Matryoshka arithmetic

Joris van der Hoeven, Grégoire Lecerf and Arnaud Minondo

Laboratoire d'Informatique de l'École Polytechnique (LIX, UMR 761)
CNRS, École polytechnique, Institut Polytechnique de Paris
Batiment Alan Turing, CS35003
1 rue Honoré d'Estienne d'Orves
91120 Palaiseau, France
{vdhoeven,lecerf,minondo}@lix.polytechnique.fr
Preliminary version soon available on Hal [**?**]

Abstract

Interval arithmetic achieves numerical reliability for a wide range of applications, at the price of a performance penalty. For applications from differential equations to homotopy continuation, one key ingredient is the efficient and reliable evaluation of complex polynomials represented by straight-line programs (SLPs). This is best achieved using ball arithmetic, a variant of interval arithmetic. In this article, we introduce new variants of ball arithmetic. For the evaluation of SLPs, this allows us to almost eliminate the overhead of ball arithmetic with respect to direct numerical evaluations.

Keywords: ball arithmetic, straight-line program, polynomial evaluation, reliable computing

1 Introduction

Interval arithmetic is a popular technique to calculate guaranteed error bounds for approximate results of numerical computations [4, 5]. The idea is to systematically replace floating point approximations by small intervals around the exact numbers we are interested in. Basic arithmetic operations on floating point numbers are replaced accordingly with the corresponding operations on intervals. When computing with complex numbers or when working with multiple precision, it is more convenient to use balls instead of intervals. In this paper, we will always do so and this variant of interval arithmetic is called *ball arithmetic* [2].

Unfortunately, ball arithmetic suffers from a non-trivial overhead: floating point balls take twice the space of floating point numbers and basic arithmetic operations are between two and approximately ten times more expensive. For certain applications, it may therefore be preferable to avoid the systematic use of balls for individual operations. Instead, one may analyze the error for larger groups of operations.

The goal is to compute reliable error bounds in a systematic fashion, while avoiding the overhead of ball arithmetic. We will focus on the case where the

function f we wish to evaluate is given by a straight-line program (SLP). Such a program is essentially a sequence of basic arithmetic instructions like additions, subtractions, multiplications, and possibly divisions [2]. The SLP framework is actually surprisingly general: at least conceptually, the trace of the execution of a more general program that involves loops or subroutines can often be regarded as an SLP[2].

2 Definitions

2.1 IEEE floating-point arithmetic and notations

Throughout this abstract, we assume that we work with a fixed floating point format that conforms to the IEEE 754 standard. We write p for the bit precision, i.e. the number of fractional bits of the mantissa plus one. We denote the set of hardware floating point numbers by $\mathbb{R}_p$. Given an $\mathbb{R}$-algebra $\mathbb{A}$, we will also denote the corresponding approximate version by $\mathbb{A}_p$. For instance, if $\mathbb{A} = \mathbb{C} = \mathbb{R}[\mathrm{i}]$, then we have $\mathbb{A}_p = \mathbb{R}_p[\mathrm{i}]$.

The IEEE 754 standard imposes correct rounding of all basic arithmetic operations. In this paper we will systematically use the *rounding to nearest* mode. We denote by $x_\circ$ the result of rounding $x \in \mathbb{R}$ according to this mode. The quantity $\varepsilon_\circ(x) := |x_\circ - x|$ stands for the corresponding rounding error, which may be $+\infty$. Given a single operation $* \in \{+, -, \cdot, \ldots\}$, we write $x *_\circ y$ for $(x * y)_\circ$. For compound expressions φ, we will also write $\circ[\varphi]$ for the full evaluation of φ using the rounding mode $\circ$. For instance, $\circ[xy + a^2 b] = x_\circ \cdot_\circ y_\circ +_\circ (a_\circ \cdot_\circ a_\circ) \cdot_\circ b_\circ$.

We denote by $\bar{\varepsilon}_\circ$ any upper bound function for $\varepsilon_\circ$ that is easy to compute. In absence of underflow, one may take $\bar{\varepsilon}_\circ(x) = |x_\circ|\, 2^{-p}$. We define SLPs the same way done in [1]. We will denote by $\lg k := \lceil \log_2(k) \rceil$ for all integers $k \geqslant 1$.

2.2 Ball arithmetic

Let $\mathbb{A}$ be an $\mathbb{R}$-algebra and let $\| \cdot \|$ be a norm on $\mathbb{A}$. We will typically take $\mathbb{A} = \mathbb{R}$ or $\mathbb{A} = \mathbb{C}$, but more general normed algebras are also allowed. Given $c \in \mathbb{A}$ and $r \in \mathbb{R}$, let $\mathcal{B}(c, r) := \{z \in \mathbb{A}, \|z - c\| \leqslant r\}$ be the closed ball with center c and radius r. We denote by $\mathcal{B}(\mathbb{A}, \mathbb{R})$ the set of all such balls. We can extend the norm from $\mathbb{A}$ to $\mathcal{B}(\mathbb{A}, \mathbb{R})$ via

$$\begin{aligned} |\cdot| : \mathcal{B}(\mathbb{A}, \mathbb{R}) &\longrightarrow \mathbb{R} \\ \mathcal{B}(c, r) &\longmapsto |c| + r \end{aligned} \tag{1}$$

which remains subadditive, submultiplicative, and positive definite. We introduce a separate notation $\circ\!\!-$ for this type of semantics: given a ball $\boldsymbol{x} \in \mathcal{B}(\mathbb{A}, \mathbb{R})$ and a number, we say that $\boldsymbol{x}$ *encloses* x if $\boldsymbol{x} \ni x$, i.e.

$$\boldsymbol{x} \circ\!\!- x \quad \Longleftrightarrow \quad x \in \boldsymbol{x}.$$

For all $a := (a_1, \ldots, a_m) \in \mathbb{A}^m$ and all $r := (r_1, \ldots, r_m) \in \mathbb{R}^m$, we denote the poly-ball $\mathcal{P}(a, r) := (\mathcal{B}(a_1, r_1), \ldots, \mathcal{B}(a_m, r_m)) \in \mathcal{B}(\mathbb{A}, \mathbb{R})^m$. We extend this

enclosure relation to poly-balls $\boldsymbol{x} \in \mathcal{B}(\mathbb{A},\mathbb{R})^m$ and $x \in \mathbb{R}^m$ as follows:

$$\boldsymbol{x} \circ\!\!- x \quad \Longleftrightarrow \quad \boldsymbol{x}_1 \circ\!\!- x_1 \wedge \cdots \wedge \boldsymbol{x}_m \circ\!\!- x_m. \tag{2}$$

Given a function $f : \mathbb{A}^m \mapsto \mathbb{A}^n$, a *ball lift* of f is a function $f : \mathcal{B}(\mathbb{A},\mathbb{R})^m \mapsto \mathcal{B}(\mathbb{A},\mathbb{R})^n$ that satisfies the *inclusion property*

$$\boldsymbol{x} \circ\!\!- x \quad \Longrightarrow \quad f(\boldsymbol{x}) \circ\!\!- f(x)$$

for all $\boldsymbol{x} \in \mathcal{B}(\mathbb{A},\mathbb{R})^m$ and $x \in \mathbb{A}^m$.

Example 1 Let $a, b \in \mathbb{A}$ and $r, s \in \mathbb{R}$, basic arithmetic operations admit the following lift:

$$\begin{aligned}
\mathcal{B}(a,r) + \mathcal{B}(b,s) &= \mathcal{B}(a+b, r+s) \\
\mathcal{B}(a,r) - \mathcal{B}(b,s) &= \mathcal{B}(a-b, r+s) \\
\mathcal{B}(a,r) \times \mathcal{B}(b,s) &= \mathcal{B}(a \times b, (|a|+r)s + |b|r).
\end{aligned}$$

2.3 Matryoshka

Let $\mathcal{B}(\mathbb{A},\mathbb{R},\mathbb{R}) := \mathcal{B}(\mathcal{B}(\mathbb{A},\mathbb{R}),\mathbb{R})$. An element $\mathcal{B}(A,R,r) := \mathcal{B}(\mathcal{B}(A,R),r)$ of $\mathcal{B}(\mathbb{A},\mathbb{R},\mathbb{R})$ will be called a *matryoshka*. Given a matryoshka $\boldsymbol{A} = \mathcal{B}(A,R,r)$, we call A its *center*, R its *large radius*, and r its *small radius*. The enclosure relation for matryoshki is defined as follows: given a matryoshka $\boldsymbol{A} = \mathcal{B}(A,R,r)$ and a ball $\boldsymbol{a} = \mathcal{B}(a,s)$, we define

$$\boldsymbol{A} \circ\!\!- \boldsymbol{a} \quad \Longleftrightarrow \quad \mathcal{B}(A,R) \circ\!\!- a \text{ and } s \leqslant r.$$

We use the abbreviation $\mathcal{B}(\mathbb{A},\mathbb{R},\mathbb{R}) := \mathcal{B}(\mathcal{B}(\mathbb{A},\mathbb{R}),\mathbb{R})$ for the set of matryoshki. Given a function $f : \mathcal{B}(\mathbb{A},\mathbb{R})^m \to \mathcal{B}(\mathbb{A},\mathbb{R})^n$, a *matryoshka lift* of f is a function $f : \mathcal{B}(\mathbb{A},\mathbb{R},\mathbb{R})^m \to \mathcal{B}(\mathbb{A},\mathbb{R},\mathbb{R})^n$ that satisfies the inclusion principle:

$$\boldsymbol{A} \circ\!\!- \boldsymbol{a} \quad \Longrightarrow \quad f(\boldsymbol{A}) \circ\!\!- f(\boldsymbol{a})$$

for all $\boldsymbol{A} \in \mathcal{B}(\mathbb{A},\mathbb{R},\mathbb{R})^m$ and $\boldsymbol{a} \in \mathcal{B}(\mathbb{A},\mathbb{R})^n$. We will write $\mathcal{B}(\mathbb{A}_p,\mathbb{R}_p)$ and $\mathcal{B}(\mathbb{A}_p,\mathbb{R}_p,\mathbb{R}_p)$ for the approximate versions of $\mathcal{B}(\mathbb{A},\mathbb{R})$ and $\mathcal{B}(\mathbb{A},\mathbb{R},\mathbb{R})$.

Example 2 Basic operations admit following lift:

$$\begin{aligned}
\mathcal{B}(\boldsymbol{a},r) + \mathcal{B}(\boldsymbol{b},s) &= \mathcal{B}(\boldsymbol{a}+\boldsymbol{b}, r+s) \\
\mathcal{B}(\boldsymbol{a},r) - \mathcal{B}(\boldsymbol{b},s) &= \mathcal{B}(\boldsymbol{a}-\boldsymbol{b}, r+s) \\
\mathcal{B}(\boldsymbol{a},r) \times \mathcal{B}(\boldsymbol{b},s) &= \mathcal{B}(\boldsymbol{a} \times \boldsymbol{b}, (|\boldsymbol{a}|+r)s + |\boldsymbol{b}|r).
\end{aligned}$$

Those formulas correspond exactly with formulas of the example 1 where this time, centers are balls.

3 Main Results

The following propositions highlight the interest of matryoshki. Up to a precomputation on a large domain, with matryohski, we are able to speed up ball arithmetic on that domain. Let Γ be an SLP that computes a function $f : \mathbb{A}^m \to \mathbb{A}^n$. Let us denote by $f_\circ : \mathbb{A}_p^m \to \mathbb{A}_p^n$ the function that we obtain by evaluating Γ using floating point arithmetic over $\mathbb{A}_p$. We control rounding errors happening in the evaluation of Γ:

Proposition 1 *Let $\boldsymbol{A} = (\boldsymbol{A}_1, \ldots, \boldsymbol{A}_m) \in \mathcal{B}(\mathbb{A}_p, \mathbb{R}_p)^m$ be a fixed poly-ball and $\mathcal{P}(\boldsymbol{A}, 0) := (\mathcal{B}(\boldsymbol{A}_1, 0), \ldots, \mathcal{B}(\boldsymbol{A}_m, 0)) \in \mathcal{B}(\mathbb{A}_p, \mathbb{R}_p, \mathbb{R}_p)^m$. Let $\boldsymbol{F}$ be a matryoshka lift of f and $(\mathcal{B}(\boldsymbol{B}_1, E_1), \ldots, \mathcal{B}(\boldsymbol{B}_n, E_n)) := \boldsymbol{F}(\mathcal{B}(\boldsymbol{A}, 0))$. Then for all $a \in \mathbb{A}_p^m$ with $\boldsymbol{a} \circ\!\!-\, a$, we have*

$$|f_{\circ,i}(a) - f_i(a)| \leqslant E_i.$$

Proposition 1 leads to a ball lift up to precomputing bounds $(B_{i,j})_{1 \leq i \leq m, 1 \leq j \leq n}$ for the Jacobian matrix:

$$\left\| \frac{\partial f_i}{\partial x_j} \right\|_{\boldsymbol{A}} := \sup_{\boldsymbol{A} \circ\!\!-\, a} \left| \frac{\partial f_i}{\partial x_j}(a) \right| \leqslant B_{i,j}. \tag{3}$$

For this, it suffices to evaluate a ball lift of the Jacobian of f at $\boldsymbol{A}$, which yields a matrix $\boldsymbol{J} \in \mathcal{B}(\mathbb{A}_p, \mathbb{R}_p)^{n \times m}$, after which we take $B_{i,j} := |\boldsymbol{J}_{i,j}|$. We now have:

Proposition 2 *Assume the above notation and let E be as in Proposition 1. Assume also that $2 \lg^2 m < \epsilon^{-1}_{\mathbb{A}_p, \mathbb{R}_p}$. For every $\boldsymbol{a} = \mathcal{P}(a, r) \in \mathcal{B}(\mathbb{A}_p, \mathbb{R}_p)^m$ with $\boldsymbol{a}_1 \subseteq \boldsymbol{A}_1, \ldots, \boldsymbol{a}_m \subseteq \boldsymbol{A}_m$, let*

$$\boldsymbol{f}_*(\boldsymbol{a}) := \mathcal{P}\left(f_\circ(a), \circ[((E_1, \ldots, E_n)^\top + Br)(1 + (\lg m + 6)\epsilon) + 3\eta]\right).$$

Then $\boldsymbol{f}_*$ defines a ball lift of f. (Note that this lift is only defined on the domain of balls $\boldsymbol{a}$ that satisfy the above restrictions.)

References

[1] P. Bürgisser, M. Clausen and M. Amin Shokrollahi. Algebraic Complexity Theory. *Springer*.

[2] Joris van der Hoeven. Ball arithmetic. *Logical approaches to barriers in complexity*, pages 179-208, 2010.

[3] Joris Van der Hoeven, Grégoire Lecerf and Arnaud Minondo. Static Bounds for Straight-Line Programs. 2025, *Hal-Science*.

[4] R. E. Moore. Interval Analysis. *Prentice Hall*, Englewood Cliff, 1966.

[5] R. E. Moore, R. B. Kearfott, and M. J. Cloud. Introduction to Interval Analysis. SIAM Press, 2009.