



When Is Double Rounding Innocuous?

Samuel A. Figueroa
Taylor University
500 West Reade Ave.
Upland, IN 46989
smfiguero@tayloru.edu

July 27, 1995

1 Introduction

Double rounding is the phenomenon that occurs when the result of an operation is rounded to fit some intermediate destination, and then again when delivered to its final destination. This can be a common occurrence when using some floating-point arithmetic engines which lack single precision registers: results of operations are typically rounded to fit in a register, whose width may be double precision or wider, before being stored in some memory location possibly in a format narrower than that of the registers. Examples of such floating-point arithmetic engines include Intel's x87 series and IBM's POWER architecture*. (Implementations of the latter are found in some IBM workstations.)

Double rounding can yield different results than if results were rounded only once before being delivered to their final destinations. [5] gives the example of computing 1.9×0.66 using decimal arithmetic. The exact product is 1.254, but if this product were rounded first to three significant digits and then to two, the final result would be 1.2 instead of 1.3 (the latter being closer to the exact product), if one were to round to the nearest number as specified in [4][†].

Double rounding can cause a headache in some situations. For example, consider the problem of converting a rational number, whose numerator and denominator are both double precision floating-point numbers, to the nearest double precision floating-point number. Although it is beyond the scope of this paper to present an algorithm that works regardless of whether double

rounding were to occur, and that can be coded portably in a high-level language, this author's thesis (which has not yet been finished) discusses how this can be accomplished.

If double rounding can be disastrous, one might ask oneself, "Is double rounding ever harmless?" This paper answers this question, at least from the context of how one can emulate single precision (binary) floating-point arithmetic when only double precision floating-point arithmetic is available.

In fact, this is the same context as that in [5], which unfortunately contains the following claim:

If x and y have p -bit significands, and $x + y$ is computed exactly and then rounded to q places, a second rounding to p places will not change the answer if $p \leq (q - 1)/2$. This is true not only for addition, but also for multiplication, division, and square root.

(The same claim also appears in the draft of the second edition of [5].) In addition to showing that this statement is not quite true for square root, this paper contains proofs that this statement is true for the other arithmetic operations.

Most of the information that follows apparently appears in some lecture notes from a course Prof. W. Kahan gave in 1988 [1,2] (and to which this author does not have access). However, one reason for submitting this paper for publication is to make this information more accessible.

*Even if an architecture lacks single precision registers, results of operations will not necessarily suffer double rounding. For example, in Motorola's PowerPC and DEC's Alpha architectures, results of single precision instructions are rounded to single precision (rather than double precision), even though they are stored in double precision registers. (Actually, as will be shown in this paper, in cases such as these, it would make no difference if double rounding were to occur in the process of computing the results of single precision instructions.)

[†][3], which is better known than [4], is almost a subset of the latter.

2 Double rounding can be innocuous

[3] defines four different rounding modes: round to nearest or even, round toward zero, round toward positive infinity, and round toward negative infinity. When rounding to nearest or even, if there are two consecutive floating-point numbers equally near to the in-

finitely precise result, the floating-point number whose significand has a zero as its least significant bit is chosen as the correctly rounded result. This rounding algorithm is also known as *unbiased rounding*. (*Biased rounding* is similar, except that in the situation above, the floating-point number with the larger magnitude is chosen as the correctly rounded result.) If the rounding mode used is not “round to nearest,” it is easy to see that double rounding cannot cause the final result to be different from what would have been obtained with single rounding, as long as single precision floating-point numbers are a subset of double precision floating-point numbers.

Even when rounding to the nearest number, double rounding is of concern only if the significant digits of the infinitely precise result form one of either one or two specific bit patterns. These patterns depend on whether biased or unbiased rounding is used. In the former case, the bit pattern of concern is

$$1d_1d_2\dots d_{p-1}011\dots 11d_{q+1}d_{q+2}\dots,$$

where p and q are the number of digits in the significands of single and double precision numbers, respectively, and each d_n , $n > 0$, is a one or a zero. The reason why this pattern is of concern is that such a number, rounded to q digits, would yield $1d_1d_2\dots d_{p-1}1$. Subsequent rounding of this number to p digits would yield the next largest p -digit number. However, the single precision number closest to the infinitely precise result would simply be $1d_1d_2\dots d_{p-1}$.

If unbiased rounding is used, the bit patterns of concern are

$$1d_1d_2\dots d_{p-2}1011\dots 11d_{q+1}d_{q+2}\dots,$$

which is similar to the pattern of concern for biased rounding, and

$$1d_1d_2\dots d_{p-2}0100\dots 00d_qd_{q+1}\dots,$$

where either d_q is one and d_{q+n} , $n > 0$, are all zeros, or d_q is zero and d_{q+n} , $n > 0$, are not all zeros. The reason why the latter pattern is of concern is that such a number, rounded to q digits, would yield $1d_1d_2\dots d_{p-2}01$. Subsequent rounding of this number to p digits would yield $1d_1d_2\dots d_{p-2}0$. However, the single precision number closest to the infinitely precise result would be the next largest p -digit number: $1d_1d_2\dots d_{p-2}1$.

The rest of this section proves, for the various arithmetic operations (addition, subtraction, multiplication, division, and square root), how many digits the significands of double precision numbers must have in order for double rounding to always yield the same result as would be obtained if rounding to yield a single precision number were to occur just once. These proofs take

into account both unbiased rounding as well as biased rounding. (In some cases, the proofs are slightly different, depending on which of these two rounding methods is used.)

2.1 Addition

Theorem 1 *Let x and y be positive binary floating-point numbers whose significands consist of at most p digits, where $p \geq 2$, and let z be the binary floating-point number that most closely approximates $x + y$, and whose significand consists of at most q digits, where $q > p$. The binary floating-point number that most closely approximates $x + y$ and whose significand consists of at most p digits is the one that most closely approximates z if and only if $q \geq 2p + 1$.*

Proof. Without loss of generality, assume $x \geq y$; otherwise, interchange x and y in the rest of this proof. We will restrict our attention to floating-point numbers x and y such that $1 \leq x + y < 2$, since all pairs of positive binary floating-point numbers can be scaled by powers of two to meet these constraints.

There are potentially two cases in which our hypothesis is not trivially true: if $x + y$ (in infinite precision) were to look something like

$$1.d_1d_2\dots d_{p-2}0100\dots 00d_qd_{q+1}\dots,$$

where either d_q is one and d_{q+n} , $n > 0$, are all zeros, or d_q is zero and d_{q+n} , $n > 0$, are not all zeros; or if $x + y$ (in infinite precision) were to look something like

$$1.d_1d_2\dots d_{p-1}011\dots 11d_{q+1}d_{q+2}\dots$$

In both cases, $y < 2^{p-q}$, since d_{q+n} , $n \geq 0$ are not all zeros. If $q \geq 2p + 1$, then $y < 2^{-p-1}$. Now, in the sum $x + y$ there are either $p + 1$ or $p + 2$ significant digits to the left of the $p + 2$ nd digit to the right of the radix point, the first and last of which are nonzero. Therefore, x cannot be a number with at most p significant digits if $q \geq 2p + 1$ and $x + y$ (in infinite precision) looks something like $1.d_1d_2\dots d_{p-2}0100\dots 00d_qd_{q+1}\dots$ or like $1.d_1d_2\dots d_{p-1}011\dots 11d_{q+1}d_{q+2}\dots$.

If $p < q < 2p + 1$, the theorem is false: Consider the case where $x = 1 + 2^{1-p}$ and $y = 2^{-p}(1 - 2^{-p})$. $x + y$ looks something like

$$1.00\dots 001011\dots 11,$$

where there are $p - 2$ consecutive zeros immediately to the right of the radix point followed by a one, a zero, and p consecutive ones. If $p < q < 2p + 1$, then $z = 1 + 3 \cdot 2^{-p}$, and the p -digit number that most closely approximates z is $1 + 2^{2-p}$, whereas the p -digit number that most closely approximates $x + y$ is x . ■

2.2 Subtraction

Theorem 2 *Let x and y be positive binary floating-point numbers whose significands consist of at most p digits, where $p \geq 2$, such that $x > y$, and let z be the binary floating-point number that most closely approximates $x - y$, and whose significand consists of at most q digits, where $q > p$. The binary floating-point number that most closely approximates $x - y$ and whose significand consists of at most p digits is the one that most closely approximates z if and only if $q \geq 2p + 1$ (using unbiased rounding) or $q \geq 2p$ (using biased rounding).*

Proof. We will restrict our attention to floating-point numbers x and y such that $1 \leq x - y < 2$, since all pairs of positive binary floating-point numbers x and y such that $x > y$ can be scaled by powers of two to meet these constraints.

There are potentially two cases in which our hypothesis is not trivially true: if $x - y$ (in infinite precision) were to look something like

$$1.d_1d_2\dots d_{p-2}0100\dots 00d_qd_{q+1}\dots,$$

where either d_q is one and d_{q+n} , $n > 0$, are all zeros, or d_q is zero and d_{q+n} , $n > 0$, are not all zeros; or if $x - y$ (in infinite precision) were to look something like

$$1.d_1d_2\dots d_{p-1}011\dots 11d_{q+1}d_{q+2}\dots$$

If $x - y$ were to look as in the first case, then $x - y = 2^{2-p}a + 2^{-p} + c$, or

$$x = 2^{2-p}a + 2^{-p} + c + y,$$

where a is an integer such that $2^{p-2} \leq a < 2^{p-1}$, and $0 < c \leq 2^{-q}$. Now, $y < 2^{p-q}$, since d_{q+n} , $n \geq 0$ are not all zeros. If $q \geq 2p + 1$, then $y < 2^{-p-1}$ and $0 < c \leq 2^{-2p-1}$, so $c + y \leq 2^{-p-1}$. The sum $2^{2-p}a + 2^{-p}$ consists of one digit to the left of the radix point and p digits to the right of the radix point, for a total of $p + 1$ significant digits. Adding $c + y$ to this quantity can only increase the number of significant digits. Therefore, x cannot be a number with at most p significant digits if $q \geq 2p + 1$ and $x - y$ (in infinite precision) looks something like $1.d_1d_2\dots d_{p-2}0100\dots 00d_qd_{q+1}\dots$.

If $x - y$ were to look as in the second case, then $x - y = 2^{1-p}a + 2^{-p} - c$, or

$$x = 2^{1-p}a + 2^{-p} - c + y,$$

where a is an integer such that $2^{p-1} \leq a < 2^p$, and $0 < c \leq 2^{-q}$. Now, $y < 2^{p-q}$, since d_{q+n} , $n \geq 0$ are not all zeros. If $q \geq 2p$, then $y < 2^{-p}$ and $0 < c \leq 2^{-2p}$, so $-2^{-2p} < y - c < 2^{-p}$. The sum $2^{1-p}a + 2^{-p}$ consists of one digit to the left of the radix point and p digits to the right of the radix point, for a total of $p + 1$ significant

digits. Adding $y - c$ to this quantity cannot decrease the number of significant digits. Therefore, x cannot be a number with at most p significant digits if $q \geq 2p$ and $x - y$ (in infinite precision) looks something like $1.d_1d_2\dots d_{p-1}011\dots 11d_{q+1}d_{q+2}\dots$.

If $p < q < 2p + 1$, the theorem is false if using unbiased rounding: Consider the case where $x = 1 + 2^{1-p}$ and $y = 2^{-p}(1 - 2^{-p})$. $x - y$ looks something like

$$1.00\dots 00100\dots 001,$$

where there are $p - 1$ consecutive zeros immediately to the right of the radix point followed by a one, $p - 1$ consecutive zeros, and a one. If $p < q < 2p + 1$, then $z = 1 + 2^{-p}$, and the p -digit number that most closely approximates z is 1, whereas the p -digit number that most closely approximates $x - y$ is x .

If $p < q < 2p$, the theorem is also false, regardless of whether biased or unbiased rounding is used: Consider the case where $x = 1 + 2^{2-p}$ and $y = 2^{-p}(1 + 2^{1-p})$. $x - y$ looks something like

$$1.00\dots 001011\dots 11,$$

where there are $p - 2$ consecutive zeros immediately to the right of the radix point followed by a one, a zero, and $p - 1$ consecutive ones. If $p < q < 2p$, then $z = 1 + 3 \cdot 2^{-p+1}$, and the p -digit number that most closely approximates z is x , whereas the p -digit number that most closely approximates $x - y$ is $1 + 2^{1-p}$. ■

2.3 Multiplication

Theorem 3 *Let x and y be positive binary floating-point numbers whose significands consist of at most p digits, where $p \geq 4$, and let z be the binary floating-point number that most closely approximates xy , and whose significand consists of at most q digits, where $q > p$. If $q \geq 2p$, the binary floating-point number that most closely approximates xy and whose significand consists of at most p digits is the one that most closely approximates z . (This is not always the case if $p < q < 2p$.)*

Proof. We will restrict our attention to floating-point numbers y where $1 \leq y < 2$, and floating-point numbers x such that $1 \leq xy < 2$, since all pairs of positive binary floating-point numbers can be scaled by powers of two to meet these constraints.

There are potentially two cases in which our hypothesis is not trivially true: if xy (in infinite precision) were to look something like

$$1.d_1d_2\dots d_{p-2}0100\dots 00d_qd_{q+1}\dots,$$

[†]If $p = 2$, this is true only if using biased rounding.

[‡]The reason for restricting p to values no smaller than 4 is that q can be less than $2p$ otherwise.

where either d_q is one and d_{q+n} , $n > 0$, are all zeros, or d_q is zero and d_{q+n} , $n > 0$, are not all zeros; or if xy (in infinite precision) were to look something like

$$1.d_1d_2\dots d_{p-1}011\dots 11d_{q+1}d_{q+2}\dots$$

Now, $x = 2^e a$ for some integer e and some integer a such that $2^{p-1} \leq a < 2^p$. Similarly, $y = 2^{1-p} b$ for some integer b such that $2^{p-1} \leq b < 2^p$. Thus, $xy = 2^{e-p+1} ab$, with $2^{2p-2} \leq ab < 2^{2p}$. Therefore, since xy consists of at most $2p$ significant digits, xy cannot look like $1.d_1d_2\dots d_{p-2}0100\dots 00d_qd_{q+1}\dots$ or like $1.d_1d_2\dots d_{p-1}011\dots 11d_{q+1}d_{q+2}\dots$.

If $p < q < 2p$, the theorem can be false if using unbiased rounding: Consider the case where $p = 4$ and $x = y = 13$. $xy = 13^2 = 169 = 10101001_2$. If $p < q < 2p$, then $z = 168$, and the p -digit number that most closely approximates z is 160, whereas the p -digit number that most closely approximates xy is 176[¶]. ■

2.4 Division

Theorem 4 *Let x and y be positive binary floating-point numbers whose significands consist of at most p digits, where $p \geq 2$, and let z be the binary floating-point number that most closely approximates x/y , and whose significand consists of at most q digits, where $q > p$. If $q \geq 2p$, the binary floating-point number that most closely approximates x/y and whose significand consists of at most p digits is the one that most closely approximates z . This is not necessarily the case when using unbiased rounding and $p < q < 2p$.*

Proof. We will restrict our attention to floating-point numbers y where $2^{p-1} \leq y < 2^p$, and floating-point numbers x such that $1 \leq x/y < 2$, since all pairs of positive binary floating-point numbers can be scaled by powers of two to meet these constraints.

There are potentially two cases in which our hypothesis is not trivially true: if x/y (in infinite precision) were to look something like

$$1.d_1d_2\dots d_{p-2}0100\dots 00d_qd_{q+1}\dots,$$

where either d_q is one and d_{q+n} , $n > 0$, are all zeros, or d_q is zero and d_{q+n} , $n > 0$, are not all zeros; or if x/y (in infinite precision) were to look something like

$$1.d_1d_2\dots d_{p-1}011\dots 11d_{q+1}d_{q+2}\dots$$

If x/y were to look as in the first case, then $x/y = 2^{2-p}a + 2^{-p} + c$, or

$$x = 2^{2-p}ay + 2^{-p}y + cy,$$

[¶]Another example is $p = 5$ and $x = y = 23$, and yet another is $p = 6$, $x = 45$, and $y = 59$. The latter example shows that this theorem can be false if $p \geq 6$ and $p < q < 2p$, even when using biased rounding.

where a is an integer such that $2^{p-2} \leq a < 2^{p-1}$, and $0 < c \leq 2^{-q}$. Now, $2^{2-p}ay + 2^{-p}y$ consists of at least p digits to the left of the radix point, and no more than p digits to the right of the radix point. If $q \geq 2p$, then $0 < cy < 2^{-p}$, and adding cy to $2^{2-p}ay + 2^{-p}y$ can only increase the number of significant digits in the latter quantity. Therefore, x cannot be a number with at most p significant digits if $q \geq 2p$ and x/y (in infinite precision) looks something like $1.d_1d_2\dots d_{p-2}0100\dots 00d_qd_{q+1}\dots$.

If x/y were to look as in the second case, then $x/y = 2^{1-p}a + 2^{-p} - c$, or

$$x = 2^{1-p}ay + 2^{-p}y - cy,$$

where a is an integer such that $2^{p-1} \leq a < 2^p$, and $0 < c \leq 2^{-q}$. Now, $2^{1-p}ay + 2^{-p}y$ consists of at least p digits to the left of the radix point, and no more than p digits to the right of the radix point. If $q \geq 2p$, then $0 < cy < 2^{-p}$, and subtracting cy from $2^{1-p}ay + 2^{-p}y$ can only increase the number of significant digits in the latter quantity. Therefore, x cannot be a number with at most p significant digits if $q \geq 2p$ and x/y (in infinite precision) looks something like $1.d_1d_2\dots d_{p-1}011\dots 11d_{q+1}d_{q+2}\dots$.

If $p < q < 2p$, the theorem is false if using unbiased rounding: Consider the case where $x = 1$ and $y = 1 - 2^{-p}$. Computing the first few terms of the Taylor series expansion yields

$$x/y = 1/(1 - 2^{-p}) \approx 1 + 2^{-p} + 2^{-2p} + 2^{-3p},$$

which looks something like

$$1.00\dots 00100\dots 00100\dots,$$

where there are $p - 1$ consecutive zeros immediately to the right of the radix point followed by a one, $p - 1$ consecutive zeros, and another one. If $p < q < 2p$, then $z = 1 + 2^{-p}$, and the p -digit number that most closely approximates z is 1, whereas the p -digit number that most closely approximates x/y is $1 + 2^{1-p}$. ■

2.5 Square root

Theorem 5 *Let x be a positive binary floating-point number whose significand consists of at most p digits, where $p \geq 2$, and let y be the binary floating-point number that most closely approximates \sqrt{x} , and whose significand consists of at most q digits, where $q > p$. The binary floating-point number that most closely approximates \sqrt{x} and whose significand consists of at most p digits is the one that most closely approximates y if and only if $q \geq 2p + 2$.*

[¶]Another example is $p = 4$, $x = 13$, and $y = 11$: $x/y = 1.0010111010001\dots$, so this theorem is most likely false if $p \geq 4$ and $p < q < 2p$, even when using biased rounding.

Proof. We will restrict our attention to floating-point numbers x where $1 \leq x < 4$, since all positive binary floating-point numbers can be written as $2^e x$, where e is an even integer and $1 \leq x < 4$.

There are potentially two cases in which our hypothesis is not trivially true: if \sqrt{x} (in infinite precision) were to look something like

$$1.d_1d_2\dots d_{p-2}0100\dots 00d_qd_{q+1}\dots,$$

where either d_q is one and $d_{q+n}, n > 0$, are all zeros, or d_q is zero and $d_{q+n}, n > 0$, are not all zeros; or if \sqrt{x} (in infinite precision) were to look something like

$$1.d_1d_2\dots d_{p-1}011\dots 11d_{q+1}d_{q+2}\dots$$

If \sqrt{x} were to look as in the first case, then

$$2^{2-p}a + 2^{-p} < \sqrt{x} \leq 2^{2-p}a + 2^{-p} + 2^{-q},$$

where a is an integer such that $2^{p-2} \leq a < 2^{p-1}$. Let $z = 2^{2-p}a$. This means that

$$\begin{aligned} x &> z^2 + 2^{1-p}z + 2^{-2p} \\ x &\leq z^2 + 2^{1-p}z + 2^{-2p} + 2^{1-q}z + 2^{1-p-q} + 2^{-2q}. \end{aligned}$$

Now, the lower bound for x is a number with exactly $2p$ digits to the right of the radix point and one or two digits to the left of the radix point. Thus, x cannot be a number with at most p significant digits unless the upper bound for x were no smaller than the smallest p -digit number larger than the lower bound.

However, if $q \geq 2p$ (a stricter constraint than was mentioned in the theorem), then the upper bound cannot be greater than or equal to this p -digit number. The reason is that $z^2 + 2^{1-p}z$ has at most $2p - 3$ digits to the right of the radix point, while $2^{-2p} + 2^{1-q}z + 2^{1-p-q} + 2^{-2q} < 2^{3-2p}$. In other words, the lower bound looks something like $d_{-1}d_0.d_1d_2\dots d_{2p-3}001$, and no matter how big $2^{1-q}z + 2^{1-p-q} + 2^{-2q}$ is, adding the latter quantity to the lower bound cannot affect the lower bound's $(2p - 3)$ rd digit. Therefore, if $q \geq 2p$, \sqrt{x} (in infinite precision) cannot look something like $1.d_1d_2\dots d_{p-2}0100\dots 00d_qd_{q+1}\dots$.

If \sqrt{x} were to look as in the second case, then

$$2^{1-p}a + 2^{-p} - 2^{-q} \leq \sqrt{x} < 2^{1-p}a + 2^{-p},$$

where a is an integer such that $2^{p-1} \leq a < 2^p$. Let $z = 2^{1-p}a$. This means that

$$\begin{aligned} x &\leq z^2 + 2^{1-p}z + 2^{-2p} \\ x &> z^2 + 2^{1-p}z + 2^{-2p} - 2^{1-q}z - 2^{1-p-q} + 2^{-2q}. \end{aligned}$$

Now, the upper bound for x is a number with exactly $2p$ digits to the right of the radix point and one or two digits to the left of the radix point. Thus, x cannot

Operation	Minimum number of digits required
Addition	$2p + 1$
Subtraction	$2p + 1$ (unbiased rounding) $2p$ (biased rounding)
Multiplication	$2p$
Division	$2p$
Square root	$2p + 2$

Table 1: Number of digits required to emulate single precision arithmetic using double precision arithmetic

be a number with at most p significant digits unless the lower bound for x were no larger than the largest p -digit number smaller than the upper bound.

However, if $q \geq 2p + 2$, then the lower bound cannot be less than or equal to this p -digit number. The reason is that $2^{1-q}z + 2^{1-p-q} - 2^{-2q} < 2^{-2p}$. Therefore, subtracting this quantity from the upper bound cannot result in a number consisting of fewer than $2p$ digits to the right of the radix point. Therefore, if $q \geq 2p + 2$, \sqrt{x} (in infinite precision) cannot look something like $1.d_1d_2\dots d_{p-1}011\dots 11d_{q+1}d_{q+2}\dots$.

If $p < q < 2p + 2$, the theorem is false: Consider $x = 1 - 2^{-p}$. Computing the first few terms of the Taylor series expansion yields

$$\sqrt{1 - 2^{-p}} \approx 1 - 2^{-p-1} - 2^{-2p-3} - 2^{-3p-5},$$

which looks something like

$$0.11\dots 11011\dots 11011\dots,$$

where there are p consecutive ones immediately to the right of the radix point followed by a zero, $p + 1$ consecutive ones, and another zero. If $p < q < 2p + 2$, then $y = 1 - 2^{-p-1}$, and the p -digit number that most closely approximates y is 1, whereas the p -digit number that most closely approximates \sqrt{x} is x . ■

3 Conclusion

This paper has shown that in order to emulate single precision floating-point arithmetic faithfully using double precision arithmetic, if results are rounded to the nearest representable floating-point number, double precision floating-point numbers must consist of more than twice as many significant digits as single precision floating-point numbers. More specifically, Table 1 lists the minimum number of significant digits required for each of the five arithmetic operations. In this table, p is the number of digits in the significands of single precision numbers, *Addition* refers to the addition of two numbers with the same sign, and *Subtraction* refers to the addition of two numbers with opposite signs.

It may not be apparent from Table 1 that with just two or three exceptions, even if the significands of double precision numbers were to consist of only $2p$ significant digits, one could avoid changing the final result of an arithmetic operation when double rounding occurs: If one were to make just a few slight modifications to Priest's proof of Theorem 5 [6] (which postdates this author's proof by a few days), one could show that if it were not for numbers of the form $2^e(1 - 2^{-p})$, where e is an even integer, only $2p$ digits would be needed for square root. Also, for subtraction (using unbiased rounding) and addition, if double precision numbers consisted of only $2p$ significant digits, double rounding could change the final result only if the operand whose magnitude is smaller were equal to $2^{e-p}(1 - 2^{-p})$, where e is the exponent of the operand whose magnitude is larger.

This author would be interested in knowing if it is possible to show that for any $p \geq 6$, q must be greater than or equal to $2p$ in order for Theorem 3 to be true, regardless of whether biased or unbiased rounding is used. Also, this author would like to know if it is possible to show that for any $p \geq 4$, q must be greater than or equal to $2p$ in order for Theorem 4 to be true when biased rounding is used.

References

- [1] D. Goldberg. Re: square root and double rounding. July 1995. Private communication via electronic mail.
- [2] D. Hough. Re: square root and double rounding. June 1995. Private communication via electronic mail.
- [3] *IEEE Standard for Binary Floating-Point Arithmetic*. The Institute of Electrical and Electronic Engineers, Inc., New York, 1985. ANSI/IEEE Std 754-1985.
- [4] *IEEE Standard for Radix-Independent Floating-Point Arithmetic*. The Institute of Electrical and Electronic Engineers, Inc., New York, 1987. ANSI/IEEE Std 854-1987.
- [5] D. A. Patterson and J. L. Hennessy. *Computer Architecture: A Quantitative Approach*, Appendix A: Computer Arithmetic, page A-29. Morgan Kaufmann Publishers, San Mateo, Calif., first edition, 1990. (The author of Appendix A is D. Goldberg).
- [6] D. M. Priest. Re: square root and double rounding. June 1995. Electronic mail message sent to the numeric-interest@validgh.com mailing list.