Complete Interval Arithmetic and its Implementation on the Computer

Ulrich W. Kulisch
Institut für Angewandte und Numerische Mathematik
Universität Karlsruhe

March 11, 2008

Abstract: Let $I\mathbb{R}$ be the set of closed and bounded intervals of real numbers. Arithmetic in $I\mathbb{R}$ can be defined via the power set $\mathbb{P}\mathbb{R}$ (the set of all subsets) of real numbers. If in case of division zero is not contained in the divisor arithmetic in $I\mathbb{R}$ is an algebraically closed subset of the arithmetic in $\mathbb{P}\mathbb{R}$. Arithmetic in $\mathbb{P}\mathbb{R}$ allows division by an interval that contains zero also. This results in closed intervals of real numbers which, however, are no longer bounded. The union of the set $I\mathbb{R}$ with these new intervals is denoted by $(I\mathbb{R})$.

The paper shows that arithmetic operations can be extended to all elements of the set $(I\mathbb{R})$. On the computer, arithmetic in $(I\mathbb{R})$ is approximated by arithmetic in the subset (IF) of closed intervals over the floating-point numbers $F \subset \mathbb{R}$. The usual exceptions of floating-point arithmetic like underflow, overflow, division by zero, or invalid operation do not occur in (IF).

Key words: computer arithmetic, floating-point arithmetic, interval arithmetic, arithmetic standards.

1 Introduction or a Vision of Future Computing

Computers are getting ever faster. The time can already be foreseen when the PC will be a teraflops computer. With this tremendous computing power scientific computing will experience a significant shift from floating-point arithmetic toward increased use of interval arithmetic. With little hardware expenditure interval arithmetic can be made as fast as simple floating-point arithmetic [3]. Nearly everything that is needed for fast interval arithmetic is already available on most existing processors (made available for multimedia applications). What is still missing are the arithmetic operations with the directed roundings. In its ultimate stage of development interval arithmetic is a well rounded complete and exception-free calculus. Exceptions of floating-point arithmetic like underflow, overflow, division by zero, or invalid operations do not occur in interval arithmetic. This will be shown in this article. For interval evaluation of an algorithm (a sequence of arithmetic operations) in the real number field a theorem by R. E. Moore [7] states that increasing the precision by k digits reduces the error bounds by b^{-k}, i.e., results can always be guaranteed to a number of correct digits by using variable precision interval arithmetic (for details see [1], [9]). Long interval arithmetic can be made very fast by an exact dot product and complete arithmetic [4]. By pipelining an exact dot product can be computed in the time the processor needs to read the data, i.e., it comes with extreme speed. Long interval arithmetic fully benefits from this speed. It can easily be applied by operator overloading.

Dagstuhl Seminar Proceedings 08021
Numerical Validation in Current Hardware Architectures
http://drops.dagstuhl.de/opus/volltexte/2008/1446

The tremendous progress in computer technology should be accompanied by extension of the mathematical capacity of the computer. A balanced standard of computer arithmetic should require that the basic components of modern computing (floating-point arithmetic, interval arithmetic, and an exact dot product) should be provided by the computer's hardware. See [5].

2 Remarks on Floating-Point Arithmetic

Computing is usually done in the set of real numbers $I\!R$. The real numbers can be defined as a conditionally complete, linearly ordered field. Conditionally complete means that every bounded subset has an infimum and a supremum. Every conditionally ordered set can be completed by joining a least and a greatest element. In case of the real numbers these are called $-\infty$ and $+\infty$. Then $I\!R^* := I\!R \cup \{-\infty\} \cup \{+\infty\}$ is a complete lattice. The elements $-\infty$ and $+\infty$, however, are not real numbers, i.e., they are not elements of the field. The cancellation law $a + c = b + c \Rightarrow a = b$, for instance, does not hold for $c = \infty$.

A real number consists of a sign, an integral, and a fractional part, for instance: $\pm 345.789123 \cdots \in I\!R$. The point may be shifted to any other position if we compensate for this shifting by a corresponding power of b (here $b = 10$). If the point is shifted immediately to the left of the first nonzero digit: $\pm 0.345789123 \cdots \cdot 10^3$ the representation is called normalized. Zero is the only real number that has no such representation. It needs a particular encoding. Thus a normalized real number consists of a signed fractional part m (mantissa) and an integer exponent e and we have $|m| < 1$.

Only subsets of these numbers can be represented on the computer. If the mantissa in truncated after the l^{th} digit and the exponent is limited by $e_{min} < e < e_{max}$ one speaks of a floating-point number. **The set F of all such floating-point numbers is a finite subset of $I\!R$.**

Arithmetic for floating-point numbers may cause exceptions. Well known such exceptions are underflow, overflow, division by zero, or invalid operation. To avoid interruption of program execution in case of an exception the so-called IEEE floating-point arithmetic standard provides additional elements and defines operations for these, for instance, $4/0 =: \infty, -4/0 =: -\infty, \infty - \infty =: NaN, 0 \cdot \infty =: NaN, \infty/\infty =: NaN, 0/0 =: NaN, 1/(-\infty) =: -0, (-0.3)/\infty =: -0$. It should be clear, however, that these artificial strategic objects $-\infty, +\infty, NaN, -0$,[1] or $+0$ with their operations are not elements of the real number field and thus are not floating-point numbers.

3 Arithmetic for Intervals of $I I\!R$ and IF

Interval Arithmetic is another arithmetical tool. **It solely deals with sets of real numbers.** All the exceptions of floating-point numbers mentioned above and the strategic objects to deal with them do not occur and are not needed in interval arithmetic. The symbol $I I\!R$ usually denotes the set of closed and bounded intervals of $I\!R$. Arithmetic in $I I\!R$ can be interpreted as a systematic calculus to deal with inequalities. We assume here that the basic rules for arithmetic in $I I\!R$ with zero not in the divisor are known to the reader. It is a fascinating result that in contrast to floating-point arithmetic interval arithmetic even on computers can be further developed into a well rounded, exception-free, closed calculus. We briefly sketch this development here.

[1] In $I\!R$, 0 is defined as the neutral element of addition. From the assumption that there are two such elements 0 and $0'$ it follows immediately that they are equal: $0 + 0' = 0 = 0'$.

In case of floating-point arithmetic the crucial operation that leads to the exceptional strategic objects mentioned above is division by zero. So we begin our study of extended interval arithmetic with defining division by an interval that contains zero.

The set $I\mathbb{R}$ is a subset of the power set $\mathbb{P}\mathbb{R}$ (which is the set of all subsets) of real numbers. For $A, B \in \mathbb{P}\mathbb{R}$ arithmetic operations are defined by

$$\bigwedge_{A,B\in\mathbb{P}\mathbb{R}} A \circ B := \{a \circ b \mid a \in A \wedge b \in B\}, \text{ for all } \circ \in \{+, -, \cdot, /\}. \tag{3.1}$$

The following properties are obvious and immediate consequences of this definition:

$$A \subseteq B \wedge C \subseteq D \Rightarrow A \circ C \subseteq B \circ D, \text{ for all } A, B, C, D \in \mathbb{P}\mathbb{R}, \tag{3.2}$$

and in particular

$$a \in A \wedge b \in B \Rightarrow a \circ b \in A \circ B, \text{ for all } A, B \in \mathbb{P}\mathbb{R}. \tag{3.3}$$

Property (3.2) is called *inclusion-isotony* (or *inclusion-monotonicity*). Property (3.3) is called the *inclusion property.* (3.2) and (3.3) are the fundamental properties of interval arithmetic. Under the assumption $0 \notin B$ for division, the intervals of $I\mathbb{R}$ are an algebraically closed subset[2] of the power set $\mathbb{P}\mathbb{R}$, i.e., an operation for intervals of $I\mathbb{R}$ performed in $\mathbb{P}\mathbb{R}$ always delivers an interval of $I\mathbb{R}$.

On the computer arithmetic in $I\mathbb{R}$ is approximated by an arithmetic in IF. An interval of IF represents a continuous set of real numbers with floating-point bounds of F. Arithmetic operations in IF are defined by those in $I\mathbb{R}$ with the lower bound of the result rounded downwards and the upper bound rounded upwards.

In case of floating-point arithmetic division by zero does not lead to a real number. In contrast to this in interval arithmetic division by an interval that contains zero can be defined in a strict mathematical manner. The result again is a set of real numbers.

In accordance with (3.1) division in $I\mathbb{R}$ is defined by

$$\bigwedge_{A,B\in I\mathbb{R}} A/B := \{a/b \mid a \in A \wedge b \in B\}. \tag{3.4}$$

The quotient a/b is defined as the inverse operation of multiplication, i.e., as the solution of the equation $b \cdot x = a$. Thus (3.4) can be written in the form

$$\bigwedge_{A,B\in I\mathbb{R}} A/B := \{x \mid bx = a \wedge a \in A \wedge b \in B\}. \tag{3.5}$$

For $0 \notin B$ (3.4) and (3.5) are equivalent. While in $\mathbb{R}$ division by zero is not defined the representation of A/B by (3.5) allows definition of the operation and interpretation of the result for $0 \in B$ also.

By way of interpreting (3.5) for $A = [a_1, a_2]$ and $B = [b_1, b_2] \in I\mathbb{R}$ with $0 \in B$ the following eight distinct cases can be set out:

1	$0 \in A$,	$0 \in B$.
2	$0 \notin A$,	$B = [0, 0]$.
3	$a_1 \le a_2 < 0$,	$b_1 < b_2 = 0$.
4	$a_1 \le a_2 < 0$,	$b_1 < 0 < b_2$.
5	$a_1 \le a_2 < 0$,	$0 = b_1 < b_2$.
6	$0 < a_1 \le a_2$,	$b_1 < b_2 = 0$.
7	$0 < a_1 \le a_2$,	$b_1 < 0 < b_2$.
8	$0 < a_1 \le a_2$,	$0 = b_1 < b_2$.

[2]as the integers are of the real numbers.

The list distinguishes the cases $0 \in A$ (case 1) and $0 \notin A$ (cases 2 to 8). Since it is generally assumed that $0 \in B$, these eight cases indeed cover all possibilities. Since every $x \in I\!R$ fulfills the equation $0 \cdot x = 0$ we obtain in case 1: $A/B = I\!R = (-\infty, +\infty)$. Here the round brackets indicate that the bounds are not included in the set. In case 2 the set defined by (3.5) consists of all elements which fulfill the equation $0 \cdot x = a$ for $a \in A$. Since $0 \notin A$, there is no real number which fulfills this equation. Thus A/B is the empty set, i.e., $A/B = \emptyset$.

case	$A = [a_1, a_2]$	$B = [b_1, b_2]$	B'	A/B'	A/B
1	$0 \in A$	$0 \in B$			$(-\infty, +\infty)$
2	$0 \notin A$	$B = [0, 0]$			$\emptyset$
3	$a_2 < 0$	$b_1 < b_2 = 0$	$[b_1, (-\epsilon)]$	$[a_2/b_1, a_1/(-\epsilon)]$	$[a_2/b_1, +\infty)$
4	$a_2 < 0$	$b_1 < 0 < b_2$	$[b_1, (-\epsilon)]$	$[a_2/b_1, a_1/(-\epsilon)]$	$(-\infty, a_2/b_2]$
			$\cup\ [\epsilon, b_2]$	$\cup\ [a_1/\epsilon, a_2/b_2]$	$\cup\ [a_2/b_1, +\infty)$
5	$a_2 < 0$	$0 = b_1 < b_2$	$[\epsilon, b_2]$	$[a_1/\epsilon, a_2/b_2]$	$(-\infty, a_2/b_2]$
6	$a_1 > 0$	$b_1 < b_2 = 0$	$[b_1, (-\epsilon)]$	$[a_2/(-\epsilon), a_1/b_1]$	$(-\infty, a_1/b_1]$
7	$a_1 > 0$	$b_1 < 0 < b_2$	$[b_1, (-\epsilon)]$	$[a_2/(-\epsilon), a_1/b_1]$	$(-\infty, a_1/b_1]$
			$\cup\ [\epsilon, b_2]$	$\cup\ [a_1/b_2, a_2/\epsilon]$	$\cup\ [a_1/b_2, +\infty)$
8	$a_1 > 0$	$0 = b_1 < b_2$	$[\epsilon, b_2]$	$[a_1/b_2, a_2/\epsilon]$	$[a_1/b_2, +\infty)$

Table 1: The eight cases of interval division A/B, with $A, B \in I I\!R$, and $0 \in B$.

In all other cases $0 \notin A$ also. We have already observed under case 2 that the element 0 in B does not contribute to the solution set. So it can be excluded without changing the set A/B.

So the general rule for computing the set A/B by (3.5) is to remove its zero from the interval B and replace it by a small positive or negative number ϵ as the case may be. The resulting set is denoted by B' and represented in column 4 of Table 1. With this B' the solution set A/B' can now easily be computed by applying the rules for closed and bounded real intervals. The results are shown in column 5 of Table 1. Now the desired result A/B as defined by (3.5) is obtained if in column 5 ϵ tends to zero.

Thus in the cases 3 to 8 the results are obtained by the limit process $A/B = \lim_{\epsilon \to 0} A/B'$. The solution set A/B is shown in the last column of Table 1 for all the eight cases. There, as usual in mathematics round brackets indicate that the bound is not included in the set. In contrast to this square brackets denote closed interval ends, i.e., the bound is included.

The operands A and B of the division A/B in Table 1 are intervals of $I I\!R$. The results of the division shown in the last column, however, are no longer intervals of $I I\!R$. The result is now an element of the power set $I\!P I\!R$. With the exception of case 2 the result is now a set which stretches continuously to $-\infty$ or $+\infty$ or both.

In two cases (rows 4 and 7 in Table 1) the result consists of the union of two distinct sets of the form $(-\infty, c_2] \cup [c_1, +\infty)$. These cases can easily be identified by the signs of the bounds of the divisor. Within the given framework of existing processors only one interval can be delivered as the result of an interval operation. In the cases 4 and 7 of Table 1 the result, yet, can be returned as an improper interval $[c_1, c_2]$ where the left hand bound is higher than the right hand bound.

Motivated by the extended interval Newton method[3] it is reasonable to separate these results into the two distinct sets: $(-\infty, c_2]$ and $[c_1, +\infty)$. The fact that an arithmetic operation delivers two distinct results seems to be a totally new situation in computing. Evaluation of the square root, however, also delivers two results and we have learned to live with it. Computing certainly is able to deal with this situation.

A principle solution of the problem would be for the computer to provide a flag for *distinct intervals*. In cases 4 and 7 of Table 1 the flag would be raised and signaled to the user. The user may then apply a routine of his choice to deal with the situation as is appropriate for his application.[4]

If during a computation in the real number field zero appears as a divisor the computation should be stopped immediately. In floating-point arithmetic the situation is different. Zero may be the result of an underflow. In such a case a corresponding interval computation would not deliver zero but a small interval with zero as one bound and a tiny positive or negative number as the other bound. In this case division is well defined by Table 1. The result is a closed interval which stretches continuously to $-\infty$ or $+\infty$ as the case may be. In the real number field zero as a divisor is an accident. So in interval arithmetic division by an interval that contains zero as an interior point certainly will be a very rare appearance. An exception is the interval Newton method. Here, however, it is clear how the situation has to be handled. See, for instance, [4].

In the literature an improper interval $[c_1, c_2]$ with $c_1 > c_2$ occasionally is called an 'exterior interval'. On the number circle an 'exterior interval' is interpreted as an interval with infinity as an interior point. We do not follow this line here. Interval arithmetic is defined as an arithmetic for sets of real numbers. Operations for real numbers which deliver ∞ as their result do not exist. Here and in the following the symbols $-\infty$ and $+\infty$ are only used to describe sets of real numbers.

After the splitting of improper intervals into two distinct sets only four kinds of result come from division by an interval of $I\mathbb{R}$ which contains zero:

$$\emptyset, \quad (-\infty, a], \quad [b, +\infty), \quad \text{and} \quad (-\infty, +\infty). \tag{3.6}$$

We call such elements extended intervals. The union of the set of closed and bounded intervals of $I\mathbb{R}$ with the set of extended intervals is denoted by $(I\mathbb{R})$. The elements of the set $(I\mathbb{R})$ are themselves simply called intervals. $(I\mathbb{R})$ is the set of closed intervals of $\mathbb{R}$. (A subset of $\mathbb{R}$ is called closed if the complement is open.)

Intervals of $I\mathbb{R}$ and of $(I\mathbb{R})$ are sets of real numbers. $-\infty$ and $+\infty$ are not elements of these intervals. It is fascinating that arithmetic operations can be introduced for all elements of the set $(I\mathbb{R})$ in an exception-free manner. This will be shown in the next section.

On a computer only subsets of the real numbers are representable. We assume now that F is the set of floating-point numbers of a given computer. An interval between two floating-point bounds represents the continuous set of real numbers between these bounds. Similarly, except for the empty set, also extended intervals represent continuous sets of real numbers.

To transform the eight cases of division by an interval of $I\mathbb{R}$ which contains zero into computer executable operations we assume now that the operands A and

[3]Newton's method reaches its ultimate elegance and strength in the extended interval Newton method. If division by an interval that contains zero delivers two distinct sets the computation is continued along two separate paths, one for each interval. This is how the extended interval Newton method separates different zeros from each other and finally computes all zeros in a given domain. If the interval Newton method delivers the empty set, the method has proved that there is no zero in the initial interval.

[4]This routine could be: modify the operands and recompute, or continue the computation with one of the sets and ignore the other one, or put one of the sets on a list and continue the computation with the other one, or stop computing, or ignore the flag, or any other action.

B are floating-point intervals of IF. To obtain a computer representable result we round the result shown in the last column of Table 1 into the least computer representable superset. That is, the lower bound of the result has to be computed with rounding downwards and the upper bound with rounding upwards. Thus on the computer the eight cases of division by an interval of IF which contains zero have to be performed as shown in Table 2.

case	$A = [a_1, a_2]$	$B = [b_1, b_2]$	$A \mathbin{\Diamond\!\!\!/} B$
1	$0 \in A$	$0 \in B$	$(-\infty, +\infty)$
2	$0 \notin A$	$B = [0, 0]$	$\emptyset$
3	$a_2 < 0$	$b_1 < b_2 = 0$	$[a_2 \bigtriangledown b_1, +\infty)$
4	$a_2 < 0$	$b_1 < 0 < b_2$	$(-\infty, a_2 \bigtriangleup b_2] \cup [a_2 \bigtriangledown b_1, +\infty)$
5	$a_2 < 0$	$0 = b_1 < b_2$	$(-\infty, a_2 \bigtriangleup b_2]$
6	$a_1 > 0$	$b_1 < b_2 = 0$	$(-\infty, a_1 \bigtriangleup b_1]$
7	$a_1 > 0$	$b_1 < 0 < b_2$	$(-\infty, a_1 \bigtriangleup b_1] \cup [a_1 \bigtriangledown b_2, +\infty)$
8	$a_1 > 0$	$0 = b_1 < b_2$	$[a_1 \bigtriangledown b_2, +\infty)$

Table 2: The eight cases of interval division with $A, B \in IF$, and $0 \in B$.

Table 3 shows the same cases as Table 2 in another layout.

	$B = [0, 0]$	$b_1 < b_2 = 0$	$b_1 < 0 < b_2$	$0 = b_1 < b_2$
$a_2 < 0$	$\emptyset$	$[a_2 \bigtriangledown b_1, +\infty)$	$(-\infty, a_2 \bigtriangleup b_2]$ $\cup\ [a_2 \bigtriangledown b_1, +\infty)$	$(-\infty, a_2 \bigtriangleup b_2]$
$a_1 \le 0 \le a_2$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$
$0 < a_1$	$\emptyset$	$(-\infty, a_1 \bigtriangleup b_1]$	$(-\infty, a_1 \bigtriangleup b_1]$ $\cup\ [a_1 \bigtriangledown b_2, +\infty)$	$[a_1 \bigtriangledown b_2, +\infty)$

Table 3: The result of the interval division with $A, B \in IF$, and $0 \in B$.

Table 2 and Table 3 display the eight distinct cases of interval division $A \mathbin{\Diamond\!\!\!/} B$ with $A, B \in IF$ and $0 \in B$. On the computer the empty interval $\emptyset$ needs a particular encoding. ($+$`NaN`, $-$`NaN`) may be such an encoding. We explicitly stress that the symbols $-\infty$, $+\infty$, $-$`NaN`, and $+$`NaN` are used here only to represent the resulting sets. These symbols are not elements of these sets and no operations are defined for them.

Division by an interval of IF which contains zero on the computer also leads to extended intervals as shown in (3.6) with $a, b \in F$. The union of the set of closed and bounded intervals of IF with such extended intervals is denoted by (IF). (IF) is the set of closed intervals of F.

4 Arithmetic for Intervals of $(I\mathbb{R})$ and (IF)

For the sake of completeness arithmetic operations now have to be defined for all elements of $(I\mathbb{R})$ and (IF). Since the development of arithmetic operations in $(I\mathbb{R})$ and (IF) follows an identical pattern we skip here the introduction of the arithmetic in $(I\mathbb{R})$ and restrict the consideration to the development of arithmetic in (IF). This is the arithmetic that has to be provided on the computer.

First of all any operation with the empty set is defined to be the empty set again.

The general procedure to define all other operations follows a continuity principle. Bounds like $-\infty$ and $+\infty$ in the operands A and B are replaced by a very large negative and a very large positive number respectively. Then the basic rules for the arithmetic operations in $I\mathbb{R}$ and IF are applied. In the following tables these rules are repeated and printed in bold letters.

In the resulting formulas the very large negative number is then shifted to $-\infty$ and the very large positive number to $+\infty$. Finally, very simple and well established rules of real analysis like $\infty * x = \infty$ for $x > 0$, $\infty * x = -\infty$ for $x < 0$, $x/\infty = x/-\infty = 0$, $\infty * \infty = \infty$, $(-\infty) * \infty = -\infty$ are applied together with variants obtained by applying the sign rules and the law of commutativity.

Two situations have to be treated separately. These are the cases shown in rows 1 and 2 of Table 1.

If $0 \in A$ and $0 \in B$ (row 1 of Table 1), the result consists of all real numbers, i.e., $A/B = (-\infty, +\infty)$. This applies to rows 2, 5, 6, and 8 of Table 8.

If $0 \notin A$ and $B = [0, 0]$ (row 2 of Table 1), the result of the division is the empty set, i.e., $A/B = \varnothing$. This applies to rows 1, 3, 4, and 7 of column 1 of Table 8.

We summarize the complete set of arithmetic operations for interval arithmetic in (IF) that should be provided on the computer in the next section.

In summary it can be said that after a possible splitting of an improper interval into two separate intervals the result of arithmetic operations for intervals of (IF) always leads to intervals of (IF) again. No exceptions or artificial strategic objects do occur performing these operations. The reader should prove this assertion by realizing the operations shown in the tables of the following section.

For the development in the preceding sections it was essential to distinguish between round an square brackets. If the bracket adjacent to a bound is round, the bound is not included in the interval; if it is square, the bound is included in the interval.

5 Complete Arithmetic for Intervals of (IF)

Addition	$(-\infty, b_2]$	$[b_1, b_2]$	$[b_1, +\infty)$	$(-\infty, +\infty)$
$(-\infty, a_2]$	$(-\infty, a_2 \overset{\triangle}{+} b_2]$	$(-\infty, a_2 \overset{\triangle}{+} b_2]$	$(-\infty, +\infty)$	$(-\infty, +\infty)$
$[a_1, a_2]$	$(-\infty, a_2 \overset{\triangle}{+} b_2]$	$[\boldsymbol{a_1} \overset{\triangledown}{+} \boldsymbol{b_1}, \boldsymbol{a_2} \overset{\triangle}{+} \boldsymbol{b_2}]$	$[a_1 \overset{\triangledown}{+} b_1, +\infty)$	$(-\infty, +\infty)$
$[a_1, +\infty)$	$(-\infty, +\infty)$	$[a_1 \overset{\triangledown}{+} b_1, +\infty)$	$[a_1 \overset{\triangledown}{+} b_1, +\infty)$	$(-\infty, +\infty)$
$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$

Table 4: Addition of extended intervals on the computer.

Subtraction	$(-\infty, b_2]$	$[b_1, b_2]$	$[b_1, +\infty)$	$(-\infty, +\infty)$
$(-\infty, a_2]$	$(-\infty, +\infty)$	$(-\infty, a_2 \overset{\triangle}{-} b_1]$	$(-\infty, a_2 \overset{\triangle}{-} b_1]$	$(-\infty, +\infty)$
$[a_1, a_2]$	$[a_1 \overset{\triangledown}{-} b_2, +\infty)$	$[\boldsymbol{a_1} \overset{\triangledown}{-} \boldsymbol{b_2}, \boldsymbol{a_2} \overset{\triangle}{-} \boldsymbol{b_1}]$	$(-\infty, a_2 \overset{\triangle}{-} b_1]$	$(-\infty, +\infty)$
$[a_1, +\infty)$	$[a_1 \overset{\triangledown}{-} b_2, +\infty)$	$[a_1 \overset{\triangledown}{-} b_2, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$
$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$

Table 5: Subtraction of extended intervals on the computer.

	$[b_1, b_2]$	$[b_1, b_2]$	$[b_1, b_2]$		$(-\infty, b_2]$	$(-\infty, b_2]$	$[b_1, +\infty)$	$[b_1, +\infty)$	
Multiplication	$b_2 \le 0$	$b_1 < 0 < b_2$	$b_1 \ge 0$	$[0, 0]$	$b_2 \le 0$	$b_2 \ge 0$	$b_1 \le 0$	$b_1 \ge 0$	$(-\infty, +\infty)$
$[a_1, a_2], a_2 \le 0$	$[\boldsymbol{a_2 \bigtriangledown b_2, a_1 \bigtriangleup b_1}]$	$[\boldsymbol{a_1 \bigtriangledown b_2, a_1 \bigtriangleup b_1}]$	$[\boldsymbol{a_1 \bigtriangledown b_2, a_2 \bigtriangleup b_1}]$	$[0, 0]$	$[a_2 \bigtriangledown b_2, +\infty)$	$[a_1 \bigtriangledown b_2, +\infty)$	$(-\infty, a_1 \bigtriangleup b_1]$	$(-\infty, a_2 \bigtriangleup b_1]$	$(-\infty, +\infty)$
$a_1 < 0 < a_2$	$[\boldsymbol{a_2 \bigtriangledown b_1, a_1 \bigtriangleup b_1}]$	$[\mathbf{min}(\boldsymbol{a_1 \bigtriangledown b_2, a_2 \bigtriangledown b_1}),$ $\mathbf{max}(\boldsymbol{a_1 \bigtriangleup b_1, a_2 \bigtriangleup b_2})]$	$[\boldsymbol{a_1 \bigtriangledown b_2, a_2 \bigtriangleup b_2}]$	$[0, 0]$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$
$[a_1, a_2], a_1 \ge 0$	$[\boldsymbol{a_2 \bigtriangledown b_1, a_1 \bigtriangleup b_2}]$	$[\boldsymbol{a_2 \bigtriangledown b_1, a_2 \bigtriangleup b_2}]$	$[\boldsymbol{a_1 \bigtriangledown b_1, a_2 \bigtriangleup b_2}]$	$[0, 0]$	$(-\infty, a_1 \bigtriangleup b_2]$	$(-\infty, a_2 \bigtriangleup b_2]$	$[a_2 \bigtriangledown b_1, +\infty)$	$[a_1 \bigtriangledown b_1, +\infty)$	$(-\infty, +\infty)$
$[0, 0]$	$[0, 0]$	$[0, 0]$	$[0, 0]$	$[0, 0]$	$[0, 0]$	$[0, 0]$	$[0, 0]$	$[0, 0]$	$[0, 0]$
$(-\infty, a_2], a_2 \le 0$	$[a_2 \bigtriangledown b_2, +\infty)$	$(-\infty, +\infty)$	$(-\infty, a_2 \bigtriangleup b_1]$	$[0, 0]$	$[a_2 \bigtriangledown b_2, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, a_2 \bigtriangleup b_1]$	$(-\infty, +\infty)$
$(-\infty, a_2], a_2 \ge 0$	$[a_2 \bigtriangledown b_1, +\infty)$	$(-\infty, +\infty)$	$(-\infty, a_2 \bigtriangleup b_2]$	$[0, 0]$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$
$[a_1, +\infty), a_1 \le 0$	$(-\infty, a_1 \bigtriangleup b_1]$	$(-\infty, +\infty)$	$[a_1 \bigtriangledown b_2, +\infty)$	$[0, 0]$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$
$[a_1, +\infty), a_1 \ge 0$	$(-\infty, a_1 \bigtriangleup b_2]$	$(-\infty, +\infty)$	$[a_1 \bigtriangledown b_1, +\infty)$	$[0, 0]$	$(-\infty, a_1 \bigtriangleup b_2]$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$[a_1 \bigtriangledown b_1, +\infty)$	$(-\infty, +\infty)$
$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$[0, 0]$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$

Table 6: Multiplication of extended intervals on the computer.

Division $\boldsymbol{0 \notin B}$	$[b_1, b_2]$ $b_2 < 0$	$[b_1, b_2]$ $b_1 > 0$	$(-\infty, b_2]$ $b_2 < 0$	$[b_1, +\infty)$ $b_1 > 0$
$[a_1, a_2], a_2 \leq 0$	$[\boldsymbol{a_2} \triangledown \boldsymbol{b_1}, \boldsymbol{a_1} \triangle \boldsymbol{b_2}]$	$[\boldsymbol{a_1} \triangledown \boldsymbol{b_1}, \boldsymbol{a_2} \triangle \boldsymbol{b_2}]$	$[0, a_1 \triangle b_2]$	$[a_1 \triangledown b_1, 0]$
$[a_1, a_2], a_1 < 0 < a_2$	$[\boldsymbol{a_2} \triangledown \boldsymbol{b_2}, \boldsymbol{a_1} \triangle \boldsymbol{b_2}]$	$[\boldsymbol{a_1} \triangledown \boldsymbol{b_1}, \boldsymbol{a_2} \triangle \boldsymbol{b_1}]$	$[a_2 \triangledown b_2, a_1 \triangle b_2]$	$[a_1 \triangledown b_1, a_2 \triangle b_1]$
$[a_1, a_2], a_1 \geq 0$	$[\boldsymbol{a_2} \triangledown \boldsymbol{b_2}, \boldsymbol{a_1} \triangle \boldsymbol{b_1}]$	$[\boldsymbol{a_1} \triangledown \boldsymbol{b_2}, \boldsymbol{a_2} \triangle \boldsymbol{b_1}]$	$[a_2 \triangledown b_2, 0]$	$[0, a_2 \triangle b_1]$
$[0, 0]$	$[0, 0]$	$[0, 0]$	$[0, 0]$	$[0, 0]$
$(-\infty, a_2], a_2 \leq 0$	$[a_2 \triangledown b_1, +\infty)$	$(-\infty, a_2 \triangle b_2]$	$[0, +\infty)$	$(-\infty, 0]$
$(-\infty, a_2], a_2 \geq 0$	$[a_2 \triangledown b_2, +\infty)$	$(-\infty, a_2 \triangle b_1]$	$[a_2 \triangledown b_2, +\infty)$	$(-\infty, a_2 \triangle b_1]$
$[a_1, +\infty), a_1 \leq 0$	$(-\infty, a_1 \triangle b_2]$	$[a_1 \triangledown b_1, +\infty)$	$(-\infty, a_1 \triangle b_2]$	$[a_1 \triangledown b_1, +\infty)$
$[a_1, +\infty), a_1 \geq 0$	$(-\infty, a_1 \triangle b_1]$	$[a_1 \triangledown b_2, +\infty)$	$(-\infty, 0]$	$[0, +\infty)$
$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$

Table 7: Division of extended intervals with $0 \notin B$ on the computer.

Division $\boldsymbol{0} \in \boldsymbol{B}$	$B =$ $[0,0]$	$[b_1, b_2]$ $b_1 < b_2 = 0$	$[b_1, b_2]$ $b_1 < 0 < b_2$	$[b_1, b_2]$ $0 = b_1 < b_2$	$(-\infty, b_2]$ $b_2 = 0$	$(-\infty, b_2]$ $b_2 > 0$	$[b_1, +\infty)$ $b_1 < 0$	$[b_1, +\infty)$ $b_1 = 0$	$(-\infty, +\infty)$
$[a_1, a_2], a_2 < 0$	$\varnothing$	$[\boldsymbol{a_2} \triangledown \boldsymbol{b_1}, +\infty)$	$(-\infty, \boldsymbol{a_2} \vartriangle \boldsymbol{b_2}]$ $\cup\, [\boldsymbol{a_2} \triangledown \boldsymbol{b_1}, +\infty)$	$(-\infty, \boldsymbol{a_2} \vartriangle \boldsymbol{b_2}]$	$[0, +\infty)$	$(-\infty, a_2 \vartriangle b_2]$ $\cup\, [0, +\infty)$	$(-\infty, 0]$ $\cup\, [a_2 \triangledown b_1, +\infty)$	$(-\infty, 0]$	$(-\infty, +\infty)$
$[a_1, a_2], a_1 \leq 0 \leq a_2$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$
$[a_1, a_2], a_1 > 0$	$\varnothing$	$(-\infty, \boldsymbol{a_1} \vartriangle \boldsymbol{b_1}]$	$(-\infty, \boldsymbol{a_1} \vartriangle \boldsymbol{b_1}]$ $\cup\, [\boldsymbol{a_1} \triangledown \boldsymbol{b_2}, +\infty)$	$[\boldsymbol{a_1} \triangledown \boldsymbol{b_2}, +\infty)$	$(-\infty, 0]$	$(-\infty, 0]$ $\cup\, [a_1 \triangledown b_2, +\infty)$	$(-\infty, a_1 \vartriangle b_1]$ $\cup\, [0, +\infty)$	$[0, +\infty)$	$(-\infty, +\infty)$
$(-\infty, a_2], a_2 < 0$	$\varnothing$	$[a_2 \triangledown b_1, +\infty)$	$(-\infty, a_2 \vartriangle b_2]$ $\cup\, [a_2 \triangledown b_1, +\infty)$	$(-\infty, a_2 \vartriangle b_2]$	$[0, +\infty)$	$(-\infty, a_2 \vartriangle b_2]$ $\cup\, [0, +\infty)$	$(-\infty, 0]$ $\cup\, [a_2 \triangledown b_1, +\infty)$	$(-\infty, 0]$	$(-\infty, +\infty)$
$(-\infty, a_2], a_2 > 0$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$
$[a_1, +\infty), a_1 < 0$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$
$[a_1, +\infty), a_1 > 0$	$\varnothing$	$(-\infty, a_1 \vartriangle b_1]$	$(-\infty, a_1 \vartriangle b_1]$ $\cup\, [a_1 \triangledown b_2, +\infty)$	$[a_1 \triangledown b_2, +\infty)$	$(-\infty, 0]$	$(-\infty, 0]$ $\cup\, [a_1 \triangledown b_2, +\infty)$	$(-\infty, a_1 \vartriangle b_1]$ $\cup\, [0, +\infty)$	$[0, +\infty)$	$(-\infty, +\infty)$
$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$	$(-\infty, +\infty)$

Table 8: Division of extended intervals with $0 \in B$ on the computer.

The rules for the operations of extended intervals on the computer in Tables 4 – 8 look rather complicated. Their implementation seems to require a major number of case distinctions. The situation, however, can be greatly simplified by the following hints.

On the computer actually only the basic rules for addition, subtraction, multiplication, and division for closed and bounded intervals of IF including division by an interval that includes zero are to be provided. In Tables 4 – 8 these rules are printed in bold letters.

The remaining rules shown in the tables can automatically be produced out of these basic rules by the computer itself if a few well established rules for computing with $-\infty$ and $+\infty$ are formally applied. With $x \in S$ these rules are

$$\begin{array}{ll} \infty + x = \infty, & -\infty + x = -\infty, \\ -\infty + (-\infty) = (-\infty) \cdot \infty = -\infty, & \infty + \infty = \infty \cdot \infty = \infty, \\ \infty \cdot x = \infty \text{ for } x > 0, & \infty \cdot x = -\infty \text{ for } x < 0, \\ \frac{x}{\infty} = \frac{x}{-\infty} = 0, & \end{array}$$

together with variants obtained by applying the sign rules and the law of commutativity. If in an interval operand a bound is $-\infty$ or $+\infty$ the multiplication with 0 is performed as if the following rules would hold

$$0 \cdot (-\infty) = 0 \cdot (+\infty) = (-\infty) \cdot 0 = (+\infty) \cdot 0 = 0.$$

These rules have no meaning otherwise.

References

[1] Alefeld, G., Herzberger, J.: *Introduction to Interval Computations*. Academic Press, New York, 1983.

[2] Kahan, W.: *A More Complete Interval Arithmetic*. Lecture Notes prepared for a summer course at the University of Michigan, June 17-21, 1968.

[3] Kirchner, R., Kulisch, U.: *Hardware support for interval arithmetic*. Reliable Computing 12:3, 225–237, 2006.

[4] Kulisch, U.,W.: *Computer Arithmetic and Validity – Theory, Implementation and Applications*. De Gruyter, Berlin, New York, 2008.

[5] IFIPWG-IEEE754R: *Letter of the IFIP WG 2.5 to the IEEE Computer Arithmetic Revision Group, 2007.*[5]

[6] Moore, R. E.: *Interval Analysis*. Prentice Hall Inc., Englewood Cliffs, New Jersey, 1966.

[7] Moore, R. E.: *Methods and Applications of Interval Analysis*. SIAM, Philadelphia, Pennsylvania, 1979.

[8] Ratz, D.: *On Extended Interval Arithmetic and Inclusion Isotony.* Preprint, Institut für Angewandte Mathematik, Universität Karlsruhe, 1999.

[9] Rump, S.M.: *Kleine Fehlerschranken bei Matrixproblemen*. Dissertation, Universität Karlsruhe, 1980.

[5] See the author's homepage.