Round-off error propagation and non-determinism in parallel applications

Vincent Baudoui (Argonne/Total SA)
vincent.baudoui@gmail.com
Franck Cappello (Argonne/INRIA/UIUC-NCSA)
Georges Oppenheim (Paris-Sud University)
Philippe Ricoux (Total SA)

Improve numerical simulations **precision**
$\rightarrow$ future **exascale** supercomputers

Round-off error management at the software level

Objectives:

- be able to **validate** simulation results against errors
- build a model for **error propagation**

Exascale particularities:

- **High dimension** problems
- Different **algorithms**
- Possibly **non-deterministic** behavior

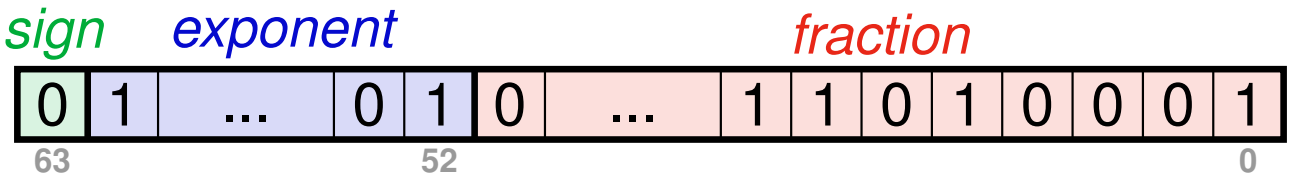
Outline

1. Round-off errors
2. Error propagation in LU decomposition
3. Impact of non-determinism

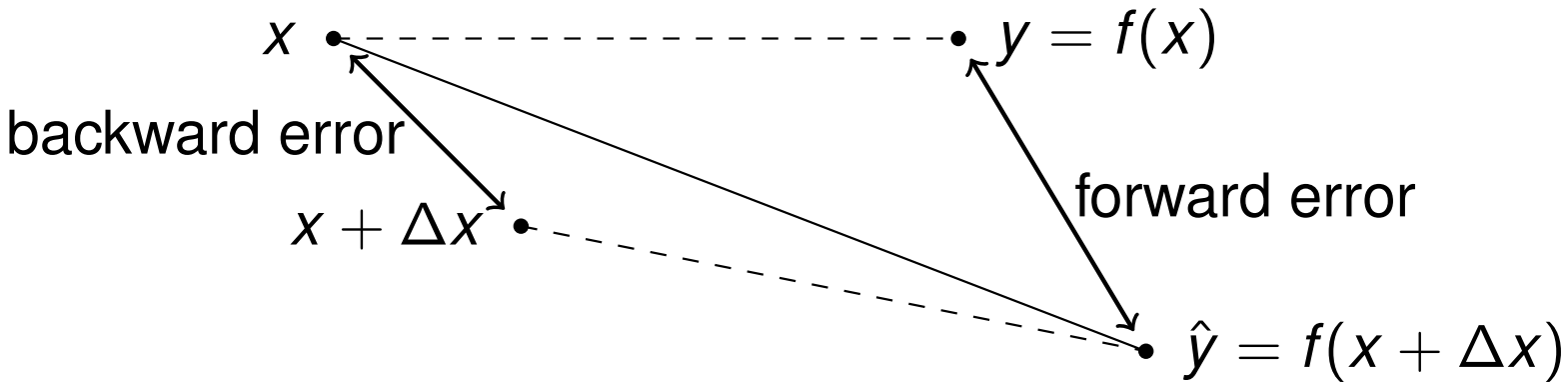
Outline

1. Round-off errors
2. Error propagation in LU decomposition
3. Impact of non-determinism

Round-off errors



Limited number of bits to encode reals → **round-off errors**
(machine precision $u \simeq 10^{-16}$ for 64-bits floating point format)



Forward error: error on the result $E = |\hat{y} - y|$
Backward error: perturbation in the data $\epsilon = \Delta x$

$$\text{forward error} \lesssim \text{condition number} \times \text{backward error}$$

Higham (2002), Accuracy and stability of numerical algorithms

Outline

1. Round-off errors
2. Error propagation in LU decomposition
3. Impact of non-determinism

LU decomposition

Study round-off error **propagation**

LU decomposition to solve linear systems in numerical simulations (e.g. wave equation in depth imaging)

Decompose $A \in \mathbb{R}^{4\times 4}$ into triangular matrices L and U

$$\begin{array}{ccccc} A & = & L & \times & U \end{array}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ l_{21} & 1 & 0 & 0 \\ l_{31} & l_{32} & 1 & 0 \\ l_{41} & l_{42} & l_{43} & 1 \end{bmatrix} \times \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ 0 & u_{22} & u_{23} & u_{24} \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & 0 & u_{44} \end{bmatrix}}_{\begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ l_{21} & u_{22} & u_{23} & u_{24} \\ l_{31} & l_{32} & u_{33} & u_{34} \\ l_{41} & l_{42} & l_{43} & u_{44} \end{bmatrix}}$$

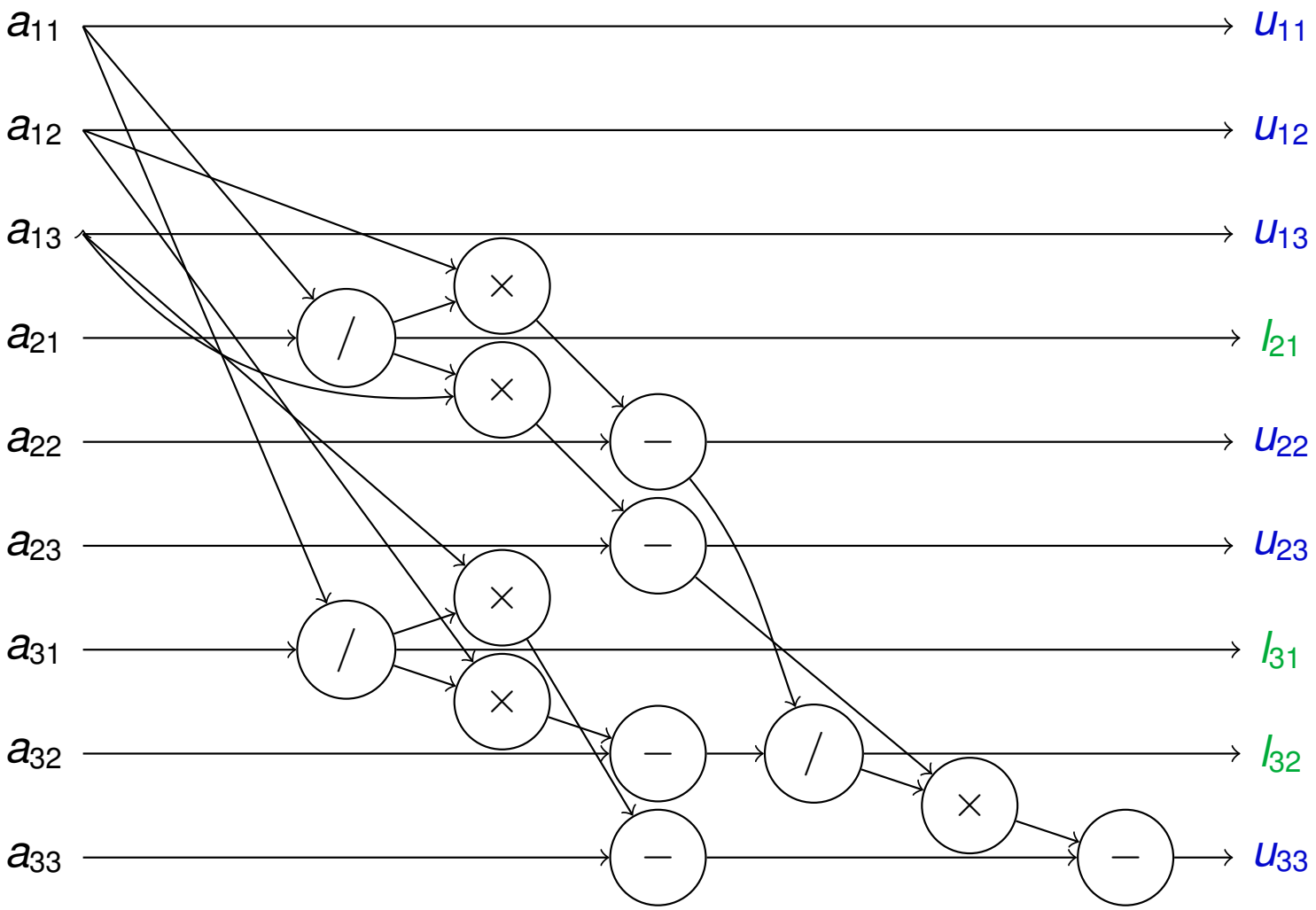
LU decomposition

LU decomposition algorithm without pivoting (fast and simple)

```
for i = 1..4 do
    for j = i..4 do
        for k = 1..(i − 1) do
            t = a_ik × a_kj
            a_ij = a_ij − t
    for j = (i + 1)..4 do
        for k = 1..(i − 1) do
            t = a_jk × a_ki
            a_ji = a_ji − t
        a_ji = a_ji / a_ii
```

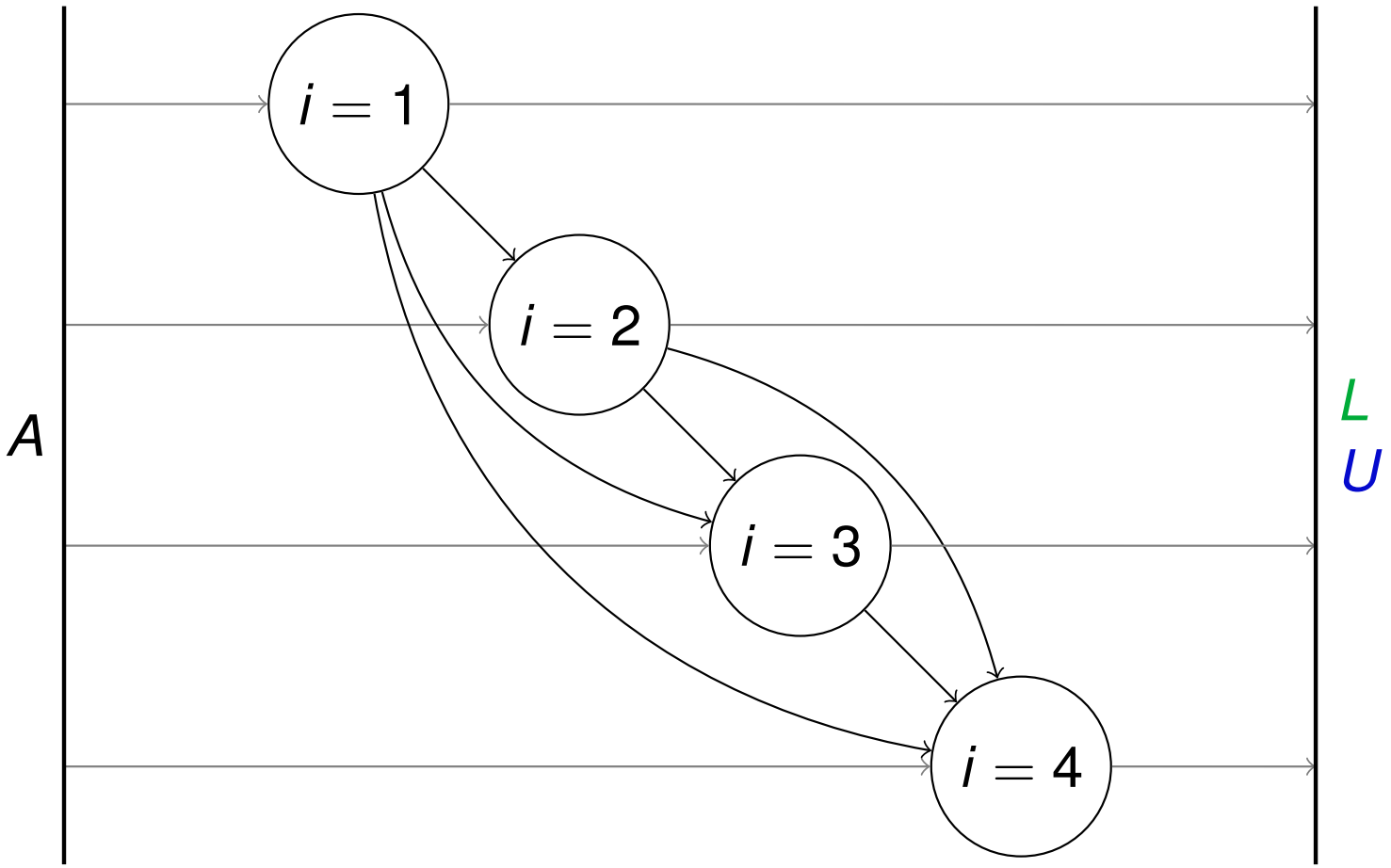
LU decomposition

Instruction level graph for $A \in \mathbb{R}^{3\times 3}$



LU decomposition

Hierarchical graph



Error propagation

- **Analytical** worst-case bounds
- **Statistical** analysis (CADNA, PRECISE)
- Error propagation through partial **derivatives**

First order approximation of output error E_y:

$$E_y \simeq \sum_i \delta_i \frac{\partial y}{\partial \delta_i}$$

with respect to instruction errors δ_i: $\hat{x}_i = x_i + \delta_i, \ |\delta_i| \leq u$

Miller & Wrathall (1980), Software for roundoff analysis of matrix algorithms

Error propagation

Partial derivatives computed using **algorithmic differentiation**:

$$\frac{\partial y}{\partial \delta_i} = \prod_{(a,b) \in P(\delta_i, y)} \frac{\partial b}{\partial a}$$

Instruction error δ_i **quantification** for elementary operations $+$, $-$, $\times$, $/$ (estimation only for the division operator):

$$s = a + b \quad \Rightarrow \quad \delta_i = (s - a) - b$$

$\rightarrow$ output error $E_y \simeq \sum_i \delta_i \frac{\partial y}{\partial \delta_i}$ quantified!

Langlois (2001), Automatic linear correction of rounding errors

Error propagation

Limitations

- First order approximation:
  - exact for "**linear**" algorithms (multiplication/division by an error-free number $\rightarrow$ no cross-interaction between instruction input errors)
- Computational **cost**:
  - derivatives computation
  - instruction error estimation

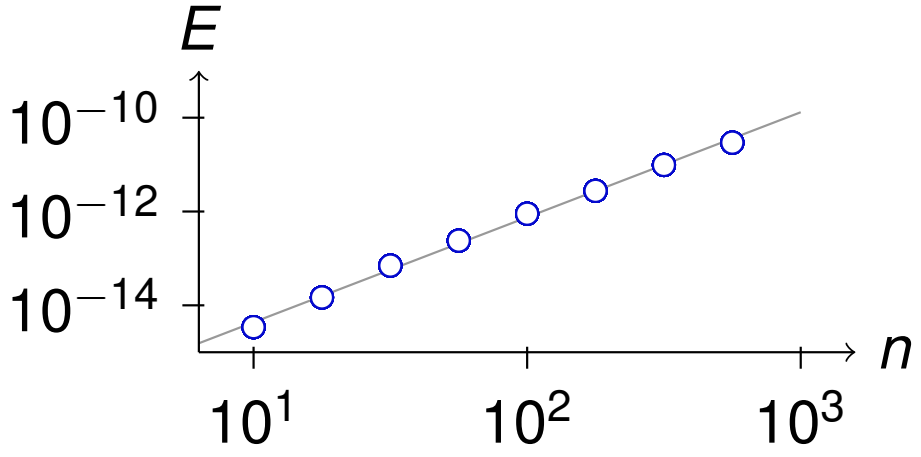
Experiments

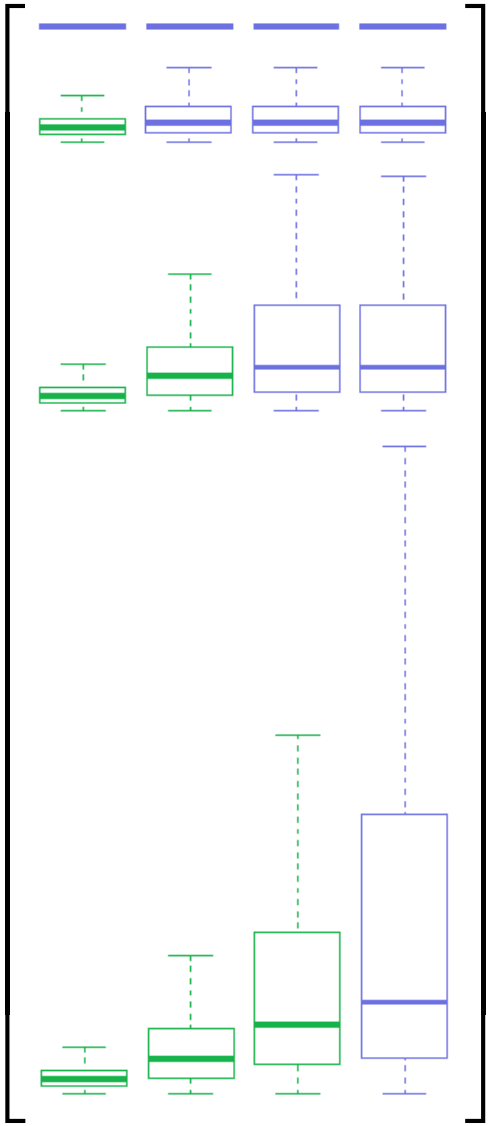
Experiments on **Gaussian random matrices**
$A \sim \mathcal{N}(0, I_4)$ (100,000 samples)

Median output error:

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 4.28 & 5.42 & 5.41 & 5.42 \\ 4.28 & 9.52 & 11.9 & 11.9 \\ 4.26 & 9.59 & 18.7 & 24.9 \end{bmatrix} \times 10^{-17}$$

Last element error evolution with matrix size:



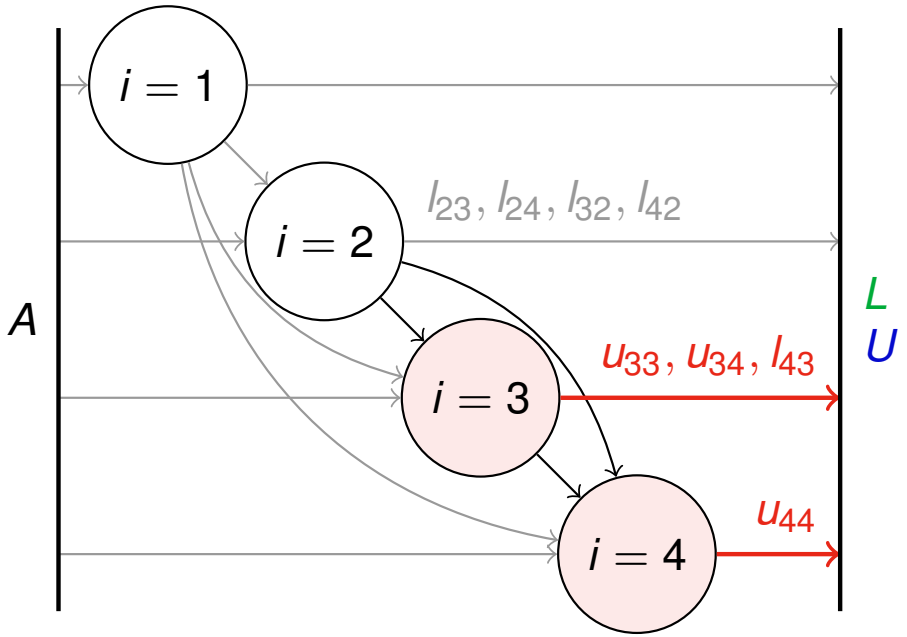


Experiments

Semi-random matrices: $A \sim \mathcal{N}(0, I_4)$ except for $a_{11} = 10^{-10}$

Median output error:

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 3.86 & 5.87 & 5.89 & 5.86 \\ 3.84 & 9.14 & \mathbf{1.35 \times 10^{10}} & \mathbf{1.35 \times 10^{10}} \\ 3.83 & 9.13 & \mathbf{2.62 \times 10^{10}} & \mathbf{4.89 \times 10^{10}} \end{bmatrix} \times 10^{-17}$$



Median derivatives ~ 0.5 but:

$$\frac{\partial u_{33}}{\partial l_{23}}, \frac{\partial u_{33}}{\partial l_{32}}, \frac{\partial u_{34}}{\partial l_{24}}, \frac{\partial u_{34}}{\partial l_{32}}, \frac{\partial l_{43}}{\partial l_{23}},$$
$$\frac{\partial l_{43}}{\partial l_{32}}, \frac{\partial l_{43}}{\partial l_{42}}, \frac{\partial u_{44}}{\partial l_{24}}, \frac{\partial u_{44}}{\partial l_{42}} \sim 10^6$$

Outline

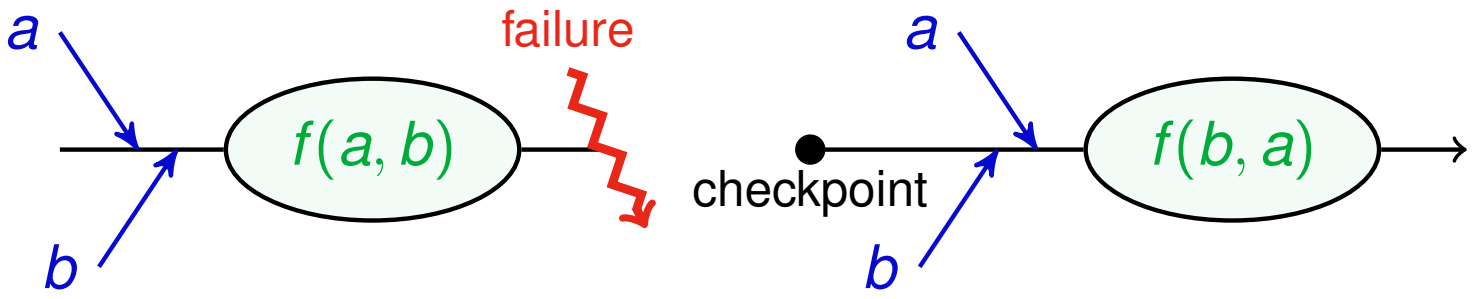
1. Round-off errors
2. Error propagation in LU decomposition
3. Impact of non-determinism

Impact of non-determinism

Non-determinism in parallel applications

- Data may arrive in **random order** (delays in message passing between processes)
- If data is processed in order of arrival, results can be **non-deterministic** because of round-off errors

If message order is not recorded:



Will we converge to the same solution in the same time?

1D stencil

Example: 1D Jacobi stencil

b_L	x_1^0	x_2^0	x_3^0	x_4^0	x_5^0	x_6^0	x_7^0	x_8^0	x_9^0	x_{10}^0	b_R

10 elements vector X with initial state X^0
and boundary conditions b

At each iteration, x_i is updated as the mean of its neighbors:

$$x_i^{n+1} = f(x_{i-1}^n, x_{i+1}^n) = \frac{x_{i-1}^n + x_{i+1}^n}{2}$$

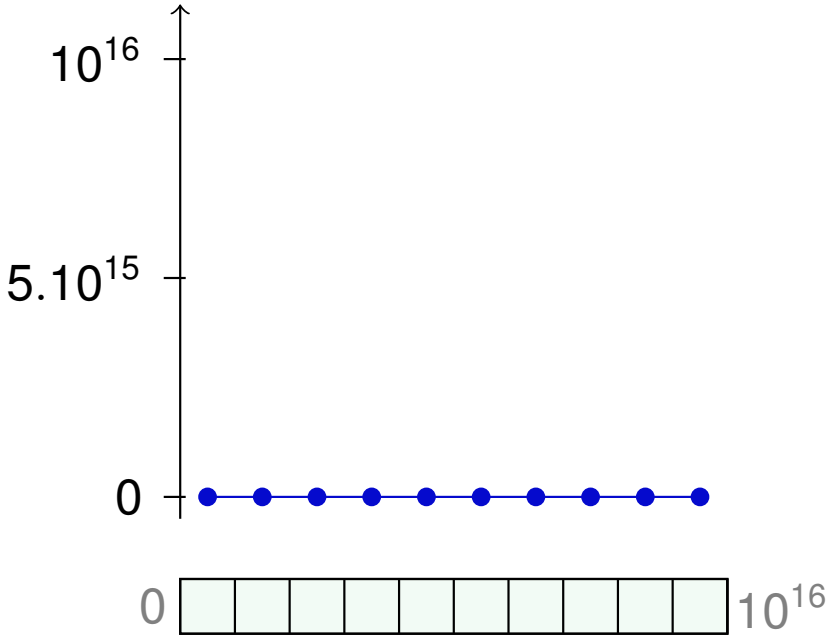
$\rightarrow$ Markov process: $X^{n+1} = F(X^n)$

Converges if F is contracting: $d(F(A), F(B)) \leq kd(A, B),\ k < 1$

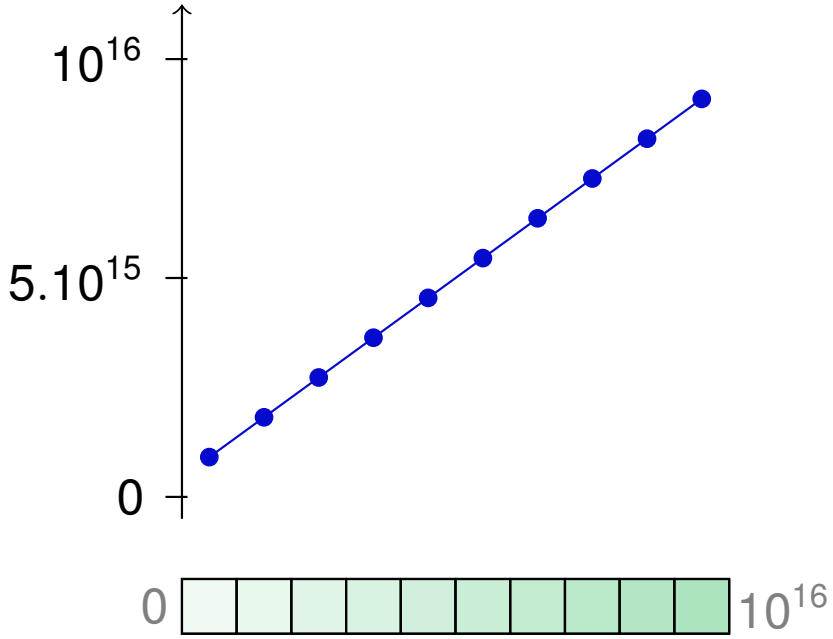
1D stencil

Deterministic behavior

initial state X^0



final state X^{832}



Non-determinism

Random data order

$$x_i^{n+1} = \begin{cases} f(x_{i-1}^n, x_{i+1}^n) & \text{with probability } p \\ f(x_{i+1}^n, x_{i-1}^n) & \text{with probability } (1-p) \end{cases}$$

Harmless if f is symmetrical

But f can be non-symmetrical in practice because of round-off errors:

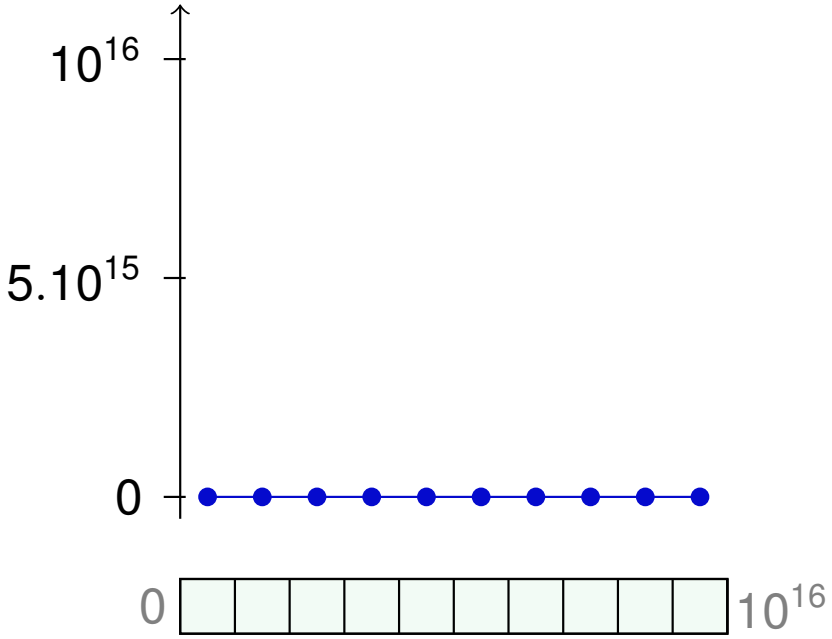
$$f(a,b) = \frac{a+b}{-1+a-a+3} \simeq \begin{cases} (a+b)/2 & \text{when } a \lesssim 10^{16} \\ (a+b)/3 & \text{otherwise} \end{cases}$$

$\rightarrow$ **non-deterministic** results

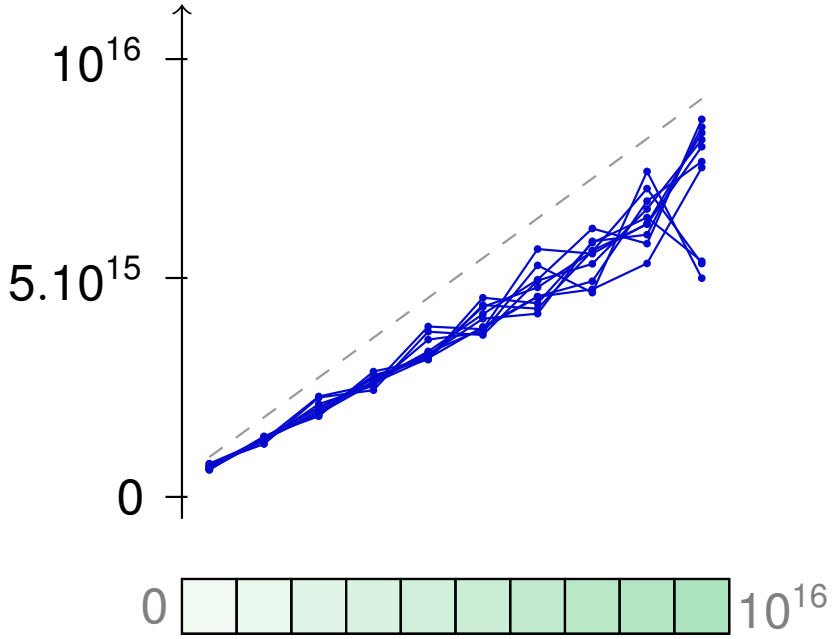
Non-determinism

Non-deterministic behavior

initial state X^0



X^{10000} for 10 runs



Non-determinism

Converges (to a stationary distribution) if the F functions are contracting in average: $\mathrm{E}[\log(k)] < 0$

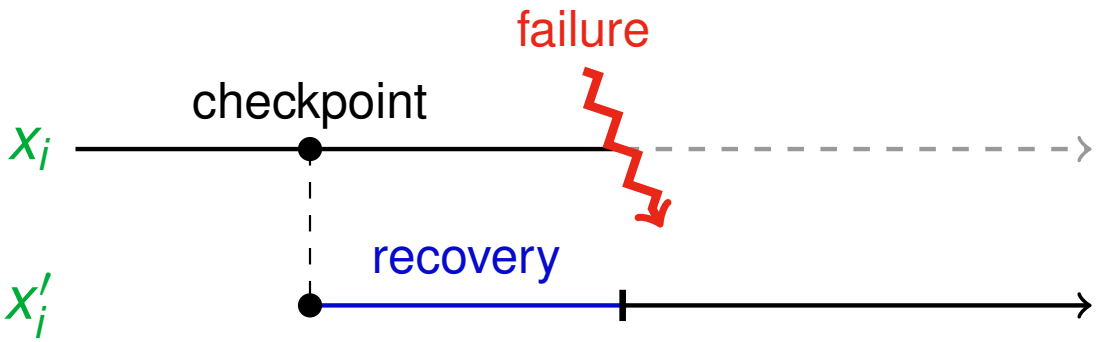
$\rightarrow$ some functions can be non-contracting if their probability is relatively low

Diaconis, P. & Freedman, D. (1999), Iterated random functions

Failure recovery

If the ith cell process fails, x_i^n is lost and rolls back to the last checkpoint $x_i^{n-\tau}$ $\rightarrow$ iterations $n-\tau$ to n must be recomputed

Result may differ because of data order randomness
$\rightarrow$ system restarts from a possibly different state X'^n



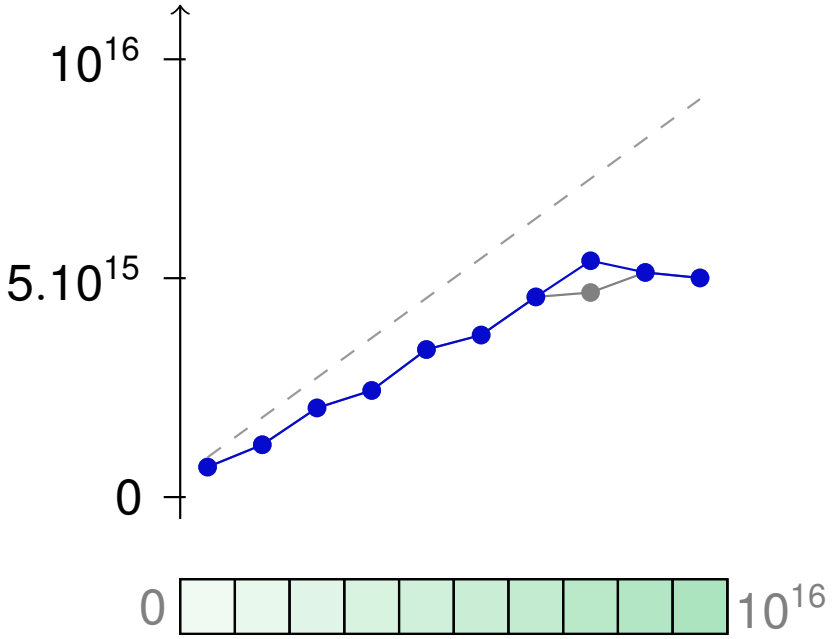
If X'^n remains in the same attraction domain as X^n
it will converge to the same distribution at the same speed
but number of iterations may vary

Failure recovery

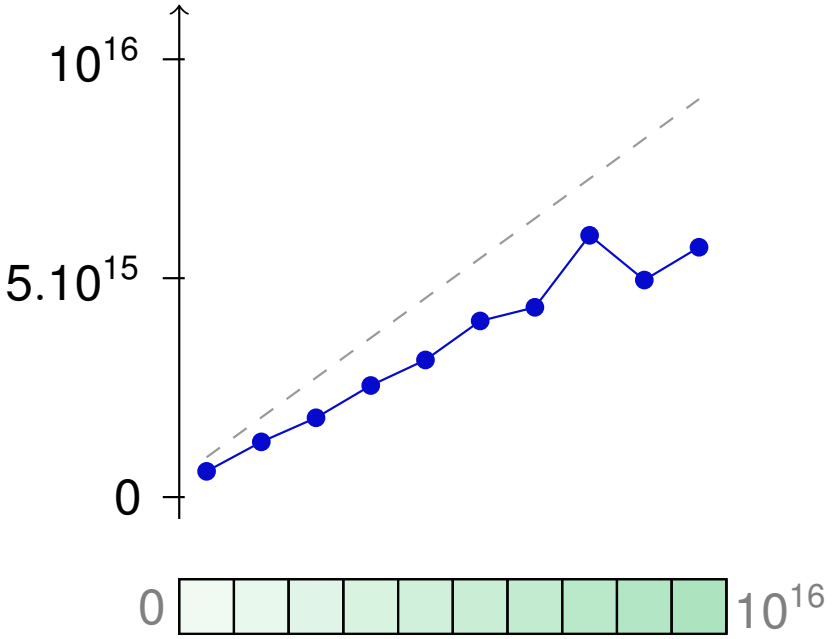
Recovery

Failure at x_8^{500}, checkpoint restart from X^{400}

X^{500} before/after recovery



X^{1256}



$\rightarrow$ goes back to the same solution after a while here

Conclusion

- Run massively parallel numerical simulations on future **exascale** supercomputers
- **Validate** simulation results against round-off errors

- Error **propagation** through derivatives
- Study other **LU decomposition algorithms**
- Define **heuristics** for graph exploration

- Impact of **non-determinism** on recovery
- No need to log **message order** under some hypothesis
- Study a **real test-case**

Round-off error propagation and non-determinism in parallel applications

vincent.baudoui@gmail.com

Thank you for your attention!