \oslash_n b = \circ_n(a/b)$.

For $a \in F_n$, we define a^+ as its successor in F_n, that is, $a^+ = \min\{b \in F_n, b > a\}$, and $ulp(a)$ as $|a|^+ - |a|$. If a is not an element of F_n, we define $ulp(a)$ as $\circ_u(a) - \circ_d(a)$. The name *ulp* is an acronym for *unit in the last place*. When $x \in F_n$, $ulp(x)$ is the weight of the last mantissa bit of x. We also define a^- as the predecessor of a.

1 Floating-point reciprocals

Given a rounding mode t, we want to investigate whether an element $x \in F_n$ has an FP-reciprocal (Floating-Point reciprocal) for $\otimes_t$, that is, whether there exists $z \in F_n$ such that $x \otimes_t z = \circ_t(xz) = 1$. From the obvious properties:

$$\begin{array}{rcc} x \in F_n, k \in \mathbb{Z} & \Rightarrow & x2^k \in F_n \\ \lambda \in \mathbb{R}, k \in \mathbb{Z} & \Rightarrow & \circ_t(\lambda 2^k) = 2^k \circ {}_t(\lambda) \\ x, z \in F_n, k \in \mathbb{Z}, x \otimes_t z = 1 & \Rightarrow & (x2^k) \otimes_t (z2^{-k}) = 1 \end{array}$$

we can assume that $1 \leq x < 2$ (that is, it suffices to focus on reciprocals of the mantissas of the elements of F_n). Before looking for FP-reciprocals, one can try to investigate whether the "true" reciprocal of a given element of F_n may belong to F_n (or some other set F_q). The answer, given by the following lemma, is quite straightforward.

Lemma 1 *Let $x \in F_n$. There exists $q \in \mathbb{N}, q \geq 1$ such that $1/x$ belongs to F_q if and only if x is a power of 2.*

Proof. The above remarks show that it suffices to assume that $1 \leq x < 2$. Define $y = 1/x$ and assume $y \in F_q$. This gives:

- $X = 2^{n-1}x \in \mathbb{N}$;
- $Y = 2^p y \in \mathbb{N}$ (since $1/2 < y \leq 1$).

The integer XY is equal to 2^{n+p-1}. Therefore, in the prime number decomposition of X and Y, 2 is the only prime number that can appear. □

1.1 Directed rounding modes

1.1.1 Unicity of FP-reciprocals

In general, an element of F_n may have more than one FP-reciprocal. Consider $x = 27/16$, and the two values $z_1 = 19/32$ and $z_2 = 5/8$. All these values belong to F_5, and $x \otimes_d z_1 = x \otimes_d z_2 = 1$.

Property 1 *For any directed rounding mode, an element of F_n has at most 2 reciprocals. Assuming rounding towards $+\infty$ ($\otimes_u$), a nonnegative number has at most one FP-reciprocal.*

Proof.

Assume $x \in F_n$, $1 < x < 2$ (the case $x = 1$ is obvious, since x is its own unique FP-reciprocal in any rounding mode). Define $y = 1/x$, and

$$\begin{aligned} z_1 &= \circ_d(y) \\ z_2 &= \circ_u(y). \end{aligned}$$

We have:

$$\begin{array}{lcccl} 1/2 & < & y & < & 1 \\ 1/2 & \leq & z_1 & \leq & 1 - 2^{-n} \\ 1/2 + 2^{-n} & \leq & z_2 & \leq & 1 \end{array}$$

Let us first assume rounding towards 0, or downwards. It is worth noticing that, since $x \neq 1$, $z_1 < y$. Hence for any $z \in F_n$ that is less than or equal to z_1, $z \otimes_d x = \circ_d(z \times x) \leq z \times x < xy = 1$. Now, consider the case $z \geq z_2^{++}$. The number z_2^{++} is greater than or equal to $z_2 + 2^{-n+1}$ (it can be larger if $z_2 = 1$ or $z_2^+ = 1$). Therefore: $xz \geq xz_2 + x2^{-n+1} \geq xy + 2^{-n+1} = 1 + ulp(1) \in F_n$. Hence $x \otimes_d z = \circ_d(xz) \geq 1 + 2^{-n+1}$.

Now, let us assume rounding towards $+\infty$, or upwards. Quite obviously (same reasoning as for z_1 and $\circ_d$), for any $z \in F_n$ that is greater than or equal to z_2, $z \otimes_u z > 1$. Now, consider a number $z \in F_n$ that is less than or equal to z_1^-. Two cases may occur:

- if $z_1 > 1/2$, then $z_1^- = z_1 - 2^{-n}$. In such a case, $xz \leq xz_1 - x2^{-n} \leq xy - 2^{-n} = 1 - 2^{-n} \in F_n$. Hence, $x \otimes z \leq 1 - 2^{-n}$;
- if $z_1 = 1/2$ then, since $xz_1 \in F_n$, we have $x \otimes_u z_1 = xz_1 < xy$, since we have assumed $y \neq 1/2$;

Hence, the only element of F_n that may be an FP-reciprocal of x is z_1. □

1.1.2 Existence of FP-reciprocals

Property 2 *Every nonnegative element of F_n has a FP-reciprocal in F_n for $\otimes_u$.*

An immediate consequence of this property is that every nonpositive element of F_n has a FP-reciprocal in F_n for $\otimes_d$.

Proof. Let $x \in F_n, x > 0$. We have to find $z \in F_n$ such that $x \otimes_d z = 1$. We assume $1 \leq x \leq 2$. Define $y = 1/x$ (y is a real number, it does not necessarily belong to F_n), and consider $z = \circ_u(y)$. From $1/2 < y \leq 1$, since the roundings are monotonic functions, and since $1/2$ and 1 belong to F_n (and therefore are equal to their own roundings, for any of the 4 rounding modes), we deduce $1/2 < z \leq 1$. Two cases may occur:

- if $z = 1$, this means that $1 - 2^{-n} < y \leq 1$. Hence, $1 \leq x < 1 + 2^{-n} + 2^{-2n} + 2^{-3n} + \ldots \leq 1 + 2^{-n+1} = 1 + ulp(1) = 1^+$. From $x \in F_n$ and $1 \leq x < 1^+$, we deduce $x = 1$. Therefore $x \otimes_d z = \circ_d(1 \times 1) = 1$;
- if $z < 1$, then the binary representation of z has the form $0.z_1 z_2 \ldots z_n$, and we have:

$$y \leq z < y + 2^{-n}.$$

Thus

$$1 \leq xz < 1 + x2^{-n} < 1 + 2^{-n+1}.$$

This implies

$$\circ_d(1) = 1 \leq x \otimes_d z < \circ_d(1 + 2^{-n+1}) = 1^+,$$

therefore $x \otimes_d z = 1$.

□

1.2 Rounding to the nearest

1.2.1 Unicity of FP-reciprocals

In rounding to the nearest mode, a number may have more than one FP-reciprocal. Consider for instance $x = 3/2$, as an element of F_5. The two following elements of F_5, $z_1 = 21/32$ and $z_2 = 11/16$ satisfy $x \otimes_n z_1 = x \otimes z_2 = 1$. This is the maximum number of FP-reciprocals a number can have.

Property 3 *In rounding to the nearest mode, the only numbers that can be FP-reciprocals of $x \in F_n$ are $\circ_d(1/x)$ and $\circ_u(1/x)$.*

The proof is ommitted, since it is very similar to the proof of the corresponding property for the directed roundings.

1.2.2 Existence of FP-reciprocals

It is worth noticing that some elements of F_n do not have an FP-reciprocal for $\otimes_n$. An exhaustive test shows that if $n \leq 5$, all elements of F_n have an FP-reciprocal. The only element of F_6 between 1 and 2 with no FP-reciprocal is $29/16 = 1.11010$. Hence all elements of F_6 with no reciprocals have the form $29 \times 2^k, k \in \mathbb{Z}$. The only element of F_7 between 1 and 2 with no FP-reciprocal is $59/32$. Table 1 gives the number $\gamma(n)$ of elements of F_n between 1 and 2 with no FP-reciprocal for small values of n.

Although the problem seems more complicated than for directed rounding modes, we can anyway give a conjecture and a few results.

Conjecture 1 *The proportion $\gamma(n)/2^{n-1}$ of elements of F_n without a FP-reciprocal converges toward $\frac{1}{2} - \frac{3}{2}\log(4/3) = 0.06847689\ldots$*

n	$\gamma(n)$	$\gamma(n)/2^{n-1}$
$n \leq 5$	0	0
6	1	0.03125
7	1	0.015625
8	6	0.046875
9	12	0.046875
10	28	0.0546875
11	55	0.0537109375
12	140	0.068359375
13	284	0.0693359375
14	551	0.06726074219...
15	1074	0.06555175781...
16	2182	0.06658935547...
17	4441	0.06776428223...
18	8849	0.06751251221...
19	17933	0.06840896606...
20	35682	0.06805801391...
21	71263	0.06796169281...
22	143467	0.06841039658...
23	286165	0.06822705269...

Table 1: *Number $\gamma(n)$ of elements of F_n between 1 and 2 with no FP-reciprocal (for $\otimes_n$), for small values of n. We also give the proportion $\gamma(n)/2^{n-1}$ of elements of F_n without a FP-reciprocal. We conjecture that this proportion converges toward $\frac{1}{2} - \frac{3}{2}\log(4/3) = 0.06847689\ldots$*

The idea behind the conjecture is the following: saying that $y \in F_n, 1 \leq y < 2$ has no reciprocal for $\otimes_n$ is equivalent[1] to saying that there is no $z \in F_n$ such that $yz \in [1 - 2^{-n-1}, 1 + 2^{-n}]$. This means, if we define integers $Y = y2^{n-1}$ and $Z = z2^n$ that YZ is not in the interval $[2^{2n-1} - 2^{n-2}, 2^{2n-1} + 2^{n-1}]$, i.e., that $2^{2n-1} + 2^{n-1} = YZ + \rho$, where ρ is not in $[0, 2^{n-2} + 2^{n-1}]$. A value $y \in F_n$ without a reciprocal therefore corresponds to an integer $Y \in [2^{n-1}, 2^n]$ such that $2^{2n-1} + 2^{n-1}$ modulo Y is larger than $3 \times 2^{n-2}$. There is obviously no such Y that is less than $3 \times 2^{n-2}$. For larger values, assuming that the "probability"[2] of having $2^{2n-1} + 2^{n-1} \mathrm{mod} Y = k$ is $1/Y$, we can approximate the number of values without reciprocal by:

$$\begin{array}{rl} & \sum_{k=2^{n-1}+2^{n-2}}^{2^n} \frac{Y - 3.2^{n-2}}{Y} \\ = & 2^{n-2} + 1 - 3.2^{n-2} \sum_{k=3.2^{n-2}}^{2^n} 1/Y. \end{array}$$

By approximating the last sum by $\log(2^n) - \log(3.2^{n-2}) = 2\log(2) - \log(3)$, we get the conjecture. □

Property 4 *Every element of F_n whose absolute value of the mantissa is between 1 and 3/2 has a FP-reciprocal in F_n for $\otimes_n$.*

This property is immediately deducible from the sentence "There is obviously no such Y that is less than $3 \times 2^{n-2}$" in the justification of Conjecture 1.

2 Division and reciprocation algorithms

We aim at being able to retreive correctly rounded quotients from approximations of them (obtained for instance using Newton-Raphson or Goldschmidt iterations). We assume that the "fused multiply-and-accumulate", MAC function is available with correct rounding, i.e. that we are able to compute, from three elements a, b and c of F_n, the number $\circ(ab + c)$, where $\circ$ is any of the four rounding modes, the following results are presented in [4].

Theorem 1 (Cornea-Hasegan, Golliver and Markstein, 1999) *Let $x \in F_n$, let $z \in \{\circ_d(1/x), \circ_u(1/x)\}$. We assume that x is not a power of 2. The following calculations (requiring fused MACs only)*

$$\begin{array}{rcl} \epsilon & = & \circ_n(1 - xz) \\ z' & = & \circ_n(z + \epsilon z) \end{array}$$

gives a value z' equal to $\circ_n(1/x)$.

Theorem 2 (Cornea-Hasegan, Golliver and Markstein, 1999) *Let $x, y \in F_n$, let $z = \circ_n(1/x)$, let $q \in \{\circ_d(y/x), \circ_u(y/x)\}$. The following calculations (requiring fused MACs only)*

$$\begin{array}{rcl} r & = & \circ_n(y - xq) \\ q^* & = & \circ_n(q + rz) \end{array}$$

gives a value q^* equal to $\circ_n(y/x)$.

[1] By neglecting the case when a value is exactly the middle of two consecutive elements of F_n, but we can easily show that this case never occurs.

[2] Of course, there is a serious lack of rigor here. This is not a proof, just a quick justification of the conjecture.

These two theorems allow to get correctly rounded quotients from the result of a Newton-Raphson iteration that approximates $1/x$. Some other iterations (such as the Goldschmidt iteration) directly compute a quotient a/b, without the preliminary computation of a reciprocal.

Kahan [6] explains that the fused MAC allows to compute remainders exactly. Let us show how it works. Let $a, b, q \in F_n$, such that

$$q \in \{\circ_d(a/b), \circ_u(a/b)\}\,.$$

Without loss of generality, we assume

$$\begin{aligned} 1 \leq a < 2 \\ 1 \leq b < 2 \end{aligned}$$

Define $q^* = \circ_n(a/b)$.

Property 5 *q^* can be computed as follows.*

1. *compute $r = \circ_n(a - bq)$. Define an integer*

$$K = \begin{cases} n+1 & \text{if } q \leq 1 \\ n & \text{if } q > 1 \end{cases}$$

2. *get*

$$q^* = \begin{cases} q^- & \text{if} \quad r < -2^{-K}b \\ q^+ & \text{if} \quad r > 2^{-K}b \\ q & \text{otherwise} \end{cases}$$

Proof

It suffices to notice that r is a multiple of 2^{-n-K+2} that is less than $2^{-K+1}b$. This suffices to show that $r \in F_n$. Hence, it is computed exactly.

Conclusion

We have given some examples of properties that one can show using the correct rounding property of good floating point arithmetic implementations. Such properties allow to derive algorithms and proofs.

References

[1] American National Standards Institute and Institute of Electrical and Electronic Engineers. IEEE standard for binary floating-point arithmetic. *ANSI/IEEE Standard, Std 754-1985,* New York, 1985.

[2] W. J. Cody. A proposed radix and word length independent standard for floating-point arithmetic. *ACM SIGNUM Newsletter*, 20:37–51, January 1985.

[3] W. J. Cody, J. T. Coonen, D. M. Gay, K. Hanson, D. Hough, W. Kahan, R. Karpinski, J. Palmer, F. N. Ris, and D. Stevenson. A proposed radix-and-word-length-independent standard for floating-point arithmetic. *IEEE MICRO*, 4(4):86–100, August 1984.

[4] M. A. Cornea-Hasegan, R. A. Golliver, and P. Markstein. Correctness proofs outline for newton-raphson based floating-point divide and square root algorithms. In I. Koren and P. Kornerup, editors, *Proceedings of the 14th IEEE Symposium on Computer Arithmetic (Arith-14, Adelaide, Australia, April 1999)*, pages 96–105. IEEE Computer Society Press, Los Alamitos, CA, 1999.

[5] American National Standards Institute, Institute of Electrical, and Electronic Engineers. Ieee standard for radix independent floating-point arithmetic. *ANSI/IEEE Standard, Std 854-1987, New York*, 1987.

[6] W. Kahan. Lecture notes on the status of IEEE-754. Postscript file accessible electronically through the Internet at the address http://http.cs.berkeley.edu/∼wkahan/ieee754status/ieee754.ps, 1996.

[7] D. M. Priest. Algorithms for arbitrary precision floating point arithmetic. In P. Kornerup and D. W. Matula, editors, *Proceedings of the 10th IEEE Symposium on Computer Arithmetic (Arith-10)*, pages 132–144, Grenoble, France, June 1991. IEEE Computer Society Press, Los Alamitos, CA.