





Beyond mixed precision: a short introduction to adaptive precision computing

Theo Mary
Sorbonne Université, CNRS, LIP6

TERATEC 2026
Paris, 20 October 2026

Why mix precisions?

	Precision	Speed*	
FP64	$\sim 10^{-16}$	33	4×
FP32	$\sim 10^{-8}$	130	30×
FP16	$\sim 10^{-4}$	4,000	4×
FP8	$\sim 10^{-2}$	17,500	2×
FP4	$\sim 10^{-1}$	35,000	

*TFLOPS, for NVIDIA Rubin

- Less precision $\equiv$ more performance, but also $\equiv$ less accuracy

$\Rightarrow$ Mixing precisions can get both performance and accuracy?

How to mix precisions?

- Traditional mixed precision algorithms limited to a small number of precisions fixed in advance (Higham and M., Acta Numerica 2022)
- Prototypical example: iterative refinement for $Ax = b$

Repeat	
$r = b - Ax$	in high precision
Solve $Ad \approx r$	in low precision
$x = x + d$	in high precision
Until convergence	

- Can work very well on some problems, but can also fail badly on other problems
  $\Rightarrow$ hard to "industrialize" (make usable as black box tool with no expert guidance)

How to mix precisions?

- Traditional mixed precision algorithms limited to a small number of precisions fixed in advance (Higham and M., Acta Numerica 2022)
- Prototypical example: iterative refinement for $Ax = b$

Repeat	
$r = b - Ax$	in high precision
Solve $Ad \approx r$	in low precision
$x = x + d$	in high precision
Until convergence	

- Can work very well on some problems, but can also fail badly on other problems $\Rightarrow$ hard to "industrialize" (make usable as black box tool with no expert guidance)

$\Rightarrow$ In this talk, I advocate for an evolution of mixed precision algorithms towards a more expressive and powerful paradigm: adaptive precision

Beyond mixed precision: adaptive precision

- Adaptive precision operates a paradigm shift: rather than computing the result in low precision and then *hope* it converges to high accuracy,

> use low precision only when and where it is safe to do so

- Adaptive precision is very "industrializable": guaranteed to work no matter what, will reduce cost whenever possible automatically and without any user guidance

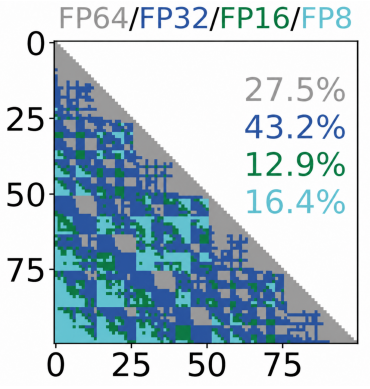
- Easy for the user, harder for the developer:
  - Where: requires partitioning computation into several parts (cleverly chosen)
  - When: requires dynamically adapting precision at runtime

Part I

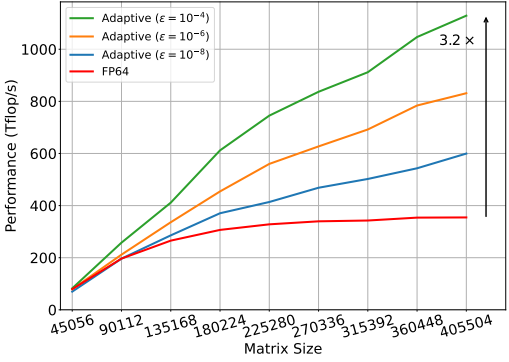
Example 1: direct solvers

AP tile factorization (Cao, M., Alomairy, Ltaief, Keyes, 2026)

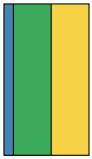
3D geostatistical squared exponential covariance matrix

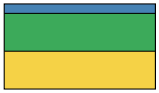


- Mixed precision tile matrix factorization: switch to low precision only tiles that are safe
- Which tiles are safe? Error analysis shows that tile precision should be chosen inversely proportional to tile norm



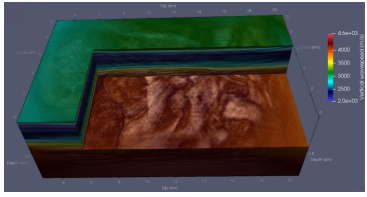
AP low-rank factorization (Amestoy, Boiteau, Buttari, Gerest, Jézéquel, L'Excellent, M., 2022)





- AP block low-rank (BLR): compress each block with a mixed precision low-rank approximation, using low precision for safe low-rank components
- Error analysis shows that precision should be chosen inversely proportional to singular values

Illustrative application impact (Operto et al., 2023):
Gorgon Model, reservoir 23km x 11km x 6.5km, grid size 15m, Helmholtz equation, 25-Hz
Complex matrix, 531 Million dofs
MUMPS solver: FP32 Full-Rank (FR) vs adaptive BLR ($\varepsilon = 10^{-5}$)
48 000 cores (500 MPI $\times$ 96 threads/MPI)



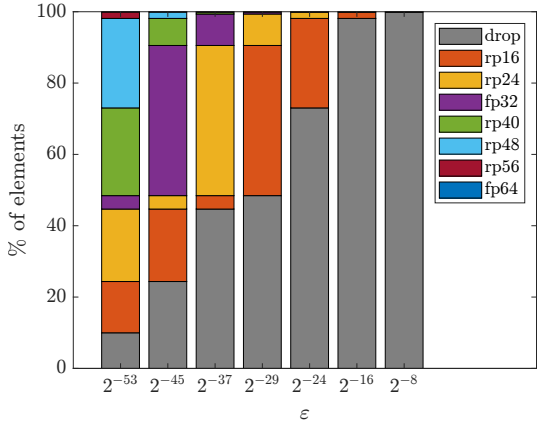
LU size (TBytes)			Flops		Time BLR + Mixed (sec)			Scaled Resid.
FR	BLR	+adapt.	FR	BLR+adapt.	Analysis	Facto	Solve	BLR+adapt.
73	34	26	2.6×10^{18}	0.5×10^{18}	446	5500	27	7×10^{-4}

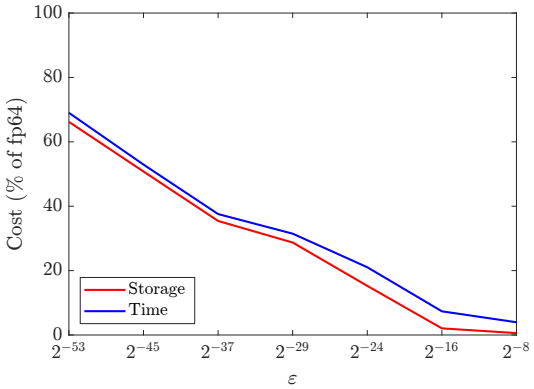
Part II

Example 2: iterative solvers

- AP sparse matrix–vector product: store each matrix coefficient in a different precision. Error analysis shows that precision should be inversely proportional to the coefficients' magnitude

Long_Coup_dt0 matrix



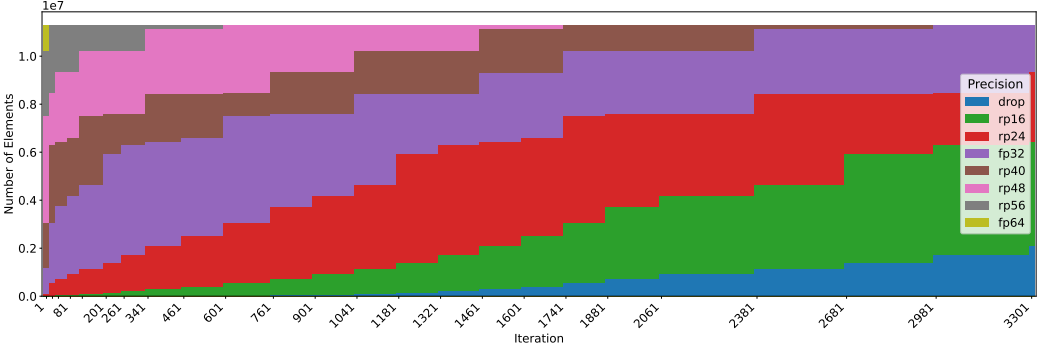


AP GMRES (Agullo, Giraud, Jolivet, M., Tabouret, 2026)

- How should the SpMV accuracy ε be selected within GMRES?
- First idea: set ε to the iterative stopping tolerance.

AP GMRES (Agullo, Giraud, Jolivet, M., Tabouret, 2026)

- How should the SpMV accuracy ε be selected within GMRES?
- First idea: set ε to the iterative stopping tolerance.
- Even better: relaxed GMRES theory (Giraud, Gratton, Langou, 2009) shows that SpMV accuracy should be inversely proportional to the residual norm $\Rightarrow$ lower and lower precision as iterations progress



$\Rightarrow$ up to $2\times$ speedup wrt fp64 GMRES