



HAL
open science

On the definition of $\text{ulp}(x)$

Jean-Michel Muller

► **To cite this version:**

Jean-Michel Muller. On the definition of $\text{ulp}(x)$. [Research Report] RR-5504, LIP RR-2005-09, INRIA, LIP. 2005, pp.16. inria-00070503

HAL Id: inria-00070503

<https://inria.hal.science/inria-00070503v1>

Submitted on 19 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

On the definition of $ulp(x)$

Jean-Michel Muller

N° 5504

February 2005

Thème SYM

A large blue rectangle occupies the lower half of the page. Overlaid on it is a large, light grey 'R' on the left. To the right of the 'R', the words 'Rapport de recherche' are written in a white, serif font. A horizontal grey brushstroke is positioned below the text.

*Rapport
de recherche*



On the definition of $\text{ulp}(x)$

Jean-Michel Muller

Thème SYM — Systèmes symboliques
Projet Arénaire

Rapport de recherche n° 5504 — February 2005 — 16 pages

Abstract: Function ulp (acronym for *unit in the last place*) is frequently used for expressing errors in floating-point computations. We present several previously suggested definitions of that function, and analyse some of their properties.

Key-words: Computer arithmetic, floating-point arithmetic, unit in the last place, ULP

This text is also available as a research report of the Laboratoire de l'Informatique du Parallélisme
<http://www.ens-lyon.fr/LIP>.

A propos de la définition de $\text{ulp}(x)$

Résumé : La fonction ulp (acronyme pour *unit in the last place*, c'est-à-dire “poids du dernier chiffre”) est fréquemment utilisée pour exprimer des erreurs en arithmétique virgule flottante. Nous présentons plusieurs définitions précédemment suggérées pour cette fonction, et analysons quelques unes de leurs propriétés.

Mots-clés : Arithmétique des ordinateurs, arithmétique virgule flottante, poids du dernier chiffre

1 Introduction

The term ulp (acronym for *unit in the last place*) was coined by W. Kahan in 1960. The original definition was [5]:

$\text{ulp}(x)$ is the gap between the two floating-point numbers nearest x , even if x is one of them.

As told by Kahan [5], the adoption of the IEEE-754 standard for floating-point arithmetic has made infinities and NaNs ubiquitous, and that must be taken into account in the definition of $\text{ulp}(x)$. Kahan now suggests the following definition:

$\text{ulp}(x)$ is the gap between the two *finite* floating-point numbers nearest x , even if x is one of them. (But $\text{ulp}(\text{NaN})$ is NaN.)

Several slightly different definitions of $\text{ulp}(x)$ appear in the literature [3, 4, 6, 8]. In this paper, we remind these various definitions and we analyze some of their properties. Among these properties, some have certainly already been found by other people having dealt with this topic (without, to my knowledge, having been published, except when I give references). And yet, I feel it may be useful to collect them in a paper.

Throughout the paper, we assume a radix- r floating-point (FP for short) arithmetic, with n -digit mantissas¹. If X is an FP number, then X^+ denotes the smallest FP number larger than X and X^- denotes the largest FP number less than X .

A good definition of function ulp :

- should (of course) agree with the “intuitive” notion when x is not in an “ambiguous area” (i.e., x is not near a power of the radix, of larger than the largest representable number, or $\pm\infty$, or zero...);
- should be *useful*: after all, for a binary n -bit format, defining $\text{ulp}(1)$ as 2^{-n} (i.e., $1 - 1^-$) or 2^{-n+1} (i.e., $1^+ - 1$) are equally legitimate from a theoretical point of view. What matters is which choice is helpful (i.e., which choice will preserve in “ambiguous areas” properties that are true when we are far from them);

Let us consider the following common claims. They are true “in general”, but they need some clarification. In the following $\text{RN}(x)$ is x rounded to the nearest (even) floating-point (FP) number, $\text{RD}(x)$ is x rounded towards $-\infty$, $\text{RU}(x)$ is x rounded towards $+\infty$, and $\text{RZ}(x)$ is x rounded towards zero. The uppercase letter X will denote an FP number, whereas x will denote a real number.

¹The possible implicit leading bit of the binary systems is counted in these n digits. For instance, in IEEE-754 double precision arithmetic, n is equal to 53.

Common claim 1

$$X = RN(x) \Rightarrow |x - X| \leq \frac{1}{2} ulp$$

Common claim 2

$$|x - X| < \frac{1}{2} ulp \Rightarrow X = RN(x)$$

Common claim 3

$$|x - X| < 1 ulp \Leftrightarrow X \in \{RD(x), RU(x)\}$$

In these claims, several things are unclear. The first one, of course, is the definition of ulp (especially near the powers of the radix). The second one is whether “ ulp ” means $ulp(x)$ or $ulp(X)$. Of course, in most practical cases, both values will be equal. But in difficult cases (e.g., X is a loose approximation to x , or these values are close to a power of the radix), they may differ.

2 Should we consider $ulp(x)$ or $ulp(X)$?

It should be clear that, for measuring the error of an approximation, the (possibly very loose) approximation should not be used for defining the measure of error: the distance between x (exact value) and X (FP approximation) should be expressed in terms of $ulp(x)$, instead of $ulp(X)$. Just consider the example given in Figure 1: we assume a binary floating-point system, with n -bit mantissas, we consider the real number $x = 1^+ = 1 + 2^{-n+1}$ and two (very poor) approximations $A = 2^- = 2 - 2^{-n+1}$ and $B = 2^+ = 2 + 2^{-n+2}$. A approximates x with error $(2^{n-1} - 2) \approx 2^{n-1} ulp(A)$, whereas B approximates x with error $(2^{n-2} + 1/2) \approx 2^{n-2} ulp(B)$. From these values, one could believe that B is a much better approximation to x than A . And yet, A is closer to x than B . This shows that ulp (approximation) cannot be a sensible unit of measurement of error.

3 Various definitions of function ulp

Definition 1 (Kahan [2, 5]) *KahanUlp(x) is the width of the interval whose endpoints are the two finite representable numbers nearest x (even if x is not contained within that interval).*

Note: in [4], Harrison attributes the previous definition of $ulp(x)$ to me, because I used approximately the same in my book on elementary functions [7] (when writing the book, I was not aware of Kahan’s definition).

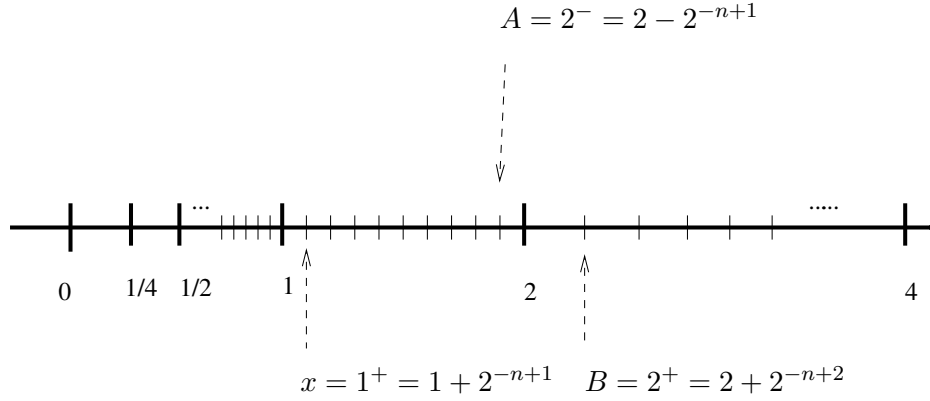


Figure 1: A approximates x with error $(2^{n-1} - 2) \approx 2^{n-1} ulp(A)$, whereas B approximates x with error $(2^{n-2} + 1/2) \approx 2^{n-2} ulp(B)$. From these values, one could believe that B is a much better approximation to x than A . And yet, A is closer to x than B .

Definition 2 (Harrison [4]) *HarrisonUlp*(x) is the distance between the closest straddling points a and b (i.e., those with $a \leq x \leq b$ and $a \neq b$), assuming an unbounded exponent range.

It is worth being noticed that Kahan's and Harrison's definitions coincide on FP numbers. However, for real numbers they may differ near powers of the radix. For instance, in radix 2 with n -bit mantissas, if $1 < x < 1 + 2^{-n-1}$ then $KahanUlp(x) = 2^{-n}$ and $HarrisonUlp(x) = 2^{-n+1}$.

Definition 3 (Goldberg [3]) *GoldbergUlp*(x) is defined as follows. If the FP number $d.dddd \dots d\beta^e$ is used to represent x , it is in error by

$$|d.dddd \dots d - (x/\beta^e)|$$

units in the last place.

This last definition uses the approximation that represents x . To use it, we will assume that the approximation is $RZ(x)$ (to keep the same exponent). We will call the obtained definition the modified *GoldbergUlp*(x).

Overton [8] defines function ulp for FP numbers only. He defines $ulp(X)$, for $X > 0$, as the gap between X and the next larger floating-point number (for $X < 0$, $ulp(X) = ulp(-X)$). His definition coincides with Goldberg's definition on FP numbers. Markstein [6] gives a very similar definition.

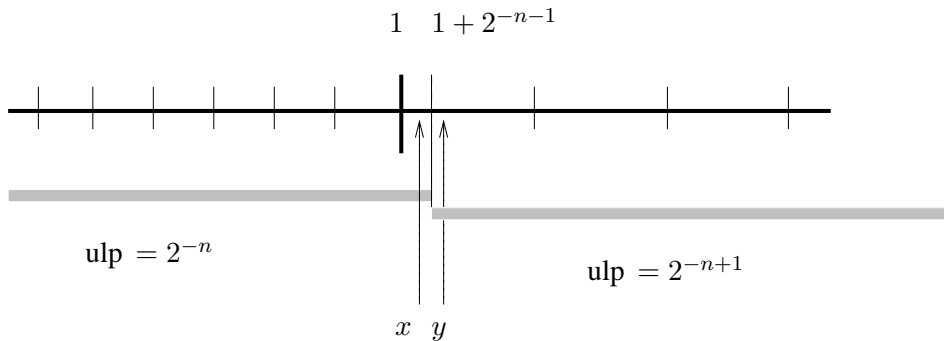


Figure 2: The values of $KahanUlp(x)$ near 1, assuming a binary FP system with n -bit mantissas. Note the strange side effect: 1 seems to be a better approximation to y than to x .

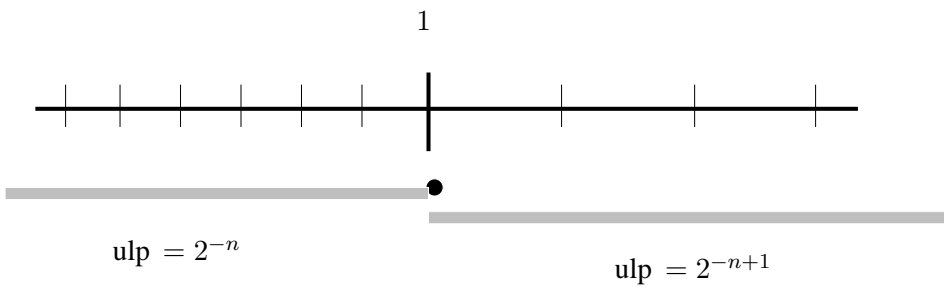


Figure 3: The values of $HarrisonUlp(x)$ near 1, assuming a binary FP system with n -bit mantissas.

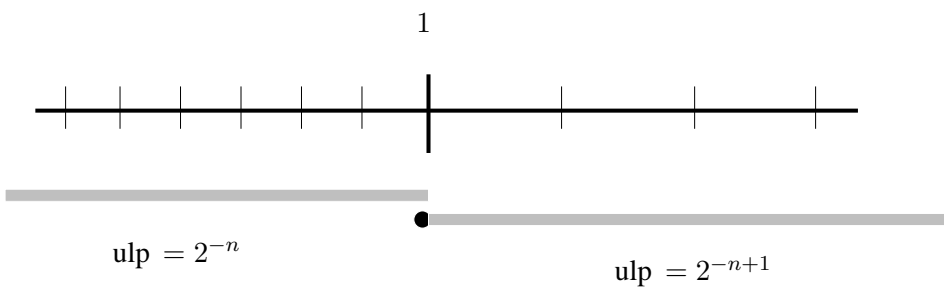


Figure 4: The values of $Modified\ GoldbergUlp(x)$ near 1, assuming a binary FP system with n -bit mantissas. Notice that $Modified\ GoldbergUlp(x)$ and $HarrisonUlp(x)$ only differ when x is a power of the radix.

4 Some properties (assuming unbounded exponents)

4.1 With rounding to nearest

Property 1 *In radix 2,*

$$|X - x| < \frac{1}{2} \text{HarrisonUlp}(x) \Rightarrow X = \text{RN}(x)$$

See Theorem 1 for proof. Property 1 is not true in radices greater than or equal to 3. Figure 5 gives a counter-example in radix 3.

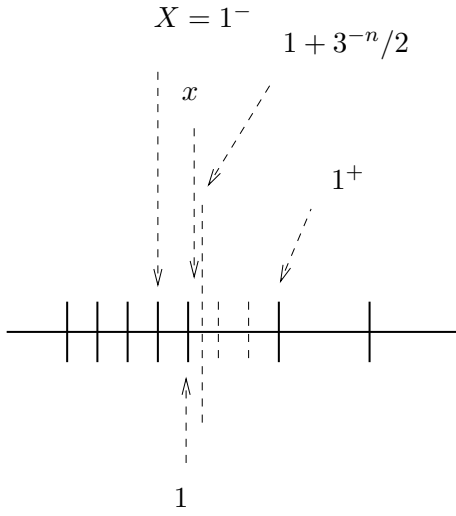


Figure 5: This example shows that Property 1 is not true in radix 3. Here, x satisfies $1 < x < 1 + \frac{1}{2}3^{-n}$ and $X = 1^- = 1 - 3^{-n}$. We have $\text{HarrisonUlp}(x) = 3^{-n+1}$, and $|x - X| < 3^{-n+1}/2$, so that $|x - X| < \frac{1}{2} \text{HarrisonUlp}(x)$. And yet, $X \neq \text{RN}(x)$.

Property 2 *For any radix,*

$$X = \text{RN}(x) \Rightarrow |X - x| \leq \frac{1}{2} \text{HarrisonUlp}(x)$$

See Theorem 2 for proof.

Property 3 *For any radix,*

$$|X - x| < \frac{1}{2} \text{KahanUlp}(x) \Rightarrow X = \text{RN}(x)$$

See Theorem 1 for proof.

Property 4 In radix 2,

$$X = RN(x) \Rightarrow |X - x| \leq \frac{1}{2} \text{KahanUlp}(x)$$

See Theorem 2 for proof. Property 4 is not true in radices greater than or equal to 3. Figure 6 gives a counter-example in radix 3.

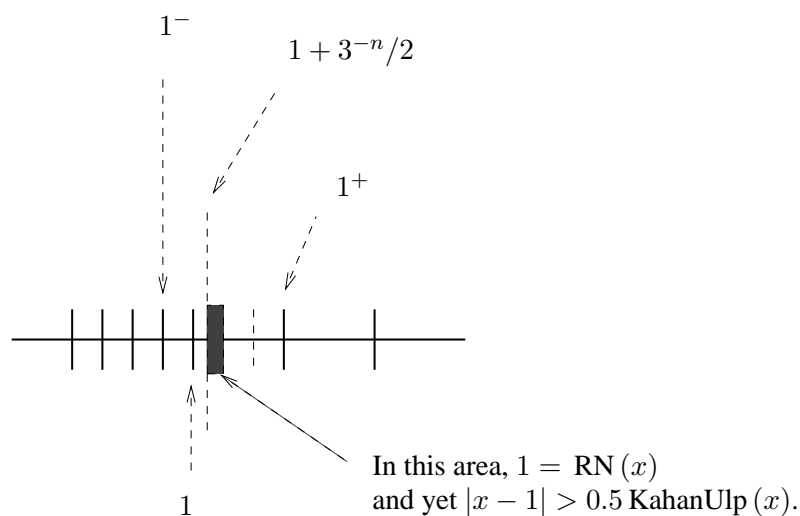


Figure 6: This example shows that Property 4 is not true in radix 3. If $1 + \frac{1}{2}3^{-n} < x < 1 + 3^{-n}$, then $1 = RN(x)$, and yet $|x - 1| > 0.5 \text{KahanUlp}(x)$.

We see that with rounding to nearest in radix 2, both Kahan's and Harrison's definitions preserve the common claims listed above. As we shall see later, the situation is different with directed roundings.

Definition 4 A regular ulp function is such that there exists a value $x_{\text{cut}} \in [1, 1 + r^{-n+1})$ so that

$$\text{ulp}(x) = r^{-n+1+k}$$

if $r^k x_{\text{cut}} < x < r^{k+1} x_{\text{cut}}$.

This does not uniquely define the value of $\text{ulp}(x)$ since there remains an ambiguity at $x = r^k x_{\text{cut}}$. This ambiguity has no importance if $x_{\text{cut}} \neq 1$, but may make a difference if $x_{\text{cut}} = 1$.

For instance, both `HarrisonUlp` and `KahanUlp` are regular ulp functions, with $x_{\text{cut}} = 1$ for `HarrisonUlp` and $x_{\text{cut}} = 1 + \frac{r^{-n}}{2}(r - 1)$ for `KahanUlp`.

Theorem 1 *To have*

$$|X - x| < \frac{1}{2} ulp(x) \Rightarrow X = RN(x)$$

for any real x and FP number X , we need

$$x_{\text{cut}} \geq 1 + r^{-n} \left(\frac{r}{2} - 1 \right). \quad (1)$$

Proof: We only consider the case $1 \leq x < 1^+$ (the other cases are either straightforward, or easily deduced from this one). First, if $x > x_{\text{cut}}$, then $ulp(x) = r^{-n+1}$. In that case, since $1^- = 1 - r^{-n}$ cannot be the FP number that is nearest x (because x is closer to 1 than to 1^-), we must have

$$x - 1^- > \frac{1}{2} ulp(x),$$

i.e.,

$$x > 1 + r^{-n} \left(\frac{r}{2} - 1 \right).$$

This gives the condition of the theorem.

Conversely, if $x_{\text{cut}} \geq 1 + r^{-n} \left(\frac{r}{2} - 1 \right)$ then

- if $1 \leq x < x_{\text{cut}}$ then $ulp(x) = 1 - 1^- = r^{-n}$. Hence, the only values that can be within $\frac{1}{2} ulp(x)$ from x (if any) are 1 and 1^+ , and at most one of these values only can be within $\frac{1}{2} ulp(x)$ from x . If there is one, it will necessary be the FP number that is nearest x ;
- if $x > x_{\text{cut}}$ then $ulp(x) = 1^+ - 1 = r^{-n+1}$. Since (1) implies that $x - 1^- > \frac{1}{2} ulp(x)$, the only values that can be within $\frac{1}{2} ulp(x)$ from x (if any) are 1 and 1^+ , and at most one of these values only can be within less than $\frac{1}{2} ulp(x)$ from x . If there is one, it will necessary be the FP number that is nearest x .

Theorem 2 *To have*

$$X = RN(x) \Rightarrow |X - x| \leq \frac{1}{2} ulp(x)$$

for any real x and FP number X , we need

$$x_{\text{cut}} \leq 1 + \frac{1}{2} r^{-n}. \quad (2)$$

Proof: Again, we only consider the case $1 \leq x < 1^+$ (the other cases are either straightforward, or easily deduced from this one).

If $x_{\text{cut}} > 1 + \frac{1}{2}r^{-n}$ then, for

$$1 + \frac{1}{2}r^{-n} < x < \min\{x_{\text{cut}}, 1 + \frac{1}{2}r^{-n+1}\}$$

we have,

$$\begin{cases} \text{RN}(x) &= 1 \\ \text{ulp}(x) &= r^{-n} \end{cases}$$

hence, we have $1 = \text{RN}(x)$, and yet $|1 - x| > \frac{1}{2} \text{ulp}(x)$. Hence the condition of the theorem.

Conversely, if $x_{\text{cut}} \leq 1 + \frac{1}{2}r^{-n}$, then for $1 \leq x \leq x_{\text{cut}}$, we have both $\text{RN}(x) = 1$ and $|1 - x| \leq \frac{1}{2} \text{ulp}(x)$, and for $x_{\text{cut}} < x < 1^+$, we have $\text{ulp}(x) = r^{-n+1} = 1^+ - 1$, so $\text{RN}(x)$ is the value X in $\{1, 1^+\}$ that is nearest x , and $|X - x|$ is obviously less than or equal to $(1^+ - 1)/2 = \frac{1}{2} \text{ulp}(x)$.

Theorem 3 *If the radix r is greater than or equal to 4, there is no regular ulp function that satisfies both*

$$|X - x| < \frac{1}{2} \text{ulp}(x) \Rightarrow X = \text{RN}(x)$$

and

$$X = \text{RN}(x) \Rightarrow |X - x| \leq \frac{1}{2} \text{ulp}(x).$$

Theorem 3 implies that for $r \geq 4$ (which means, in practice, for $r = 10$, since radices different from 2 and 10 seem no longer used) we have to choose between both properties: they will be true “in general”, but one of them will be wrong when x is close to a power of r . Theorem 3 is an immediate consequence of Theorems 1 and 2 (conditions (1) and (2) become incompatible for $r \geq 4$). For $r = 3$, the only allowable value of x_{cut} is $1 + 3^{-n}/2$. For $r = 2$, $x_{\text{cut}} \in [1, (1 + 1^+)/2]$.

4.2 With directed roundings

Property 5 *For any value of the radix r ,*

$$X \in \{RD(x), RU(x)\} \Rightarrow |X - x| < 1 \text{HarrisonUlp}(x)$$

But now the converse is not true. There are values X and x for which $|X - x| < 1 \text{ HarrisonUlp}(x)$, and yet X is not in $\{\text{RD}(x), \text{RU}(x)\}$ (consider the case x slightly above 1 and X equal to 1^- , the FP predecessor of 1).

With $\text{KahanUlp}(x)$, also, there are values X and x for which $|X - x| < 1 \text{ HarrisonUlp}(x)$, and yet X is not in $\{\text{RD}(x), \text{RU}(x)\}$ (consider, in radix 2 with n -bit mantissas, the case $X = 1 - 2^{-n}$ and x between $1 + 2^{-n-1}$ and $1 + 2^{-n}$).

With $\text{KahanUlp}(x)$, there is no equivalent of property 5. As noticed by Harrison [4], we can have $X \in \{\text{RD}(x), \text{RU}(x)\}$, and $|X - x|$ significantly larger than $1 \text{ KahanUlp}(x)$ (it can be arbitrarily close, without being equal, to $r \text{ KahanUlp}(x)$). Consider the radix-2 case depicted by Figure 7.

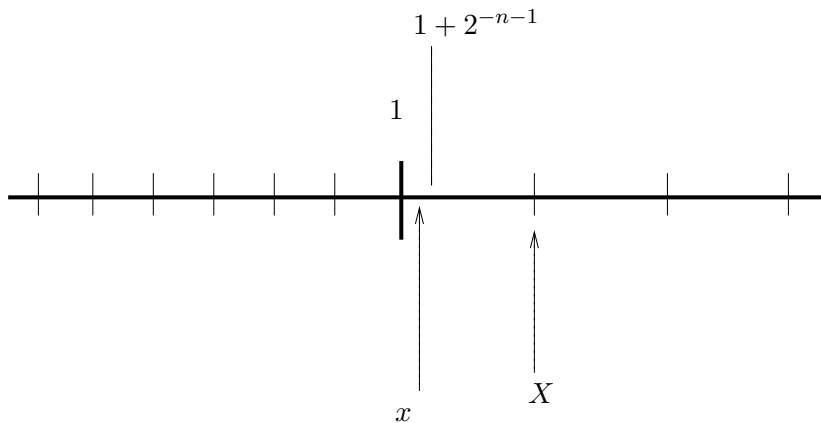


Figure 7: We assume radix 2 and n -bit mantissas. X is equal to $\text{RU}(x)$, and yet $|X - x|$ is very close to $2 \text{ KahanUlp}(x)$ [4].

4.3 If anyway one decides to use $\text{ulp}(X)$

Although we have indicated in Section 2 that using $\text{ulp}(x)$ as the measure of error seems much preferable, one may, for some application, find a good reason for using $\text{ulp}(X)$. In such a case, we list the obtained properties below.

Property 6 *Assuming unbounded exponents, we find, for any value of the radix r :*

$$\begin{aligned}
& |X - x| < \frac{1}{2} \mathit{HarrisonUlp}(X) \Rightarrow X = \mathit{RN}(x) \\
& X = \mathit{RN}(x) \text{ does not imply } |X - x| \leq \frac{1}{2} \mathit{HarrisonUlp}(X) \\
& |X - x| < \mathit{HarrisonUlp}(X) \Rightarrow X \in \{\mathit{RD}(x), \mathit{RU}(x)\} \\
& X \in \{\mathit{RD}(x), \mathit{RU}(x)\} \text{ does not imply } |X - x| \leq \mathit{HarrisonUlp}(X) \\
& |X - x| < \frac{1}{2} \mathit{KahanUlp}(X) \Rightarrow X = \mathit{RN}(x) \\
& X = \mathit{RN}(x) \text{ does not imply } |X - x| \leq \frac{1}{2} \mathit{KahanUlp}(X) \\
& |X - x| < \mathit{KahanUlp}(X) \Rightarrow X \in \{\mathit{RD}(x), \mathit{RU}(x)\} \\
& X \in \{\mathit{RD}(x), \mathit{RU}(x)\} \text{ does not imply } |X - x| \leq \mathit{KahanUlp}(X) \\
& X = \mathit{RN}(x) \Rightarrow |X - x| \leq \frac{1}{2} \mathit{GoldbergUlp}(X) \\
& |X - x| < \frac{1}{2} \mathit{GoldbergUlp}(X) \text{ does not imply } X = \mathit{RN}(x) \\
& X \in \{\mathit{RD}(x), \mathit{RU}(x)\} \Rightarrow |X - x| \leq \mathit{GoldbergUlp}(X) \\
& |X - x| < \mathit{GoldbergUlp}(X) \text{ does not imply } X \in \{\mathit{RD}(x), \mathit{RU}(x)\}
\end{aligned}$$

In that case, Kahan's and Harrison's definitions satisfy the same properties, which is not surprising since they coincide on FP numbers.

5 Properties near infinity

Kahan's definition is the only one that clearly defines function ulp for big numbers. Define L as the largest finite FP number, and L^- as its predecessor. If x is larger than L , then it is clear from definition 1 that

$$\mathit{KahanUlp}(x) = L - L^-.$$

From this, it is clear that

$$|X - x| < \frac{1}{2} \mathit{KahanUlp}(x) \Rightarrow X = \mathit{RN}(x)$$

So, property 3 is always true (there is no need to assume unbounded exponents, as in the previous section).

Interestingly enough, with IEEE-754 FP (binary) numbers, the converse holds. This is due to a feature of the IEEE-754 Standard [1] (which by the way makes $\mathit{RN}(x)$ quite

different from what one would expect from the term “rounding to *nearest*”). The standard says that an infinitely precise result with magnitude at least

$$2^{\text{emax}} (2 - 2^{-n})$$

shall round to ∞ with no change in sign. With that convention, if X is *finite*,

$$X = \text{RN}(x) \Rightarrow |X - x| \leq \frac{1}{2} \text{KahanUlp}(x),$$

i.e., Property 4 remains true for big numbers.

The intuitive generalization of Harrison’s definition (for big numbers, the straddling points would be L and $+\infty$), would give $+\infty$ for $\text{ulp}(x)$ when $X > L$. This would make sense, but would be useless: any FP number would be within $1/2 \text{ulp}$ from such an x .

6 Conclusion

It appears that a definition that would preserve most properties would be

$$\text{ulp}(x) = \begin{cases} \text{HarrisonUlp}(x) & \text{if } |x| \leq L \\ \text{KahanUlp}(x) = L - L^- & \text{otherwise,} \end{cases}$$

which could be given as follows:

Definition 5 *If x is a real number that lies between two finite consecutive FP numbers a and b , without being equal to one of them, then $\text{ulp}(x) = |b - a|$, otherwise $\text{ulp}(x)$ is the distance between the two finite FP numbers nearest x . Moreover, $\text{ulp}(\text{NaN})$ is NaN .*

Acknowledgement

In November 2004, I had the opportunity to discuss these topics (as well as other aspects of floating-point arithmetic) with Professor Kahan. These were enlightening discussions. Although we may occasionally disagree on minor issues, we share the same feeling that the big improvements brought to numerical computing by the IEEE-754 and 854 standards for floating-point arithmetic are endangered: we must explain to computer architects, compiler designers and numerical application programmers that some features of the standards that sometimes seem arcane or that seem to hinder performance may be crucial when reliability and/or portability are at stake.

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] 754 R committee. DRAFT standard for floating-point arithmetic p754 d0.6.5. October 2004.
- [3] D. Goldberg. What every computer scientist should know about floating-point arithmetic. *ACM Computing Surveys*, 23(1):5–47, March 1991.
- [4] J. Harrison. A machine-checked theory of floating-point arithmetic. In Y. Bertot, G. Dowek, A. Hirschowitz, C. Paulin, and L. Théry, editors, *Theorem Proving in Higher Order Logics: 12th International Conference, TPHOLS'99*, volume 1690 of *Lecture Notes in Computer Science*, pages 113–130, Nice, France, September 1999. Springer-Verlag.
- [5] W. Kahan. A logarithm too clever by half. Available at <http://http.cs.berkeley.edu/~wkahan/LOG10HAF.TXT>, 2004.
- [6] P. Markstein. *IA-64 and Elementary Functions : Speed and Precision*. Hewlett-Packard Professional Books. Prentice Hall, 2000. ISBN: 0130183482.
- [7] J.M. Muller. *Elementary Functions, Algorithms and Implementation*. Birkhauser, Boston, 1997.
- [8] M. A. Overton. *Numerical Computing with IEEE Floating-Point Arithmetic*. SIAM, 2001.

Appendix: Maple programs that compute $\text{ulp}(x)$ in double precision

The following two Maple programs compute $\text{KahanUlp}(t)$ and $\text{ulp}(t)$ as suggested in Definition 5 for any real number t , assuming that the used floating-point format is the double precision format of the IEEE-754 standard (i.e., $r = 2$ and $n = 53$).

```
KahanUlp := proc(t);
x := abs(t);
if x < 2^(-1021) then res := 2^(-1074)
```

```

else if x > (1-2^(-53))*2^(1024) then res := 2^971
else
  powermin := 2^(-1021); expmin := -1021;
  powermax := 2^1024; expmax := 1024;
# x is between powermin = 2^expmin and powermax = 2^expmax
  while (expmax-expmin > 1) do
    expmiddle := round((expmax+expmin)/2);
    powermiddle := 2^expmiddle;
    if x >= powermiddle then
      powermin := powermiddle;
      expmin := expmiddle
    else
      powermax := powermiddle;
      expmax := expmiddle
    fi;
  od;
# now, expmax - expmin = 1
# and powermin <= x < powermax
  if x/powermin <= 1+2^(-54) then res := 2^(expmin-53)
  else res := 2^(expmin-52)
  fi;
fi;
res;
end;

SuggestedUlp := proc(t);
x := abs(t);
if x < 2^(-1021) then res := 2^(-1074)
else if x > (1-2^(-53))*2^(1024) then res := 2^971
else
  powermin := 2^(-1021); expmin := -1021;
  powermax := 2^1024; expmax := 1024;
# x is between powermin = 2^expmin and powermax = 2^expmax
  while (expmax-expmin > 1) do
    expmiddle := round((expmax+expmin)/2);
    powermiddle := 2^expmiddle;
    if x >= powermiddle then
      powermin := powermiddle;
      expmin := expmiddle
    else
      powermax := powermiddle;
      expmax := expmiddle
    fi;
  od;
fi;
res;
end;

```

```
        fi;
      od;
# now, expmax - expmin = 1
# and powermin <= x < powermax
  if x = powermin then res := 2^(expmin-53)
    else res := 2^(expmin-52)
  fi;
fi;
fi;
res;
end;
```



Unité de recherche INRIA Rhône-Alpes
655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)

Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)

Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)

Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)

<http://www.inria.fr>

ISSN 0249-6399