Exact Numerical Processing

Juan Manuel García Chamizo
DTIC. University Alicante, Spain
juanma@dtic.ua.es

Jerónimo Mora Pascual
DTIC. University Alicante, Spain
jeronimo@dtic.ua.es

Higinio Mora Mora
DTIC. University Alicante, Spain
hmora@dtic.ua.es

Abstract

A model of an exact arithmetic processing is presented. We describe a representation format that gives us a greater expressive capability and covers a wider numerical set. The rational numbers are represented by means of fractional notation and explicit codification of its periodic part. We also give a brief description of exact arithmetic operations on the proposed format. This model constitutes a good alternative for the symbolic arithmetic, in special when numerical exact values are required. As an example, we show an application of the exact numerical processing to calculate the perpendicular vector to another one for aerospace purposes.

1. Introduction

Certain applications require elaborate mathematical calculations and, at the same time, impose important accuracy restrictions. For example, the calculation of trajectories for moving bodies in space or over long distances, guidance and positioning systems [1], [2], high frequency communications, antennae alignment, etc. In these applications, a slight lack of precision in the operations may cause considerable deviations in the results obtained and to cause dramatic consequences.

The use of computational algorithms raises the following important question: what is considered to be suitable calculation precision in order to solve each problem satisfactorily? Related to this subject, calculus inaccuracy is due to the limitations of current representation formats. They may cause errors in operations such as the ones described in [3], [4], [5] and [6]. On the other hand, the use of high level methods cannot be adapted to problems in which temporal cost is also an important aspect to be considered.

Our work is focused on conceiving methods to process error-free rational operands. This arithmetic model includes numerical representation and the execution of several elemental operations. This short paper describes the model and develops mainly the characteristics of the exact arithmetic architecture related to the exact representation of rational numbers. In later works, we hope to study operation methods in greater depth.

2. Architecture

In the first place, we formalized the characteristics of the arithmetical architecture in order to define the exact calculation problem.

2.1. Formalization

Let f be a generic mathematical function.

Definition 1: We define *implementation function* Γ *of* f, to any computable function whose result comes near to f according to a particular design, so that:

$$\text{codomain}(\Gamma_f) \subseteq \text{codomain}(f) \quad (1)$$

and

$$\forall\ \overline{X} \in \text{domain}(\Gamma_f),\ |\Gamma_f(\overline{X}) - f(\overline{X})| \leq \varepsilon \quad (2)$$

where $\overline{X}$:function's operands; ε: approximation degree of f by Γ_f.

The general task of an arithmetical unit is to process mathematical functions. The calculation of these functions is its ideal objective.

Definition 2: An *architecture* Λ is characterized both: by the set of functions that it provides and by the form in which it implements them. Let the following set of functions be:

$$\Phi = \{f_1, f_2, ..., f_n\} \quad (3)$$

An architecture Λ_Φ that provides these functions will be formed by:

$$\Lambda_\Phi = \{\Gamma_{f1}, \Gamma_{f2}, ..., \Gamma_{fn}\} \quad (4)$$

That is, Λ_Φ will contain the specific implementation of the Φ functions set, where each Γ_{fi} produces an approach to its f_i. Each f_i is the arithmetic unit objective, whereas each Γ_{fi} corresponds to the function that is finally provided. The function implementation does not have to be unique, several implementations of the same function that represent different approaches to f, with different values of ε can exist. It is also possible for the same architecture to contain several implementations for the same function f, giving the function several degrees of approach. For

example, the different arithmetic adder implementations for each operand size of different representation format.

Definition 3: We say that an *implementation* Γ_f *has an exact evaluation of* f if the result that Γ_f provides agrees with the mathematical result of f. That is, according to the equation (2), $\varepsilon = 0$.

$$\forall\ \overline{X} \in \text{domain}(\Gamma_f),\ \Gamma_f(\overline{X}) = f(\overline{X}) \qquad (5)$$

In this case, the Λ architecture implements that function effectively. An exact arithmetic architecture requires all its functions to be implemented in a totally effective way.

2.2. Exact arithmetic architecture

We propose a simple arithmetic architecture unit capable of performing a set of exact operations on rational numbers.

$$\Phi_{\mathbb{Q}} = \{\text{identity, addition, multiplication, root, } \ldots\} \qquad (6)$$

The architecture implements these operations in an effective way:

$$\Lambda_{\Phi\mathbb{Q}} = \{\Gamma_{\text{identity}}, \Gamma_{\text{addition}}, \Gamma_{\text{multiplication}}, \Gamma_{\text{root}}, \ldots\} \qquad (7)$$

The identity function expresses the ability to represent, and therefore to operate, rational numbers. Actually, this function corresponds to the numeric format that codifies any error-free rational number. An inverse identity function is necessary to show the results in their original way.

3. Identity function

Numeric formats are conditioned fundamentally by the characteristics of the numerical set to be represented. Different methods exist for codification for natural, integer or real numbers [7], [8], [9].

In the representation of real set, codification of rational numbers is of particular interest since it is the largest subset of $\mathbb{R}$ where the exact value of their elements can be written in a positional representation format. There are symbolic representation methods characterized by their ability to carry out an exact expression of the data they operate [8], [10], nevertheless, in several applications it is necessary to process numerical values directly and to provide a result by means of a number, instead of processing and providing symbolic expressions.

This section is focused on the representation of rational numbers by means of a representation format capable of expressing error-free values of the $\mathbb{Q}$ set in a fractional positional notation. The idea are similar to was presented in [11], even though, the progressive improvement in performance provided by electronic technology justifies the search for proposals that would probably have been prohibitive some time ago, as well as, the conception of operation methods with that representation.

3.1. Specification of the proposed format

We consider the number representation procedure to be an implementation of the identity function: Γ_{identity}

$$f_1 \equiv \text{identity}: \mathbb{Q} \to \mathbb{Q} \qquad (8)$$

$$\forall\ x \in \mathbb{Q}.\ \text{identity}(x) = x \qquad (9)$$

The inherent characteristics of the set of the rational numbers $\mathbb{Q}$ suggest the possibility of obtaining a representation model that fulfills the following objectives:

1. To contain the exact positional expression of the number it represents in a direct way.
2. To allow an indeterminate number of exact fractional digits to be obtained according to the requirements of each application.
3. To reach very high or very low extreme values.

Rational numbers permit a compact expression that defines them exactly: in fractional notation they are formed on one integer part and a finite fractional part followed by a group of digits that form a period that is repeated indefinitely, This characteristic, when the periodic part is not null, disables its exact representation in the conventional fractional code systems (fixed-point and floating-point), therefore, the explicit expression of this period permits the error-free codification of the whole number with a minimum number of significant digits.

The proposed model of representation is based on the classic floating-point format and incorporates, if necessary, a second mantissa that represents the period of the rational values. This format distributes the significant digits of the number (WL) into three parts (fig. 1): fixed mantissa wordlength (M_fWL), periodic mantissa wordlength (M_pWL) and exponent wordlength (EWL). The sign bit is included at the beginning. The fixed mantissa comprises the non-periodic significant part of the rational number, whereas the periodic mantissa represents the repetitive digits. The exponent expresses the order of magnitude of the number.

Figure 1. Proposed representation format

As an illustrative example, decimal number 2.18 has not an exact representation in IEEE-754 simple precision codification; in the proposed format, the codification of 2.18 consists of exponent: 010; fixed mantisa: 100; periodic mantisa: 01011100001010001111.

The significance will be made by concatenation of the fixed mantissa and the periodic mantissa an indefinite quantity of times (fig. 2 illustrates its construction). With this technique, numbers with infinite fractional numbers are obtained from a finite codification. Their exact representation can be obtained by computer.

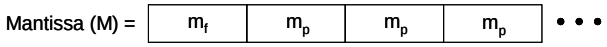

Figure 2. Mantissa construction

The value of the number is now obtained according to the same expression as in the floating point format:

$$A = (-1)^s \cdot M \cdot B^E \qquad (10)$$

where B is the base of the representation, M the complete mantissa formed by the concatenation of the fixed mantissa and the periodic mantissa infinite times and E the exponent in sign/magnitude representation.

The exact value of the complete mantissa can be obtained by means of the following expression:

$$M = \frac{m_f m_p - m_f}{\underbrace{(B-1)\cdots(B-1)}_{M_P WL}\ \underbrace{0\ \cdots\ 0}_{M_F WL}} \qquad (11)$$

where $m_f m_p$ is the concatenation of the fixed mantissa and the periodic mantissa once. In order to avoid multiple codification of the same number, it is necessary to normalize the representation.

3.2. Characteristics of the proposed format

Greater expressiveness is obtained if we use a variable size for each field according to the representation requirements, including the possibility of maintaining some field empty. We have observed that with a sufficient number of digits, the implementation $\Gamma_{identity}$ fulfills the injective and surjective properties:

Injective: Any rational value has a different codification.

Surjective: Any codification corresponds to only one rational value.

Existence of inverse: Due to the conjunction of the two previous properties, the correspondence function allows its inverse on the set $\mathbb{Q}$, that is, it is possible to construct a function $\Gamma_{identity}^{-1}$, that obtains the initial rational value again from any codification.

As a result of these properties, any standardized rational value according to the format will have a characteristic expression, composed of an exponent, one fixed mantissa and one periodic mantissa.

When limitations in the number of digits to be represented exist, the method can introduce a rounding stage that approaches the number to the nearest representable value using any IEEE floating-point rounding mode. In this situation where it is not possible to express all the number's digits, the field of the period lacks relevance and makes the most of its space to code the largest possible number of digits of the fixed mantissa.

4. Arithmetic operators

In this section we describe the ideas on which the sum and multiplication methods are based and which permit effective data processing and its practical application. The arithmetic algorithms according to number representation format $\Gamma_{identidad}$ must solve the following questions:

(a) Variable field length of the operands advises against the application of rigid process procedures; (b) existence of a periodic mantissa causes the sequence of significant digits to become infinite. (c) normalization of the results in the new format implies taking additional steps.

In answer to the previous questions, the arithmetic algorithms are based on the following principles:

1. The stages of standard floating-point operation are taken as a starting-point for the new methods [12], [13], [14].
2. The application of iterative methods that process the data successively and facilitate a parallel design.
3. The development of strategies adapted for the treatment of each significant mantissa and its later integration in obtaining the result.

Hence, the calculation methods provide the exact result of the sum and multiplication operations and the rounding stage is avoided. The length of the positional fractional representation of the exact result is proportional to the initial size of the operands, being feasible the design of strategies to manage the precision and the result length.

It is necessary to conceive a memory unit that allows us to lodge data of variable length. It avoids the rigidities of the fixed-size registers and facilitates the manipulation of a number with a variable number of digits.

5. Application example

The growth of the significant digits of the computation results of this exact processing method recommends its application in critical calculations with a limited number of consecutive operations and where an exact or very precise result is essential. In this section we show a representative exact numerical processing application example.

The experiment is located in the *Communications Systems and Telematic Applications* research group of the *University of Alicante.* One of the most interesting lines of research consists of the development of a distance calculation method by means of a multifrequency technique [15], [16].The accuracy of the calculations plays a fundamental role in the correct working of the method.

Let us give a simple example: the orientation of the antenna is a critical aspect in the emission and reception of signals, even more so when the objective is the positioning and direction of an object. A geostationary satellite needs to orient its antenna perpendicularly to its movement vector $\vec{s}$ (fig. 3): $\vec{s} \perp \vec{r}$ where $\vec{s} = (x_s, y_s, z_s)$ is the satellite movement vector and $\vec{r} = (x_r, y_r, z_r)$ is the antenna orientation vector.

The following expressions calculate the vector $\vec{r}$ components by means of additions and multiplications:

$$x_r = x_s \cdot z_s;\ y_r = y_s \cdot z_s;\ z_r = -(x_s^2 + y_s^2) \qquad (12)$$

A slight lack of precision in the calculations of the vector direction causes deviations in the alignment of the antenna, so that the angle is not perfectly perpendicular: $\pi/2 \pm\varepsilon$, (fig. 3). This problem causes a deviation of the signal towards a zone around the desired point. The error worsens as a result of atmospheric conditions that affect the angle of incidence, for example, refraction. According to Snell's Law:

$$n_1 \cdot \sin \alpha_1 = n_2 \cdot \sin \alpha_2 \quad (13)$$

A completely perpendicular signal will not be affected by atmospheric refraction, since,

$$\sin 0 = 0 \Rightarrow \forall\, n_1, n_2.\ \alpha_1 = \alpha_2 \quad (14)$$

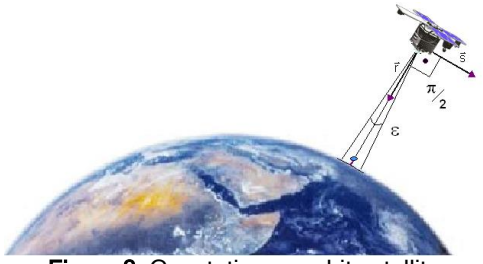

Figure 3. Geostationary orbit satellite

Nevertheless, a wrong angle $\alpha_1 \neq 0$, will cause a greater deviation in the signal, especially when the satellite is located at a considerable distance. As a result, a slight lack of precision in the calculations of the vector componentes will cause great deviations in the alignment of the antenna with the target. The following numerical example provides proof of that: let $\vec{s}$ be the movement vector of a geostationary satellite (orbit located between 35000 km and 37000 km),

$$\vec{s} = (-5.69, -0.07, -5.71)$$

The perpendicular vector to $\vec{s}$ according to (12) equation with the proposed arithmetic is:

$$\vec{r}_1 = (32.4899, 0.3997, -32.381)$$

Same calculation result, using simple precision IEEE-754 format and arithmetic operators, give us:

$$\vec{r}_2 = (32.48989868, 0.39970001578, -32.381004334)$$

We verified that

$$\vec{s} \cdot \vec{r}_1 = 0;\ \vec{s} \cdot \vec{r}_2 = 3.162 \cdot 10^{-5} \neq 0.$$

Despite of the deviation being small, it produces an error in the Earth's surface of ±15.39m. A direct incidence antenna capable of being in tune with the satellite signal must have a surface of 744.25 m^2. It is possible to reduce this error by using more fractional digits, for example double precision.

6. Conclusions

In this work a new arithmetic model on rational numbers is presented. The design methodology has been oriented towards providing exact calculation results. The arithmetic architecture has been provided with several elementary operations.

The identity operation performs numerical representation of rational numbers. Its main characteristic is that, with a sufficient number of finite bits, it enables any rational number to be error-free represented. The proposed format uses a fractional numerical representation that directly expresses the magnitude of the number and allows its value to be known immediately. Its floating-point characteristics, together with the variable size of the fields, provide flexibility and an adjustment capacity to meet the requirements of each problem. These qualities allow us to codify rational numbers and to overcome the errors made in decimal-binary conversion in human interaction. It offers an alternative to symbolic calculation for exact processing

The addition and multiplication operations acquire the same exact processing properties from operating with numbers expressed in the proposed format. The experiments and the application example clearly show the disadvantages of standard representation formats and demonstrate the need for exact calculations in critical operations.

7. References

[1] European Space Agency, http: //www.esrin.esa.it/htdocs/tidc/Press/Press96/ariane5rep.html, "ARIANE 5 Failure". 1996.

[2] US General Accounting OR,http://www.fas.org/spp/starwars/gao/im92026.htm, Patriot-Missile-Defence-Software, 1992.

[3] M.J. Schulte, "A Family of Variable-Precision Interval Arithmetic Processors", *IEEE Transactions on Computers*,Vol.49, no.5, 2000.

[4] D. Michelucci, J-M. Moreau, "Lazy Arithmetic", *IEEE Transactions on Computers*, Vol . 46, no. 9, pp. 961-975, Septiembre 1997.

[5] G. Bohlender, "What Do We Need Beyond IEEE Arithmetic?", C. Ullrich, ed., Boston: Academic Press, 1990, pp 1-32.

[6] C.M. Hoffmann. "The Problems of Accuracy and Robustness in Geometric Computation", *IEEE Computer*, vol. 22, no. 3, 1989.

[7] D.A. Patterson, J.L. Hennessy, "Computer Architecture a quantitative approach", *Morgan Kaufmann Publishers*, 2002.

[8] A. M. Nielsen, "Number systems and Digit Serial Arithmetic", *PhD Thesis*, Odense University, 1997.

[9] P. Kornerup, "Digit-Set Conversions: Generalizations and Applications", *IEEE Transactions on Computers*, vol. 43, no. 5, 1994.

[10] C. D. Moore, "An Introduction to Continued Fractions". *The National Council of Teachers of Mathematics*, 1964

[11] E. Hehner, R. Horspool, "A New Representation of the Rational Numbers for fast Easy Arithmetic", SIAM J. Comp., vol. 8, no. 2, 1979.

[12] S.F. Oberman, "Design Issues in High Performance Floating Point Arithmetic Units", *T. R.* CSL-TR-96-711, Stanford University, 1996.

[13] N.T. Quach, M.J. Flynn, "An improved Algorithm for High-Speed Floating-Point Addition", *T.R.* CSL-TR-90-442, Stanford U., 1990.

[14] G.W. Bewick. "Fast multiplication: Algorithm and implementation". *PhD Thesis*, Stanford University, 1994.

[15] F. Pujol, F.J. Ferrández, J.M. García. "A new Method for Position Location in Random Media", 8th Inter. Conf. on Electromagnetics of Complex Media, Lisboa (Portugal), 2000.

[16]F.J. Ferrández, F. Pujol, J.M. García "Método para la determinación de distancias mediante técnica multifrecuencial", Int. Conf. of Telecom. and Electronics, Santiago de Cuba (Cuba), 2000.