Probabilistic estimation of the accuracy of inner products and application to stochastic validation

Fabienne Jézéquel, Théo Mary
LIP6, Sorbonne Université, CNRS, Paris, France

Interflop project meeting
13-14 June 2024

Accuracy of inner products

The inner product $s = x^T y$ of $x, y \in \mathbb{R}^n$ is a widely used kernel.

The computed $\widehat{s}$ satisfies

$$\frac{|\widehat{s} - s|}{|s|} \leq nu \frac{|x|^T |y|}{|x^T y|}$$

where u is the unit roundoff (in double precision $u = 2^{-53}$).

But this bound cannot be used as a reliable estimator of the accuracy.

- nu (the backward error) is often **pessimistic**
- $\kappa := \frac{|x|^T |y|}{|x^T y|}$ (the condition number) is in general sharp but **cannot be reliably computed**: it requires the true inner product $x^T y$.

$\Rightarrow$ numerical validation based on **stochastic arithmetic**

Stochastic arithmetic

For each arithmetic operation $c = a \,\mathrm{op}\, b$ where $\mathrm{op} \in \{+, -, \times, /\}$:

$$c^{(1)} = \mathrm{SR}(a^{(1)} \,\mathrm{op}\, b^{(1)})$$
$$\ldots$$
$$c^{(d)} = \mathrm{SR}(a^{(d)} \,\mathrm{op}\, b^{(d)})$$

where the stochastic rounding operator $\mathrm{SR}(\cdot)$ rounds either up or down at random with equal probability.

computed result: $\bar{c} = \frac{1}{d} \sum_{i=1}^{d} c^{(i)}$

its number of correct digits is estimated as

$$D_c = \log_{10}\left(\frac{\sqrt{d}\,|\bar{c}|}{\sigma \tau_\beta}\right) \text{ where } \sigma^2 = \frac{1}{d-1} \sum_{i=1}^{d} \left(c^{(i)} - \bar{c}\right)^2$$

τ_β is the value of Student's distribution for $d-1$ degrees of freedom and a confidence level β.

Inner product in stochastic arithmetic

The computation of an inner product $s = x^T y$, $x, y \in \mathbb{R}^n$:

$$s_0 = 0,$$
$$s_k = s_{k-1} + x_k y_k, \quad \text{for } k = 1: n,$$
$$s = s_n$$

becomes with stochastic arithmetic:

$$s_0^{(i)} = 0, \quad \text{for } i = 1: d,$$
$$s_k^{(i)} = \mathrm{SR}(s_{k-1}^{(i)} + \mathrm{SR}(x_k^{(i)} y_k^{(i)})), \quad \text{for } k = 1: n \text{ and } i = 1: d,$$
$$s^{(i)} = s_n^{(i)}, \quad \text{for } i = 1: d.$$

SR applied after each addition and multiplication
prevents the use of optimized libraries which do not support SR

⇒ **major performance hurdle**

In this talk

We present 2 methods to get a reliable accuracy estimation by **standard deterministic inner products**.

randomness introduced to estimate the accuracy:

- random perturbations to the input x and/or y
- or random perturbations to the output $\widehat{s}$.

⇒ **no intrusive modifications** in the intermediate computations of the inner products

⇒ can rely on **optimized implementations** such as the BLAS.

Overview

1. Principles of the 2 methods
2. Probabilistic analysis of their accuracy estimation
3. Numerical experiments and comparison with CADNA

Method 1

We randomly perturb each representative of x and compute

$$s^{(1)} = (x^{(1)} + \Delta x^{(1)})^T y, \quad |\Delta x^{(1)}| \le \delta |x^{(1)}|,$$
$$\dots$$
$$s^{(d)} = (x^{(d)} + \Delta x^{(d)})^T y, \quad |\Delta x^{(d)}| \le \delta |x^{(d)}|,$$

where $\Delta x^{(1)}, \dots, \Delta x^{(d)}$ are random perturbations.

the perturbed representatives of x differ by a factor of order δ
$\Rightarrow$ the $s^{(i)}$ will differ by a factor of order $\kappa\delta$.

Method 1 **implicitly** estimates the condition number κ.

Method 2

We compute a deterministic inner product $\widehat{s}$.

Then we compute $\widehat{\kappa}$, an explicit estimation of κ.

Randomness introduced in the d representatives of s:

$$s^{(i)} = \widehat{s} + \Delta s^{(i)}, \quad i = 1 : d,$$

where $|\Delta s^{(i)}| \approx \delta \widehat{\kappa} |\widehat{s}|$, δ controlling the size of the perturbations.

Method 2 **explicitly** estimates κ, and randomizes the *output* with a perturbation of size $\kappa\delta$.

Probabilistic analysis of the accuracy estimation

Method 1: input randomization

Given $x, y \in \mathbb{R}^n$, we define perturbed vectors $x^{(1)}, \ldots x^{(d)}$:

$$x^{(1)} = x \circ (1 + \delta\xi^{(1)}),$$
$$\ldots$$
$$x^{(d)} = x \circ (1 + \delta\xi^{(d)}),$$

where $\circ$ denotes the Hadamard (componentwise) product, $\delta > 0$, and $\xi^{(i)} \sim \mathcal{N}(0,1)^n$ (standard normal random vectors).

We compute the d inner products $s^{(i)} = (x^{(i)})^T y$.

We show that:

1. $$\varepsilon = \frac{\sigma_s}{\sqrt{d(d-1)}} \quad \text{where} \quad \bar{s} = \frac{1}{d}\sum_{i=1}^{d} s^{(i)} \quad \text{and} \quad \sigma_s^2 = \sum_{i=1}^{d} (\bar{s} - s^{(i)})^2$$
   is a good estimator of the accuracy $|s - \bar{s}|$ of $\bar{s}$.
2. the rounding errors in the computed $\widehat{\bar{s}}$ do not significantly affect the quality of the estimator as long as $\delta \gg u$.

Method 1: accuracy $|s - \bar{s}|$ of the exact $\bar{s}$

We show that

$$t = \frac{\bar{s} - s}{\varepsilon}$$

follows Student's t-distribution with $d - 1$ degrees of freedom.

Therefore, we can bound the quality of the estimation.

For any probability $\beta > 0$, $\exists \tau_\beta$ s.t.

$$\beta = \mathbb{P}(|t| \leq \tau_\beta) = \mathbb{P}\left(|\bar{s} - s| \leq \tau_\beta \varepsilon\right)$$

For Student's distribution, small values of τ_β suffice to make β close to 1.

$\Rightarrow$ ε is a good estimator of the accuracy of $\bar{s}$.

Method 1: accuracy $|s-\widehat{s}|$ of the computed $\widehat{s}$

We show that

for any probability $\beta > 0$

$$\mathbb{P}(|s-\widehat{s}| \le 2\tau_\beta \varepsilon) \ge \beta\beta'$$

where

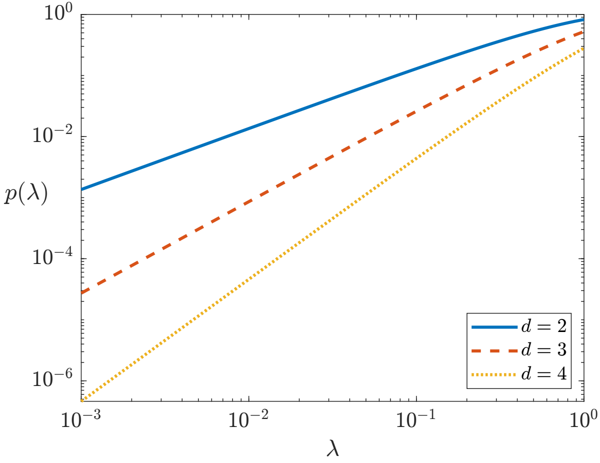
$$\beta' \ge 1 - p\left(\left(\frac{\sqrt{nd(d-1)}}{\tau_\beta}\frac{\gamma_{n+d}}{\delta}\right)^2\right)$$

for any integer k s.t. $ku < 1$, $\gamma_k = \frac{ku}{1-ku} \approx ku$

and for any $\lambda \le d-1$,

$$p(\lambda) = \left(\frac{\lambda}{d-1}\right)^{(d-1)/2} \exp\Big((d-1-\lambda)/2\Big).$$

Method 1: as closer look at $p(\lambda)$



$p(\lambda)$ quickly vanishes as λ decreases, even for small values of d.

$\Rightarrow$ $\beta' \geq 1 - p(\lambda)$ very close to 1 if λ, and so u/δ, is sufficiently small.

Method 2: output randomization

Given $x, y \in \mathbb{R}^n$, we first compute in deterministic arithmetic the inner products

$$\widehat{s} = x^T y \quad \text{and} \quad \widehat{r} = |x|^T |y|.$$

Then, we compute an estimate of the condition number

$$\widehat{\kappa} = \widehat{r} / |\widehat{s}|.$$

Finally, we randomize the output by defining its d representatives as

$$s^{(i)} = \widehat{s}(1 + \xi^{(i)} \delta \widehat{\kappa}),$$

where $\xi^{(i)} \sim \mathcal{N}(0,1)$ and $\delta > 0$ is an estimation of the numerical noise introduced in the previous computation.

Method 2 estimates the accuracy of the inner product as $\delta \widehat{\kappa}$.

Method 2: output randomization

- if x, y are known exactly, we should set $\delta \approx u$.
  Method 2 only requires the computation of two inner products (s and r) with deterministic arithmetic.

- if x, y are the result of a previous computation, we must estimate the noise δ affecting them.

  With a stochastic validation tool, we have d representatives $x^{(i)}, y^{(i)}$.

  We estimate their noise δ and use Method 2 on an arbitrary choice of representatives, or possibly on

$$\bar{x} = \frac{1}{d}\sum_{i=1}^{d} x^{(i)}, \ \bar{y} = \frac{1}{d}\sum_{i=1}^{d} y^{(i)}.$$

Method 2: output randomization

Assuming δ to be a reliable measure of the noise, the loss of accuracy is bounded by $\delta\kappa$ where $\kappa = r/|s|$ is the true condition number.

Is $\delta\widehat{\kappa} = \delta\widehat{r}/|\widehat{s}|$ a reliable estimate?

We show that

$$\kappa\frac{(1-\gamma_n)}{(1+\gamma_n\kappa)} \leq \widehat{\kappa} \leq \kappa\frac{(1+\gamma_n)}{(1-\gamma_n\kappa)}.$$

$\Rightarrow$ $\widehat{\kappa}$ is a reliable estimate of κ as long as $\gamma_n\kappa \ll 1$.

If $\gamma_n\kappa \approx 1$, all digits of the result should be lost to numerical noise.

Numerical experiments and comparison with CADNA

The CADNA library

cadna.lip6.fr

- implements stochastic arithmetic with $\beta = 95\%$ and $d = 3$
  $\Rightarrow$ estimates the number of correct digits within a 95% confidence interval
- can be used in C/C++ or Fortran codes
- provides stochastic types (3 floating-point variables and an integer)
- all operators and mathematical functions overloaded
  $\Rightarrow$ few modifications in user programs
- support for MPI, OpenMP, GPU codes
- in one CADNA execution: accuracy of any result, complete list of numerical instabilities

[Chesneaux'90], [Jézéquel & al'08], [Lamotte & al'10], [Eberhart & al'18], [Jézéquel & al'21],...

Experimental setting

We compute in double precision 200 pairs of vectors x and y of size $n = 100$.

To simulate previous errors, we generate a triplet of perturbed vectors

$$\mathbf{x} = \left(x^{(1)}, x^{(2)}, x^{(3)}\right) = \left(x + \Delta x^{(1)}, x + \Delta x^{(2)}, x + \Delta x^{(3)}\right)$$

where $|\Delta x^{(j)}| \leq \eta |x|$.

η represents the noise affecting the vectors

if $\eta = 0$ the vectors are exact

experiments (not shown here) where y was also perturbed
$\Rightarrow$ similar conclusions

Implementation of Method 1

- if $\eta = 0$ (exact vector), the triplet $\mathbf{x}$ consists of 3 identical copies of x so randomness must be added. We thus define

$$\widetilde{\mathbf{x}} = \left(\widetilde{x}^{(1)}, \widetilde{x}^{(2)}, \widetilde{x}^{(3)}\right) = \left(x + \Delta x^{(1)}, x + \Delta x^{(2)}, x + \Delta x^{(3)}\right)$$

  where $|\Delta x^{(j)}| \leq \delta |x|$.

- if $\eta \neq 0$ ($\mathbf{x}$ affected by noise), we define $\widetilde{\mathbf{x}} = \mathbf{x}$ ($\delta = \eta$).

Method 1 computes $\widetilde{\mathbf{x}}^T y$, producing a stochastic number

$$\mathbf{s}_{M_1} = \left((\widetilde{x}^{(1)})^T y,\ (\widetilde{x}^{(2)})^T y,\ (\widetilde{x}^{(3)})^T y\right)$$

where each inner product $(\widetilde{x}^{(i)})^T y$ is computed with deterministic arithmetic.

Implementation of Method 2

We compute the inner product $s^{(1)} = (x^{(1)})^T y$ with deterministic arithmetic.

We estimate the condition number as

$$\widehat{\kappa} = \frac{|x^{(1)}|^T |y|}{|s^{(1)}|}.$$

We finally compute the stochastic triplet

$$\mathbf{s}_{M_2} = (s^{(1)}, s^{(1)} + \Delta s^{(2)}, s^{(1)} + \Delta s^{(3)})$$

where $|\Delta s^{(j)}| \leq \delta \widehat{\kappa} |s^{(1)}|$ for a given δ.

Similarly to Method 1,

- if $\eta = 0$ (exact vector), we test various values of $\delta \geq u$
- if $\eta \neq 0$ (vector affected by noise), we take $\delta = \eta$

Comparison of CADNA, Method 1, and Method 2

CADNA computes $\mathbf{x}^T y$ with stochastic arithmetic and thus produces a stochastic number

$$\mathbf{s}_C = \left(s_C^{(1)}, s_C^{(2)}, s_C^{(3)}\right).$$

For CADNA, Method 1, and Method 2, we report:

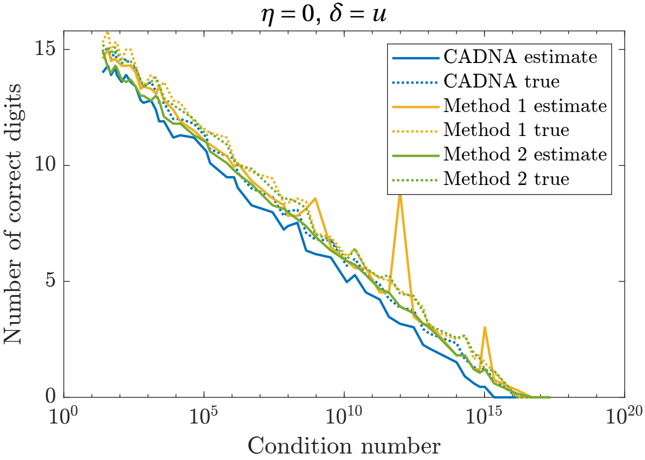
- the *estimated* accuracy provided by the method
- the *true* accuracy obtained by comparing the computed result to the correctly rounded result.

The accuracy is measured as the number of correct decimal digits of the result.

Since the computed result $\mathbf{s} = (s^{(1)}, s^{(2)}, s^{(3)})$ is a stochastic number,

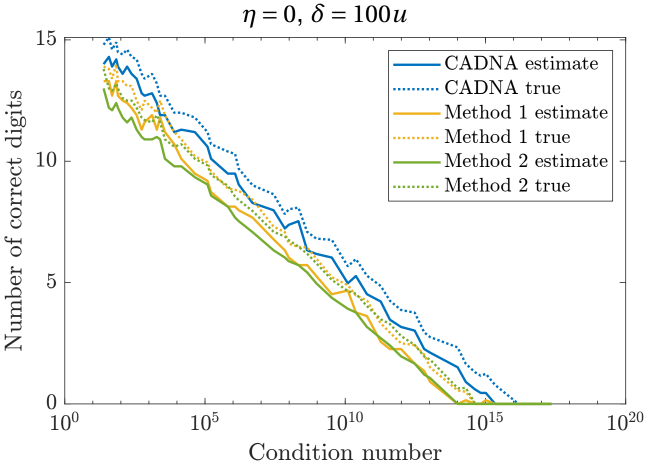
- the estimated accuracy is obtained using a CADNA function
- the true accuracy is measured using the average result $\bar{s} = \sum_{i=1}^{3} s^{(i)}/3$.

Exact input vectors



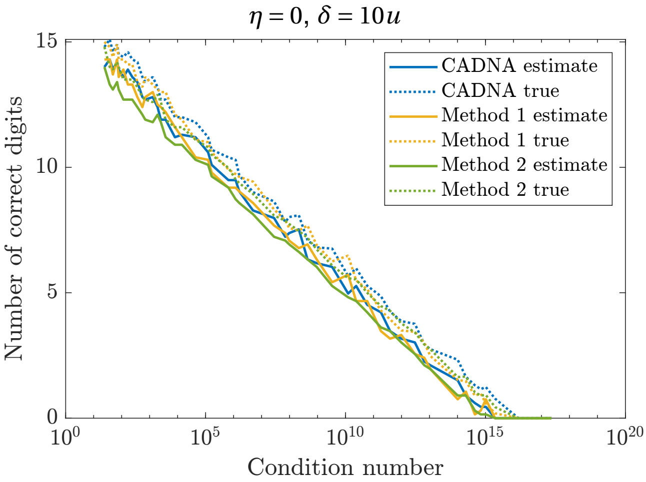
For a too small δ, Method 1 (and Method 2) can overestimate the number of correct digits.

Exact input vectors



For a too large δ, the estimation is reliable but the computed result is noticeably less accurate than with CADNA, due to the introduction of an error $\approx \delta$.

Exact input vectors



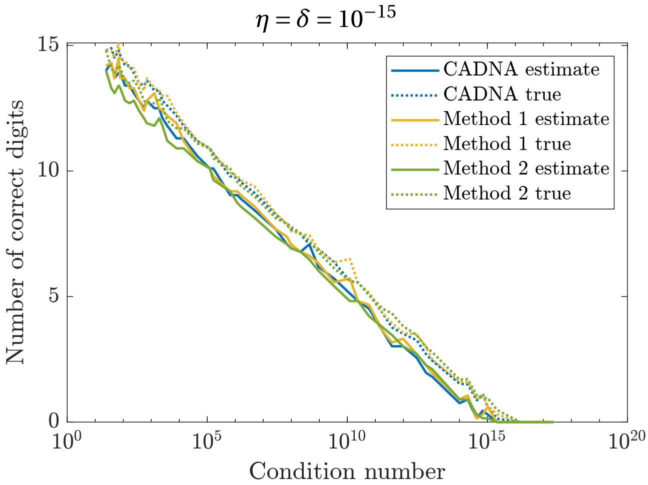
$\delta = 10u$ appears to be a suitable choice.
Method 1 and Method 2 compute a result with comparable accuracy to CADNA, while providing a reasonably tight estimate of their accuracy.

Exact input vectors

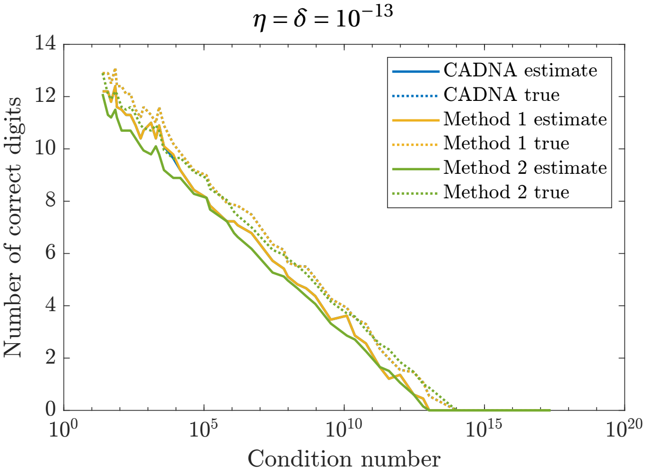
- All methods can reliably estimate the accuracy of the inner product.
- A suitable value for δ should be chosen.
- Method 2 underestimates the accuracy more frequently than Method 1
  $\Rightarrow$ slightly more pessimistic, although overall still quite reliable

Perturbed input vectors

input vectors affected by a stochastic perturbation of size η ($\delta = \eta$)



Perturbed input vectors



Method 1 equivalent to CADNA

Perturbed input vectors

- When $\eta \gg u$, the initial perturbation dominates the rounding errors.

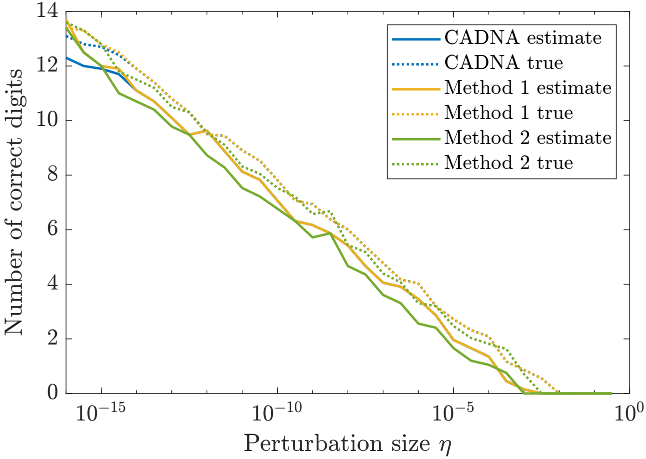
  So deterministic arithmetic in Method 1 does not change the result nor its estimated accuracy compared with CADNA.

  From a certain perturbation η ($\eta \geq 10^{-13}$) Method 1 is as reliable an estimator as CADNA.

- Method 2 is also a reliable estimator, although slightly pessimistic.

Perturbed input vectors

Accuracy of the inner product w.r.t. η, for one pair of input vectors with $\kappa \approx 1.5 \times 10^3$.



As soon as $\eta \gg u$ Method 1 becomes equivalent to CADNA.

Conclusion

- new numerical validation methods to estimate the accuracy of inner products with stochastic arithmetic.
- both methods allow for the use of efficient deterministic inner products
  performance gain for Method 1 already evaluated in [NSV'20]
- probabilistic analysis proves both methods to be reliable estimators
- reliability also confirmed via experiments: both methods compute estimations comparable to the stochastic validation method implemented in CADNA

Which of Method 1 or Method 2 should be preferred?

- Method 2 tends to be slightly more pessimistic than Method 1.
- In terms of cost,
  - Method 1 computes d inner products (in practice $d = 3$), whereas Method 2 only computes 2 of them.
  - when the input vectors are already affected by stochastic perturbations, Method 2 also requires to measure the noise δ.

Recommendation

- Method 2 when the input vectors are exact and performance is paramount
- Method 1 when the input vectors are already perturbed and/or a very tight estimate is desired.

References

This work:

F. Jézéquel, Théo Mary, Probabilistic estimation of the accuracy of inner products and application to stochastic validation, 2024.
https://hal.science/hal-04554459v1

Performance gain thanks to Method 1:

F. Jézéquel, S. Graillat, D. Mukunoki, T. Imamura, R. Iakymchuk, Can we avoid rounding-error estimation in HPC codes and still get trustworthy results?, NSV'20, 13th International Workshop on Numerical Software Verification, 2020.
https://hal.science/hal-02925976

Stochastic Arithmetic and CADNA:

J. Vignes, Discrete Stochastic Arithmetic for Validating Results of Numerical Software, Num. Algo., 37, 1–4, p. 377–390, 2004.

P. Eberhart, J. Brajard, P. Fortin, and F. Jézéquel, High Performance Numerical Validation using Stochastic Arithmetic, Reliable Computing, 21, p. 35–52, 2015.
https://hal.science/hal-01254446

CADNA: http://cadna.lip6.fr

Thanks for your attention!