Mixed-Precision for solving large biological models

SMAI Biennal - 2025

Arsène Marzorati

June 3rd

Environment

Directors: Samuel Bernard (MUSICS) & Jonathan Rouzaud-Cornabas (Biotic).

Collaborator: Mouhammad Al-Sayed Ali

AEx Inria:ExODE

RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

Inria

Grid'5000

Context

Scaling ODE solvers

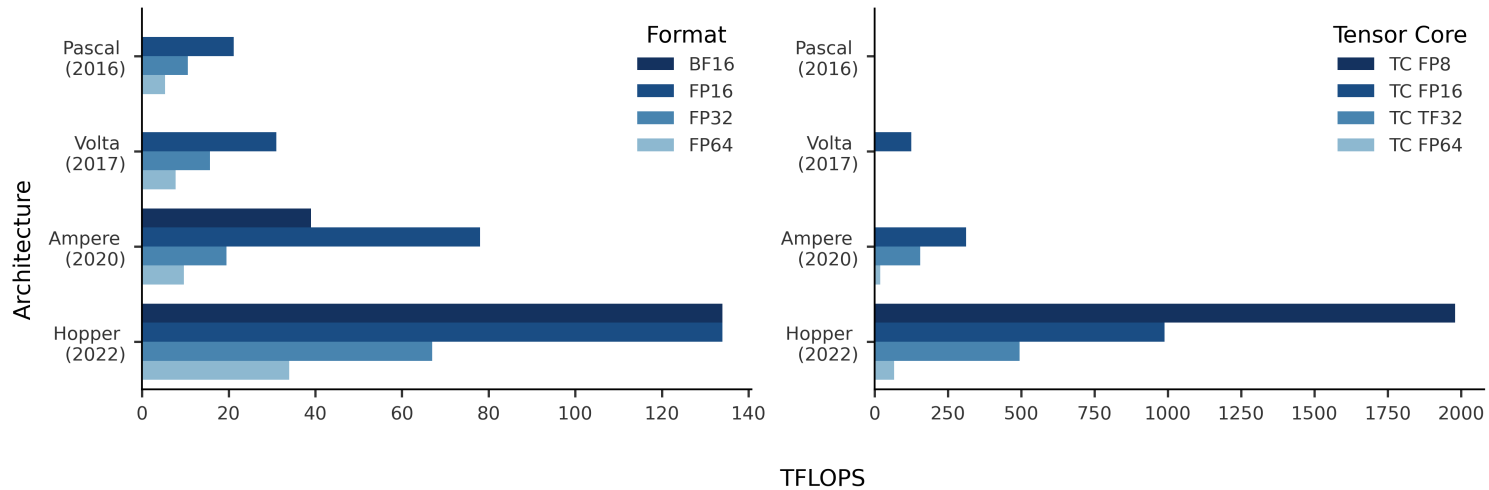
- **Agent-based models** for modelling biological processes.
- Large scale: "real-life" size, more rules by agent.
- Solving high dimensional and complex systems of ODEs.

General model

$$\dot{X}_i = \mathrm{F_i}(X) = \mathrm{H_i}(X_i) + \frac{1}{N}\sum_{j=1}^{N} W_{ij} \odot \mathrm{G}_{ij}(X_i, X_j) \; i \in \{1, ..N\}, \; X_i \in \mathbb{R}^d.$$

Mixed precision: a trade-off ?

- Use several numerical formats (*e.g.* FP64, FP32, FP16[1]) inside one computational tool.
- Computational acceleration, less memory needed, better error control.
- A mandatory step ?[2]



[1] Zuras, Dan, et al. "IEEE standard for floating-point arithmetic." IEEE Std 754.2008 (2008): 1-70.

[2] "NVIDIA A100 tensor core GPU architecture." Whitepaper (2020), V1.0.
"NVIDIA H100 tensor core GPU architecture." Whitepaper (2023), V1.04.

Explicit solver for ODE

Discretization and explicit scheme

- *ODE system:* $\dot{\mathrm{X}} = \mathrm{F}(t, \mathrm{X}),\ t \in [0, T]$.
- *Step:* $h_n = t_{n+1} - t_n > 0$, $\mathrm{X}_n = \mathrm{X}(t_n)$.
- *Stage(s):* $k_s = \mathrm{F}(t_n + c_s h, \mathrm{X}_n + h \sum_{j=1}^{s-1} a_{sj} k_j),\ \forall s \in \{1, ..., p\}$.

$$\mathrm{X}_{n+1} = \mathrm{X}_n + h \sum_{s=1}^{p} b_s k_s.$$

Adaptive scheme (*ODE23*)

- Error indicator and step validation modifying the step size (h_n).
- Combination of 2 **embedded** Runge-Kutta methods (First Same As Last property).

All the coefficients (a_{sj}, c_s, b_s) characterizing a numerical scheme can be grouped in a Butcher Table.

Mixed-precision at different levels

- For function evaluation, 3 possibilities can be chosen :

$$\dot{X}_i = \mathrm{H_i}(X_i) + \frac{1}{N}\sum_{j=1}^{N} W_{ij} \odot \mathrm{G}_{ij}(X_i, X_j).$$

- For each stage (p-stages, 3 in our case), different *precision-specification* can be chosen.

	Precision		
Stage	$\mathrm{H_i}(\cdot)$	$\frac{1}{N}\sum_{j=1}^{N}$	$W_{ij} \odot \mathrm{G}_{ij}(\cdot)$
k_2	S	S	S
k_3	D	D	S
k_4	D	D	D

Benchmarks

Linear coupled oscillators

$\forall i \in \{1, ..., N\}$,

$$\begin{cases} \frac{dx_i}{dt} = y_i + \frac{1}{N} \sum_{j=1}^{N} (x_j - x_i) \\ \frac{dy_i}{dt} = -x_i \end{cases}$$

why: Analytic solution

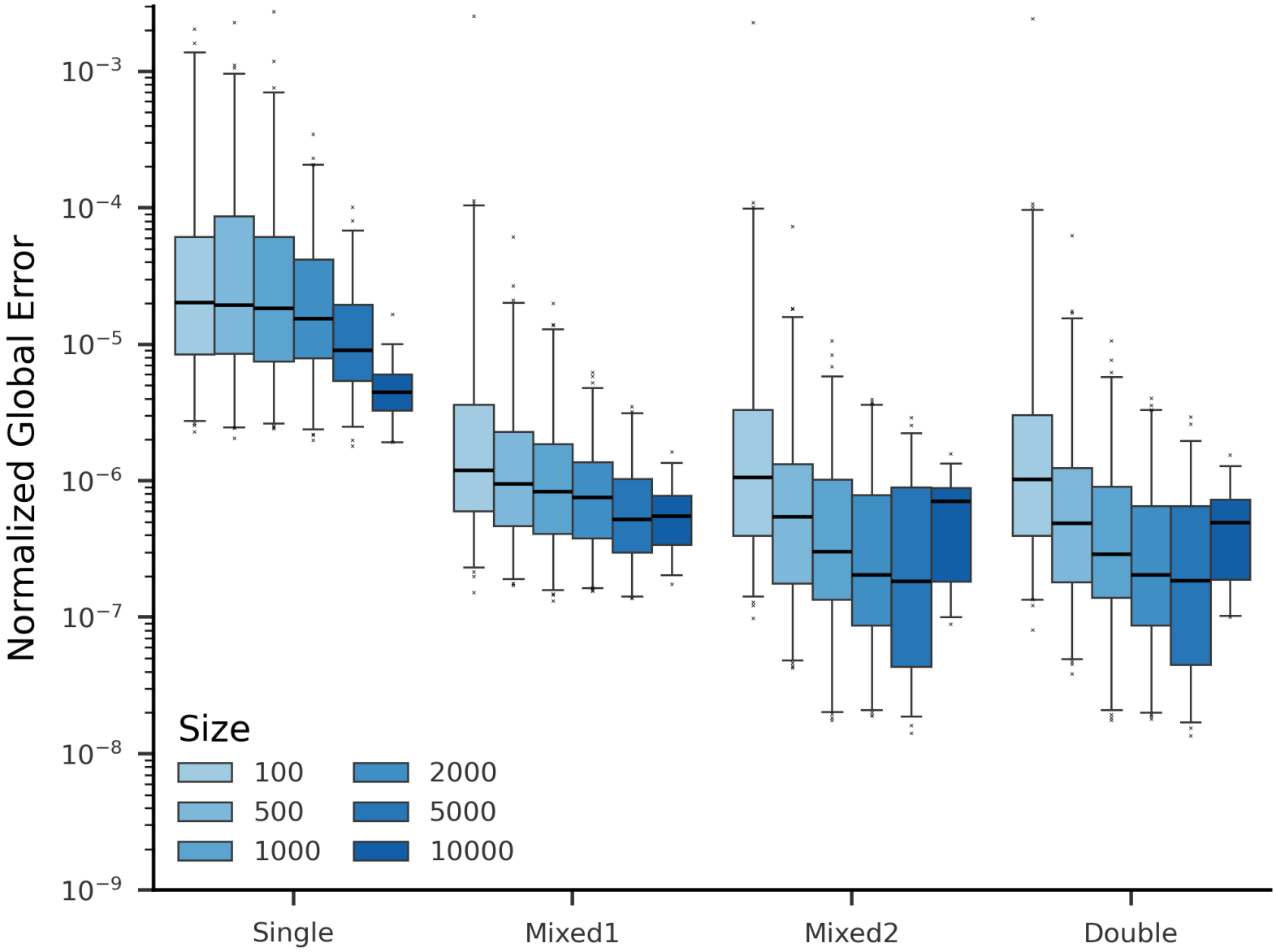
Kuramoto

$\forall i \in \{1, ..., N\}$,

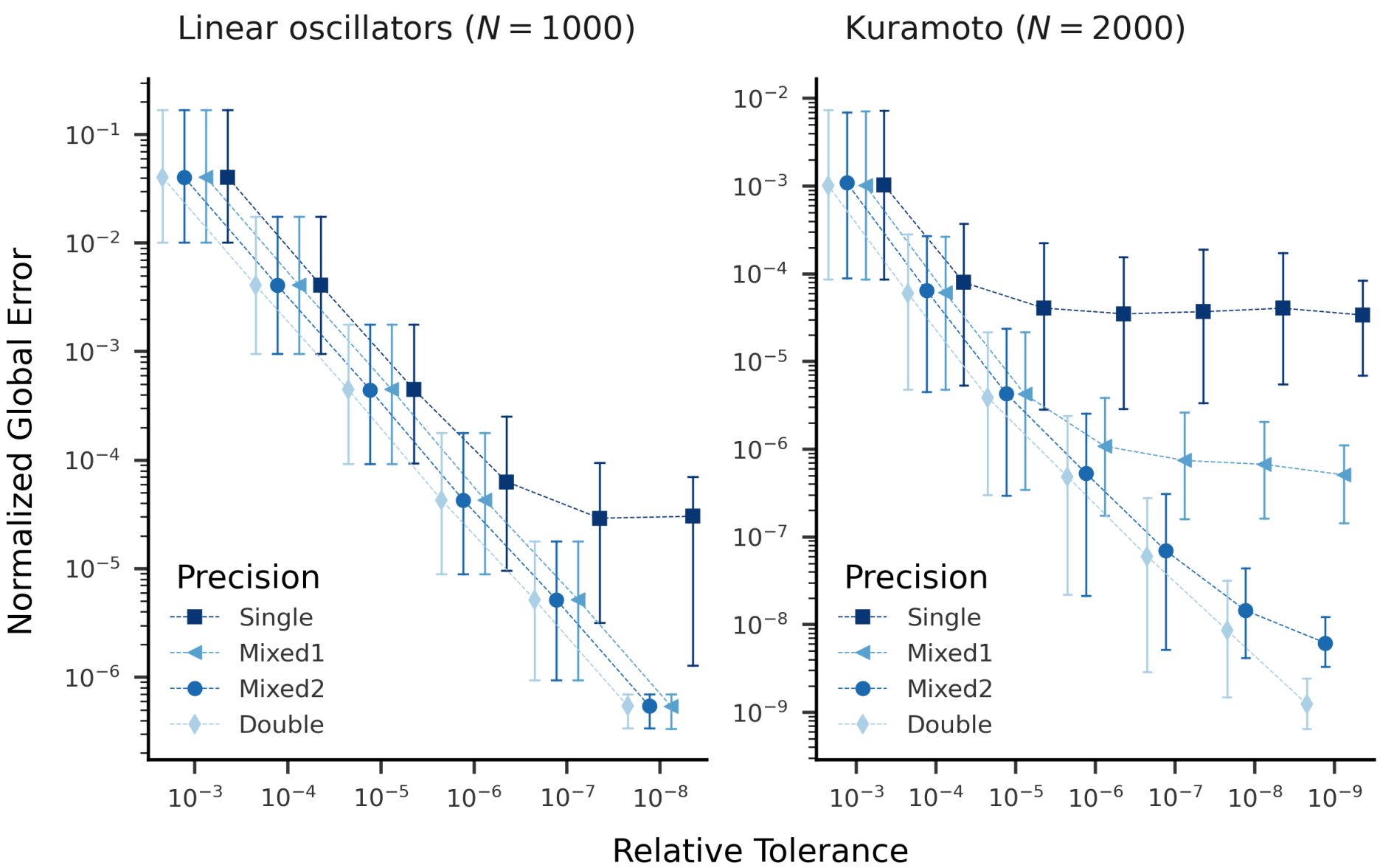
$$\frac{dx_i}{dt} = \omega_i + \frac{1}{N} \sum_{j=1}^{N} K \sin(x_j - x_i)$$

why: Non-linear interaction term (sine)

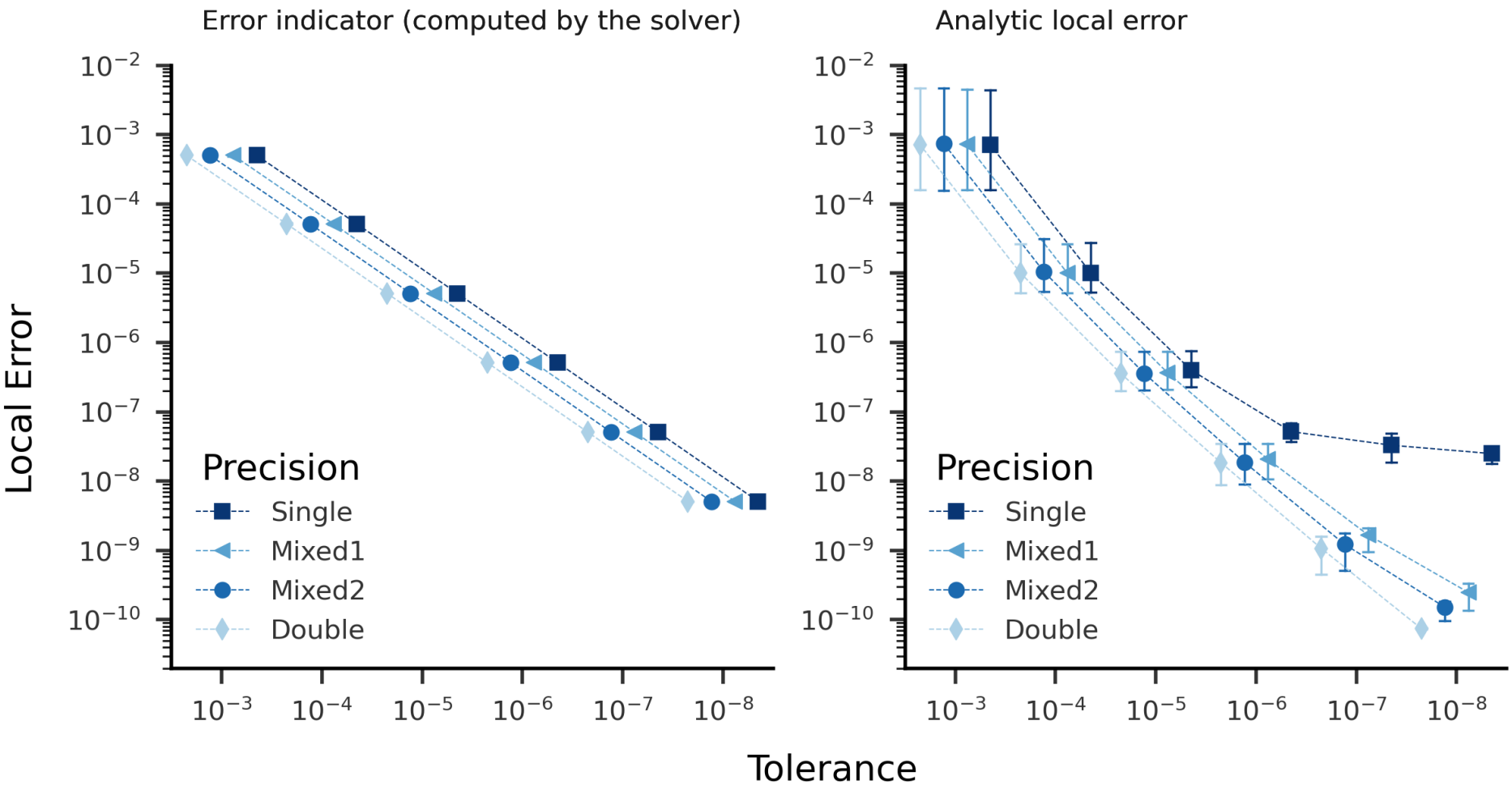
Normalized Global Error VS System Size (on Kuramoto)



Normalized Global Error VS Tolerance



Local Error with linear coupled oscillators



Conclusion on Error Analysis

1. Benefit with the size.
2. Mixed-precision is more stable against stringent tolerances.
3. Local error better estimated in mixed-precision.

What about performance ?

Adaptive scheme and first criteria

Let be a given solver precision: P (double, single, mixed).

- Number of steps: $N_{step,P}$.
- Problem complexity (number of FP operations): Θ.
- Fraction of single precision operations: ϱ.
- FP operation times: t_{single}, t_{double} and $r = \frac{t_{single}}{t_{double}}$.

The solving time for solver P

$$T_P = N_{step,P}\Theta\big(\varrho t_{single} + (1-\varrho)t_{double}\big).$$

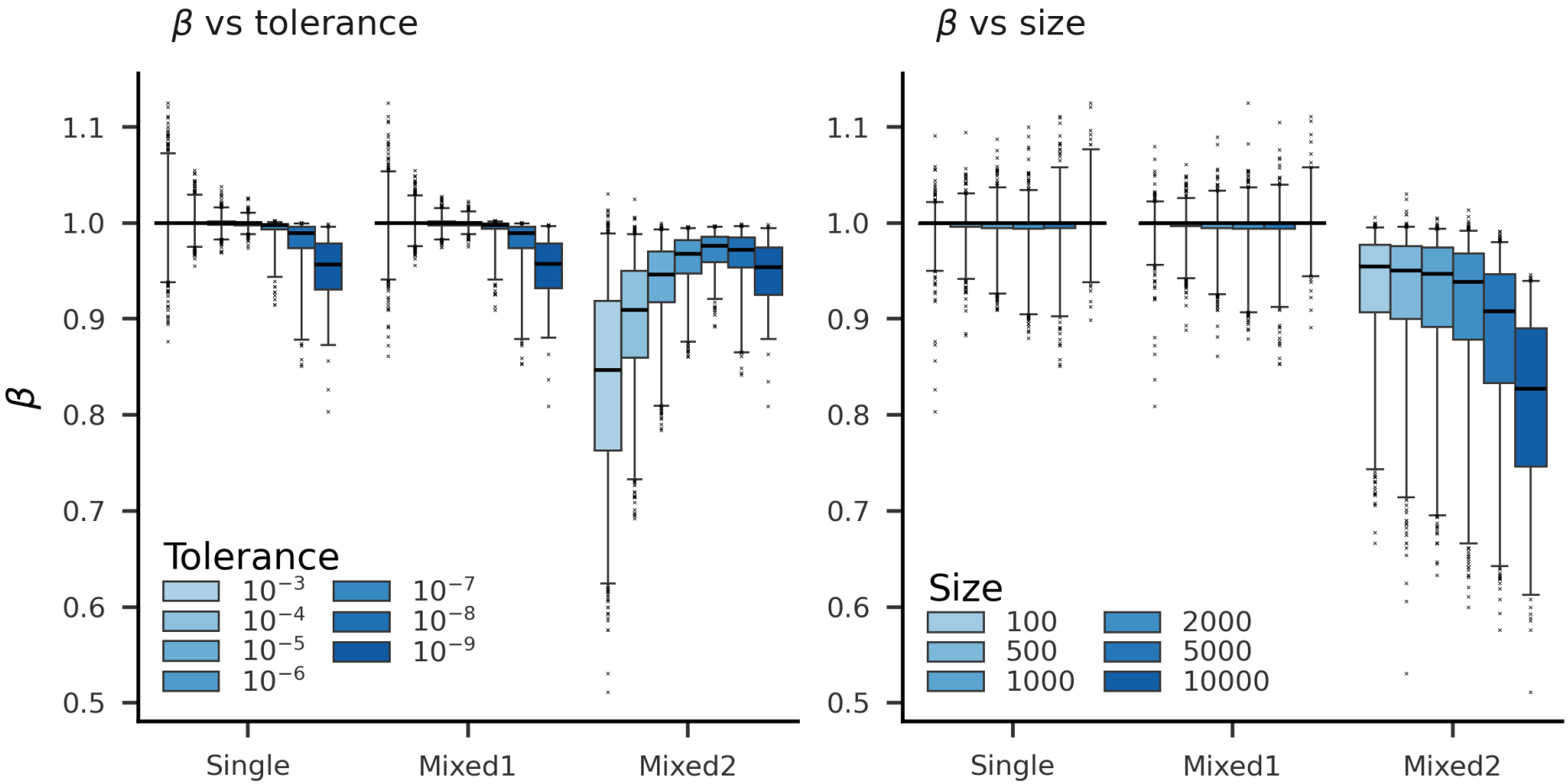
Performance parameters

Taking "*double*" as reference, we introduce:

$$\beta := \frac{N_{step,double}}{N_{step,P}}, \quad \gamma := \varrho r + (1-\varrho),$$

$$\Gamma := \frac{T_P}{T_{double}} = \frac{\gamma}{\beta}.$$

β parameter: number of steps ratio



Real performance measure (Γ)

Recall on Γ

$$\Gamma := \frac{T_P}{T_{double}} = \frac{\gamma}{\beta}.$$

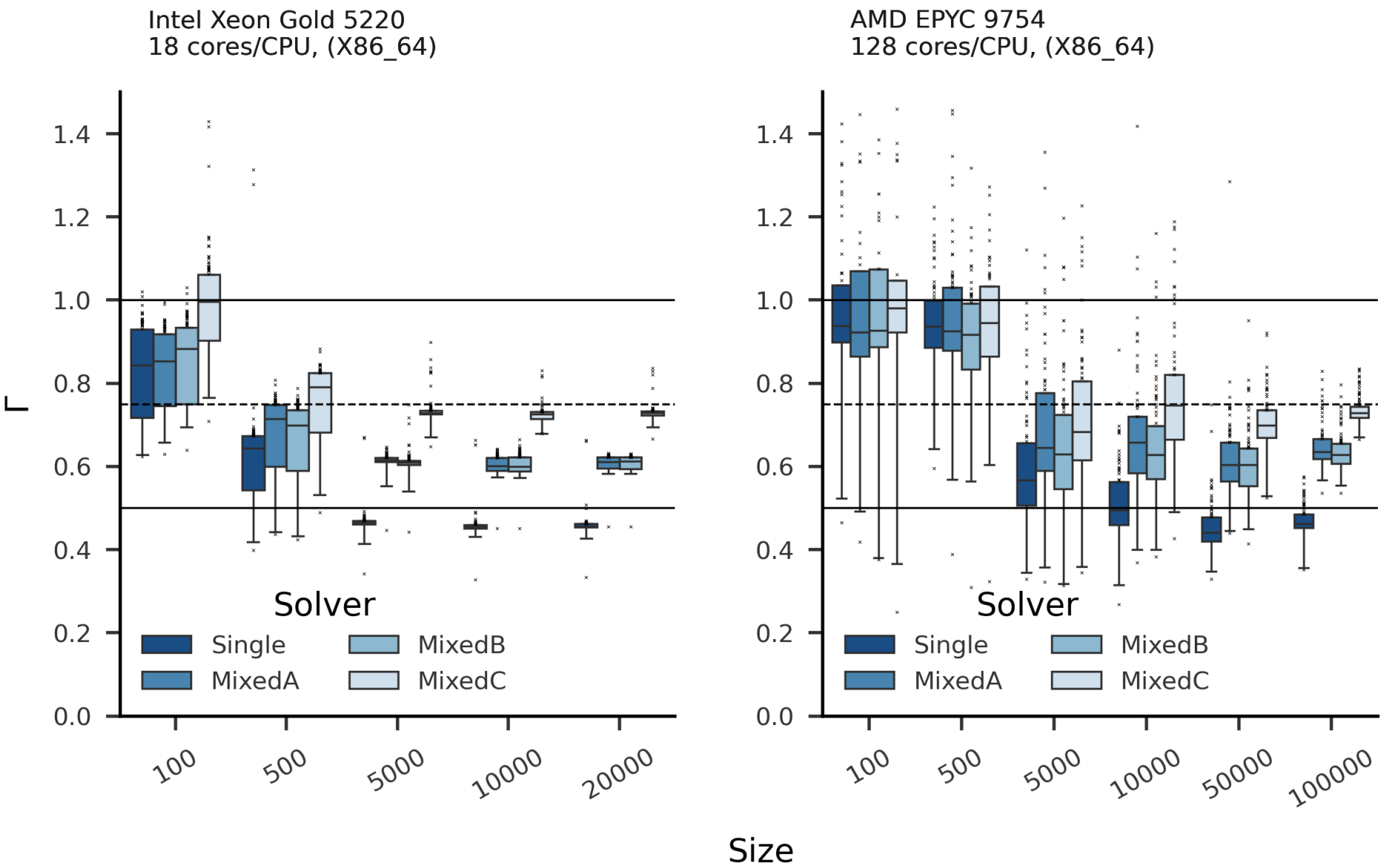
Performance gain if $\Gamma < 1 \Leftrightarrow \gamma < \beta$.
$\gamma = \varrho r + (1 - \varrho)$.

Real performance

An approach with fixed step scheme.
Why? Identical number of function evaluations. [a]

[a] Ignoring the cast/convert.

Γ parameter on Kuramoto



Conclusion

Take home message

- Error in mixed-precision close to double one.
- Error benefits from the size.
- Performance gain can be optimized (r, ϱ, model).

Future work

- More complex problems (heterogeneous W_{ij}, interactions...), real use cases.
- Selective precision heuristic with more than 2 FP formats.
- Explore storage performance.

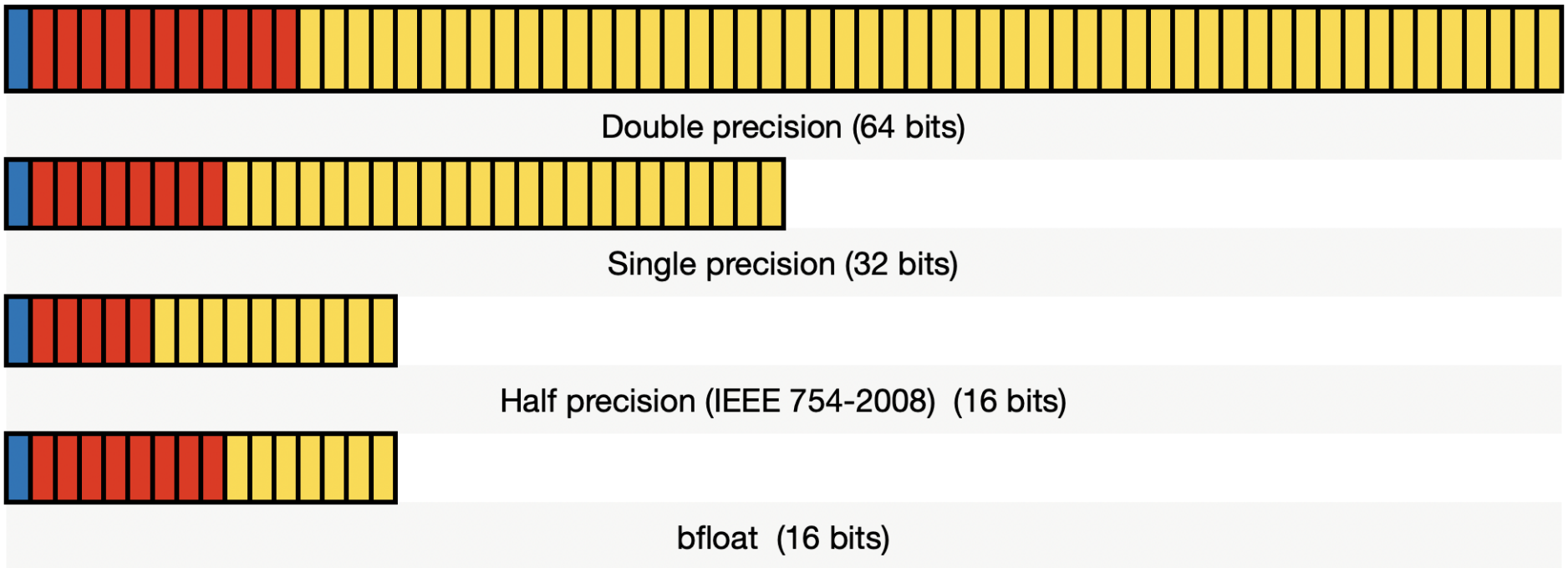
Selected bibliography

1. Bogacki P. and Shampine L.F., "A 3 (2) pair of Runge-Kutta formulas." Applied Mathematics Letters 2.4 (1989): 321-325.
2. Burnett B., Gottlieb S., Grant Z.J., and Heryudono A., "Performance evaluation of mixed-precision Runge-Kutta methods. IEEE High Performance Extreme Computing Conference (HPEC) (2021).
3. Croci M. and de Souza G.R., "Mixed-precision explicit stabilized Runge-Kutta methods for single- and multi-scale differential equations." J. Comput. Phys.(2022)
4. Higham N.J. and Mary T. "Mixed precision algorithms in numerical linear algebra." Acta Numerica 31 (2022): 347-414.
5. Haidar A., Tomov S., Dongarra J. and Higham N.J., "Harnessing GPU tensor cores for fast FP16 arithmetic to speed up mixed-precision iterative refinement solvers." SC18: Int. Conf. High Perform. Comput. Netw. Storage Anal. IEEE (2018).
6. Choquette J. and Wish G., "Nvidia a100 gpu: Performance & innovation for gpu computing." IEEE Hot Chips 32 Symposium (HCS). IEEE Computer Society, (2020).
7. Hayford J., Goldman-Wetzler J., Wang E. and Lu L., "Speeding up and reducing memory usage for scientific machine learning via mixed precision", arXiv preprint (2024).

Floating-point formats

Classic Floating-Point formats are encoded on n bits split into:

- Sign bit (*blue*)
- Exponent bits (*red*)
- Significand bits (*yellow*)



Floating-point properties

The number of bits and their distribution impact some properties[1]:

Format	Bits (S+E+M)	$\epsilon_m = 2^{-M}$	MAXVAL	MINVAL
Double	1 + 11 + 52	$2^{-52} \approx 2.22\,10^{-16}$	$1.80\,10^{308}$	$2.23\,10^{-308}$
Single	1 + 8 + 23	$2^{-23} \approx 1.19\,10^{-7}$	$3.40\,10^{38}$	$1.18\,10^{-38}$
Half	1 + 5 + 10	$2^{-10} \approx 9.77\,10^{-4}$	65504	$6.10\,10^{-5}$
bfloat	1 + 8 + 7	$2^{-7} \approx 7.81\,10^{-3}$	$3.40\,10^{38}$	$1.18\,10^{-38}$

- Range, ϵ machine
- Cancellation
- Handling exceptions...

[1] Goldberg, David. "What every computer scientist should know about floating-point arithmetic." ACM computing surveys (CSUR) 23.1 (1991): 5-48.

Precision-selection

Adaptive scheme

- Reference solution: `ODE45`[a] with relative tolerance at 10^{-9}.
- *ODE23* coded in mixed-precision. Double and Single are performed with the mixed-precision version.

[a] 5(4) Dormand-Prince pair

	Mixed1			Mixed2		
Stage	$\mathrm{H_i}(\cdot)$	$\sum$	$\mathrm{G}_{ij}(\cdot)$	$\mathrm{H_i}(\cdot)$	$\sum$	$\mathrm{G}_{ij}(\cdot)$
k_2	S	S	S	D	D	S
k_3	S	S	S	D	D	S
k_4	D	D	S	D	D	S

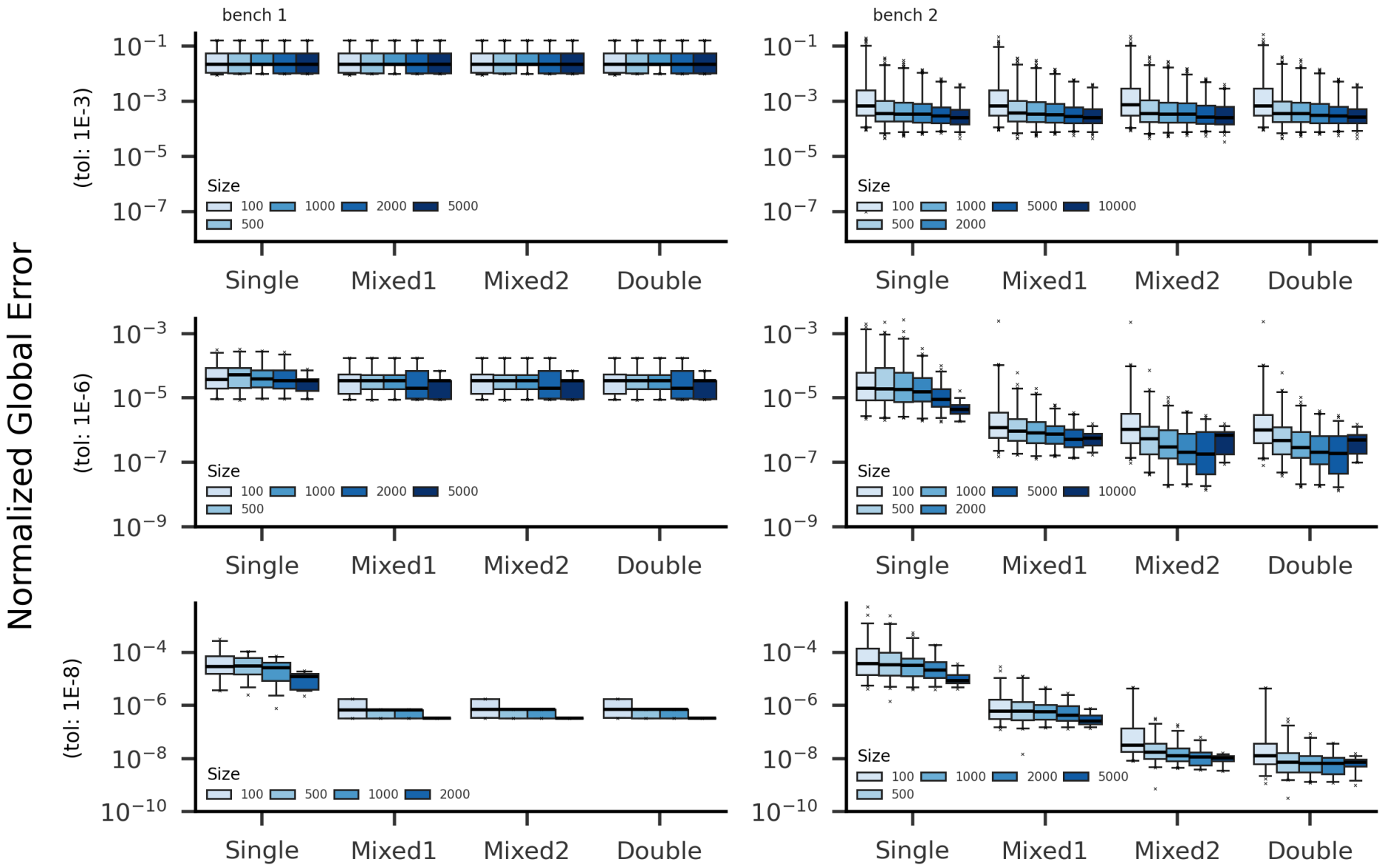
Precision-selection

Fixed-step scheme

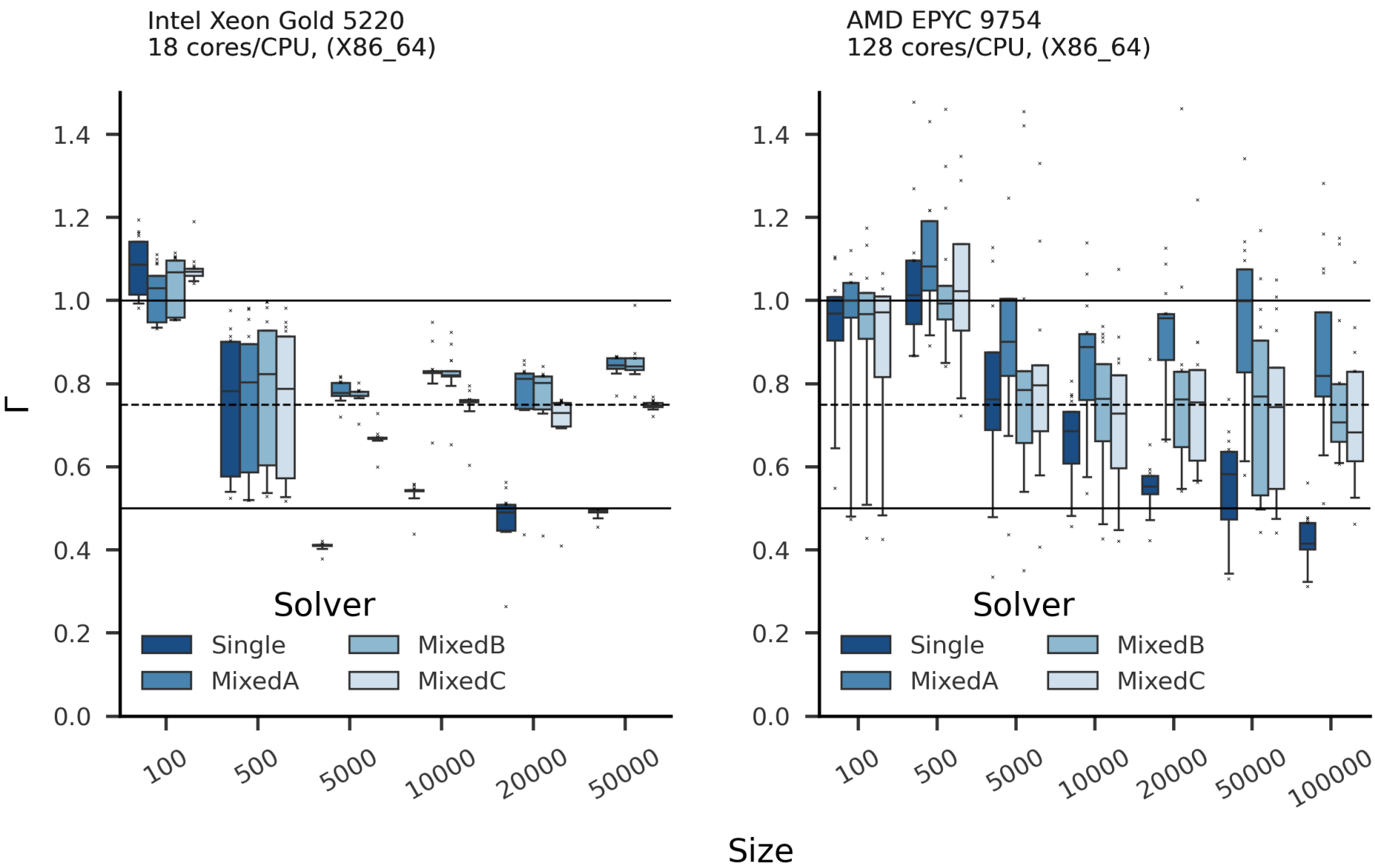
- Reference solution: RK5 (Nyström's fifth-order method) with 5000 steps.
- "Original" RK4 coded in mixed-precision. Double and Single are performed with the mixed-precision version.

	MixedA			MixedB		
Stage	$\mathrm{H_i}(\cdot)$	$\sum$	$\mathrm{G}_{ij}(\cdot)$	$\mathrm{H_i}(\cdot)$	$\sum$	$\mathrm{G}_{ij}(\cdot)$
k_1	D	D	S	D	D	D
k_2	D	D	S	S	S	S
k_3	D	D	S	S	S	S
k_4	D	D	S	D	D	D

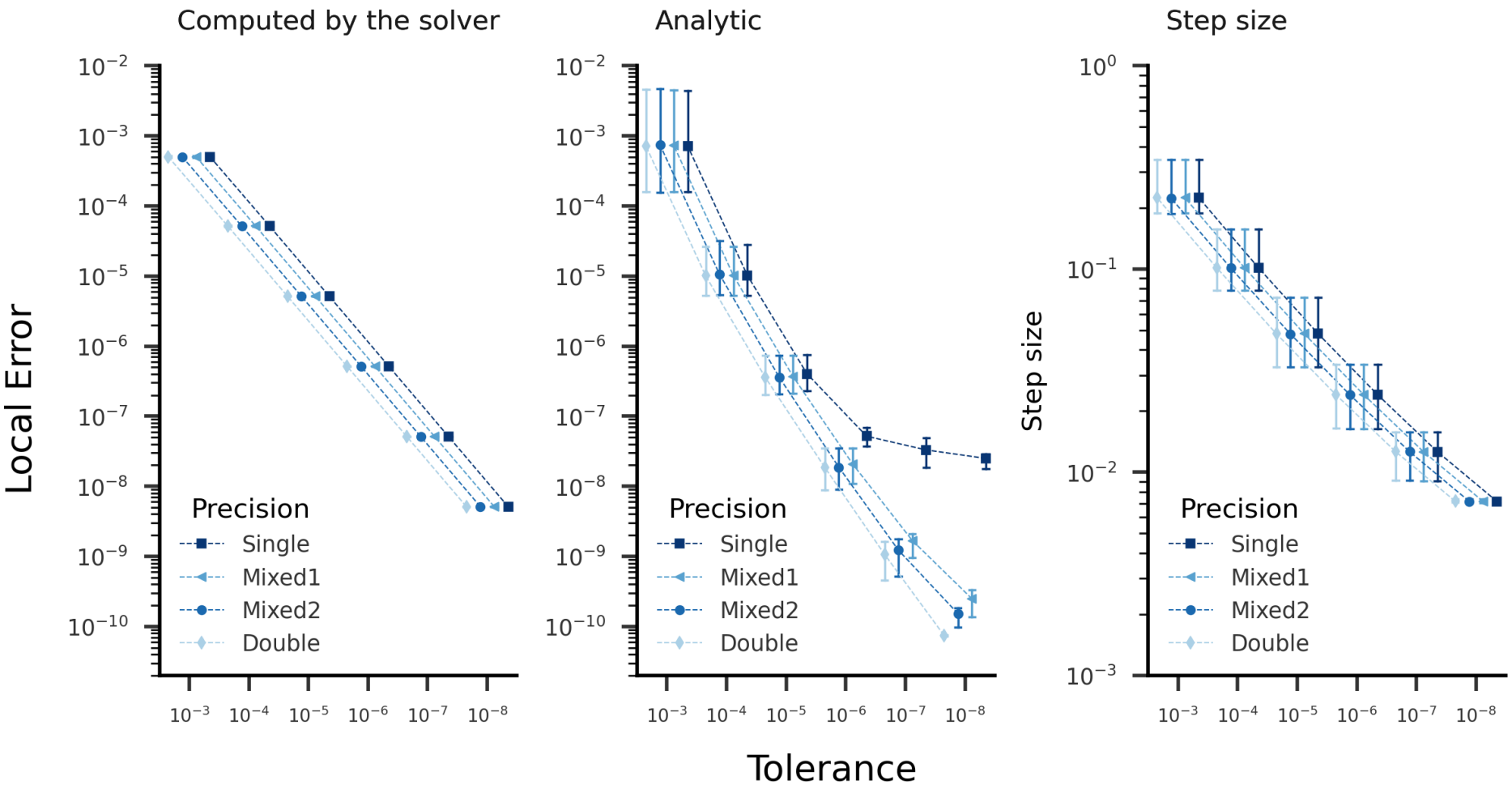
Error vs Size



Γ parameter on Linear coupled oscillators



Local Error and accumulation (Additional material)



Specificity of the *C++* implementation

- Store the solution in both formats (FP32 & FP64)
- Weighting Matrix supposed low rank
- Use *OpenMP* (**parallel** for over the agents, **SIMD** for one agent)

Relative tolerance

Tolerance and step validation

At each step $X_1^{(n)}$ and $X_2^{(n)}$ are computed, $X_2^{(n-1)}$ is the solution at previous step.

$$err := \left|\left|(X_1^{(n)} - X_2^{(n)})./ \max\left(|X_2^{(n)}|, |X_2^{(n-1)}|, \frac{Ab}{Rel}\right)\right|\right|_\infty.$$

'./' *division term by term*.

Normalized final error

Normalized final error

Reference solution, X_{ref}[a].

The normalized[b] final error at final time t_f:

$$||X_{ref}(t_f) - X(t_f)||_X = \frac{||X_{ref}(t_f) - X(t_f)||_2}{\sqrt{N}},$$

[a] Computed with Matlab solver `ODE45` using the 5(4) Dormand-Prince pair with a 10^{-9} relative tolerance

[b] The normalization by the size of the system enables to facilitate the comparison of the error on each element with respect to the theoretical error and for different sizes.

Real local error

Real local error

$$E^n_{analytic} = \max_k \left(\left| \frac{X_k^{n+1} - X_{k,ex}(X^n, h_n)}{X_{k,ex}(X^n, h_n)} \right| \right), k \in \{1, ..., dN\}.$$

where $h_n = t_{n+1} - t_n$, X_{ex} analytic solution at t_n with X_n as initial condition and computed in high precision.