جامعة الملك عبدالله
للعلوم والتقنية
King Abdullah University of
Science and Technology

HPC in Healthcare

Hatem Ltaief
Principal Research Scientist, KAUST
CTO, AlgoDoers

Acknowledgments

Academic Collaborators:

KAUST: R. Alomairy (now at MIT), S. Bougoffa, D. E. Keyes, and J. Ren
UTK: Q. Cao (now at St Louis University)

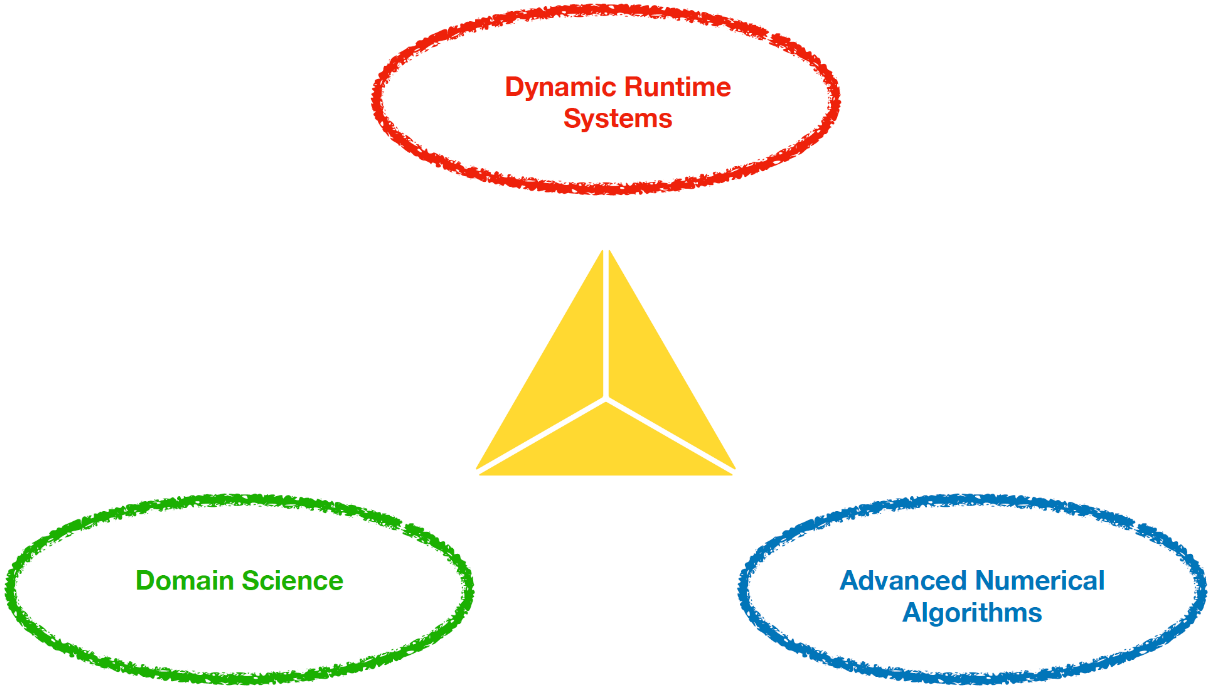
Vendor Collaborators:

NVIDIA: R. Abdelkhalek, T. Kurth, G. Paciucci, D. Ruau, and L. Slim

Resource Allocations:

Shaheen-2/3 and Ibex @ KAUST Supercomputing Lab, Saudi Arabia
HAWK @ HLRS, Germany (M. Resch)
Frontier @ ORNL, USA (US DOE)
Summit @ ORNL, USA (US DOE)
Fugaku @ Riken, Japan (S. Matsuoka)
Leonardo @ CINECA, Italy (G. Scipione)
Fromage @ NVIDIA, UK (F. Spiga)

Key Approach Based on a Separation of Concerns

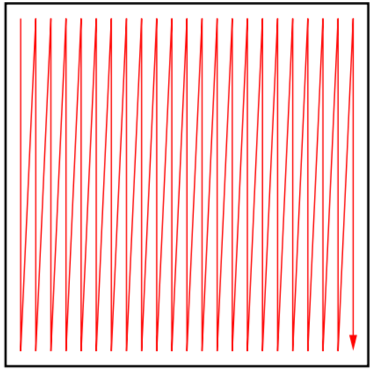


Dynamic Runtime Systems

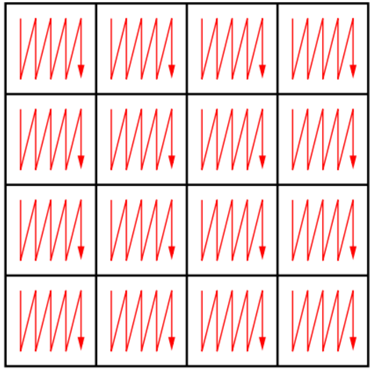
Domain Science

Advanced Numerical Algorithms

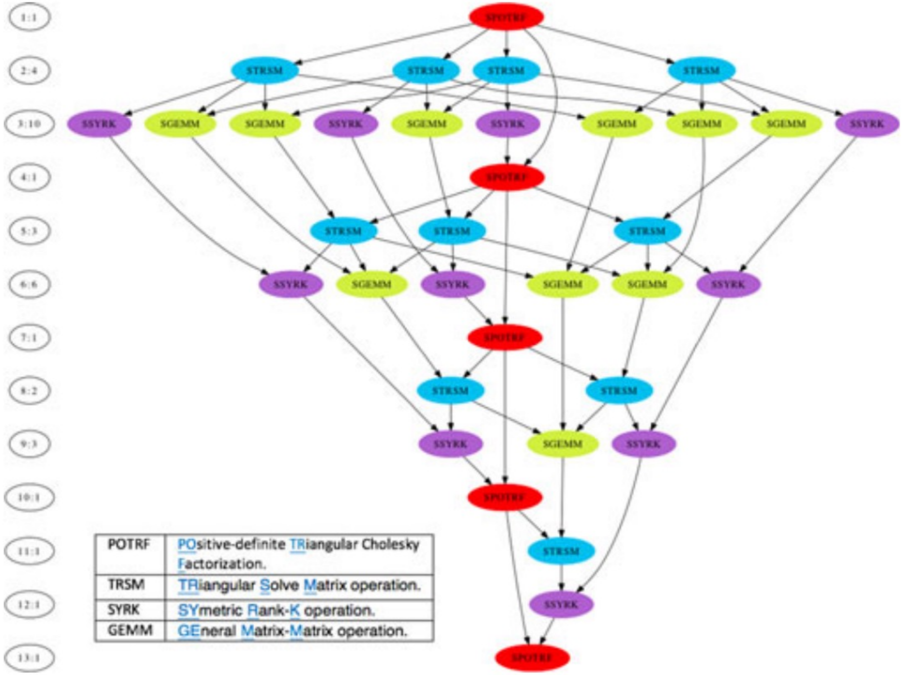
DAG Asynchronous Scheduling



LAPACK: Column-major data layout format.



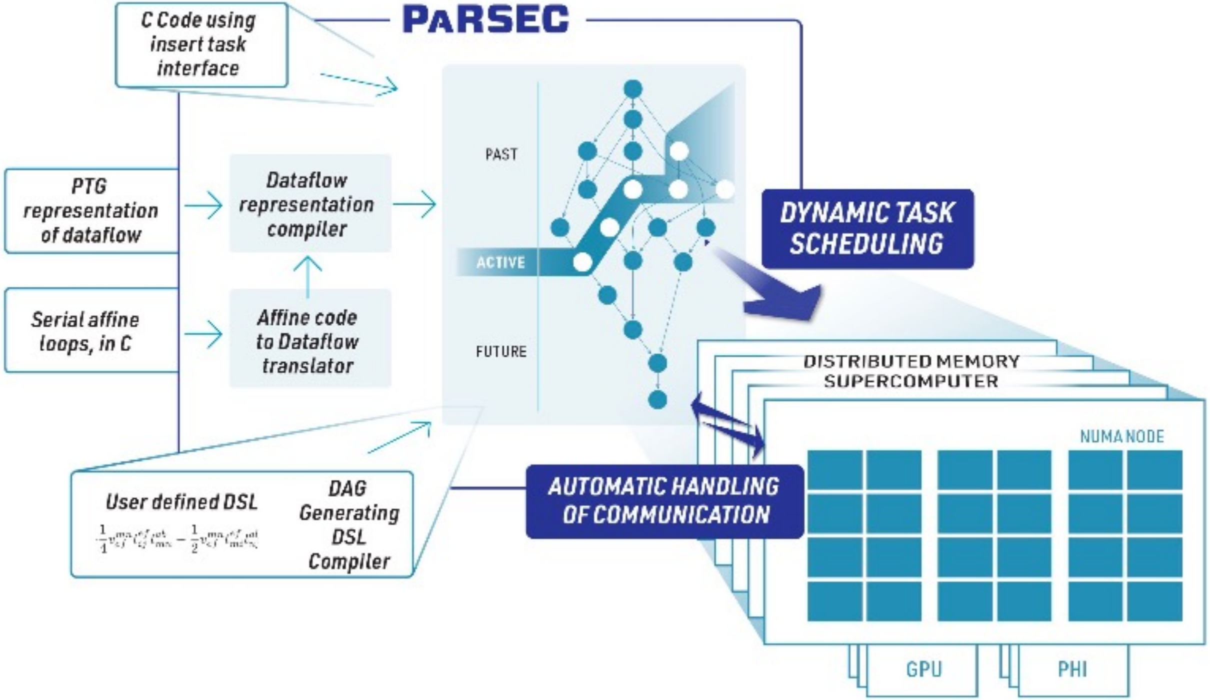
Chameleon: Tile data layout format.



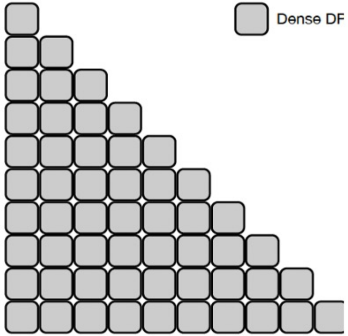
POTRF	POsitive-definite TRiangular Cholesky Factorization.
TRSM	TRiangular Solve Matrix operation.
SYRK	SYmetric Rank-K operation.
GEMM	GEneral Matrix-Matrix operation.

Cholesky factorization DAG

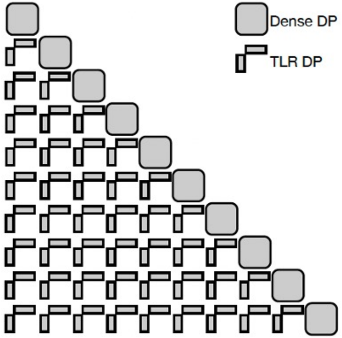
The ECP PaRSEC Dynamic Runtime System


PaRSEC
C Code using insert task interface
PTG representation of dataflow
Dataflow representation compiler
Serial affine loops, in C
Affine code to Dataflow translator
User defined DSL
DAG Generating DSL Compiler
PAST
ACTIVE
FUTURE
DYNAMIC TASK SCHEDULING
AUTOMATIC HANDLING OF COMMUNICATION
DISTRIBUTED MEMORY SUPERCOMPUTER
NUMA NODE
GPU
PHI

Tile-Centric Matrix Approximations in ExaGeoStat

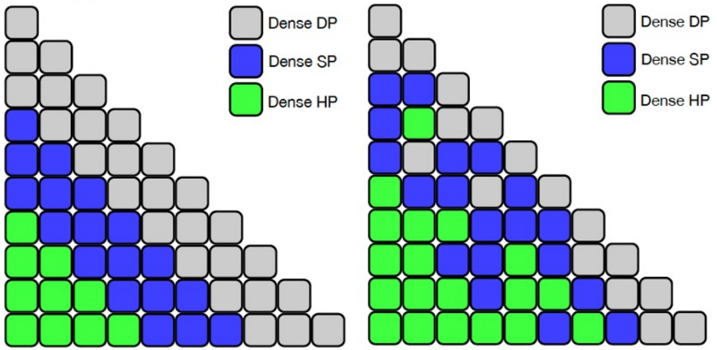


Exact Computation



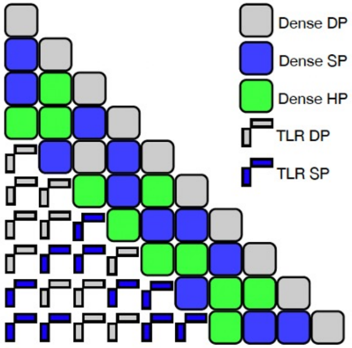
Tile Low-Rank (TLR)
Computation

Reduce memory footprint
ACM PASC’20
ACM SC’23 GB Finalist



Mixed-precision (MP)
Higham and Mary, 2021

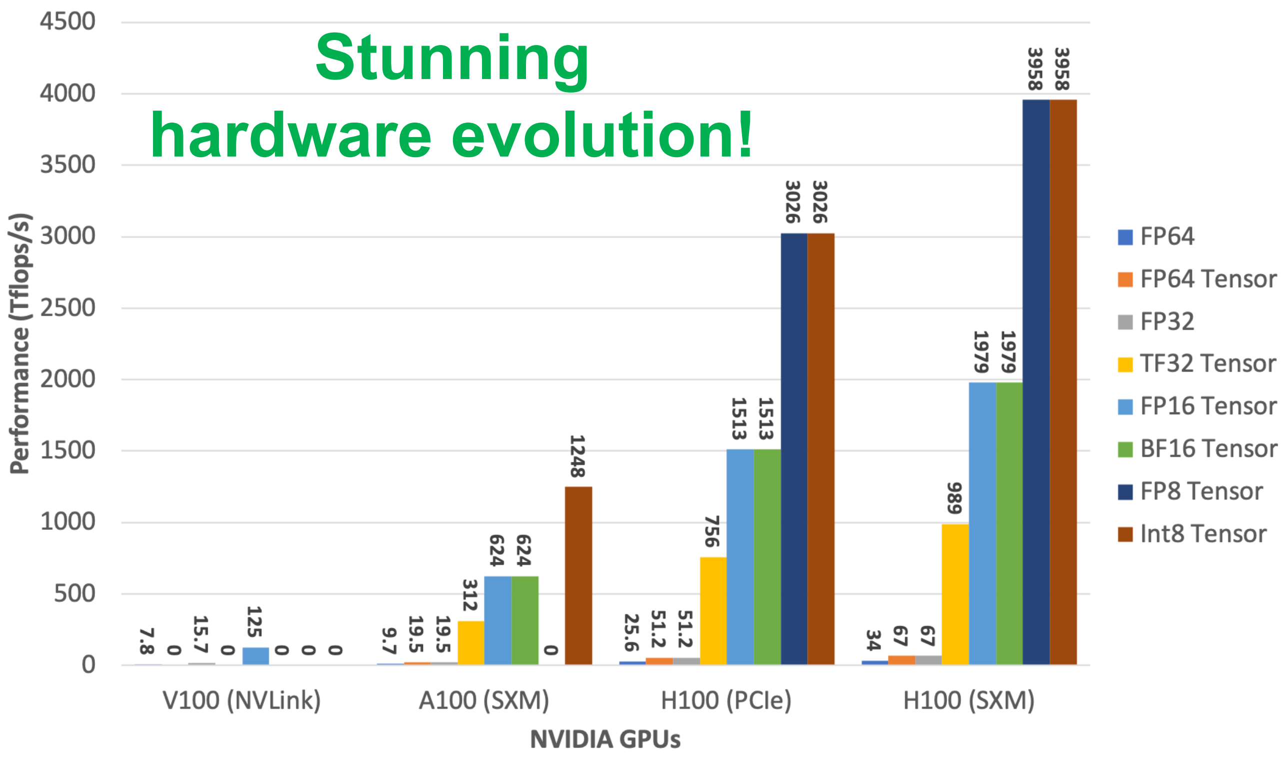
Increase arithmetic intensity
IEEE TPDS’22
IEEE Cluster’23
ACM SC’24 GB Finalist



MP + TLR

Combine best
of the two worlds
ACM SC’22 GB Finalist

Peak Performance of NVIDIA GPUs (Tflops/s)



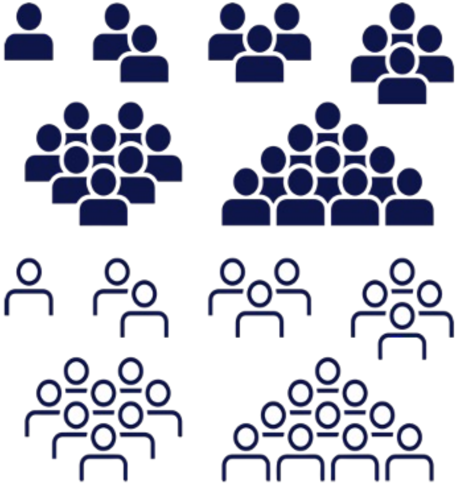
FP4 Tensor Core: 14 Pflops/s
FP8/FP6 Tensor Core: 7 Pflops/s
INT8 Tensor Core: 7 POPs
FP16/BF16 Tensor Core: 3.5 Pflops/s
TF32 Tensor Core: 1.8 Pflops/s
FP64 Tensor Core: 60 Pflops/s

B200

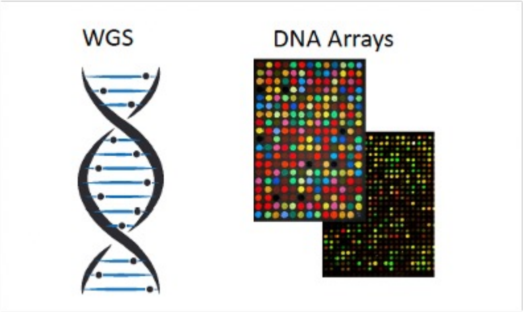
https://images.nvidia.com/content/technologies/volta/pdf/tesla-volta-v100-datasheet-letter-fnl-web.pdf
https://www.nvidia.com/content/dam/en-zz/Solutions/Data-Center/a100/pdf/nvidia-a100-datasheet-us-nvidia-1758950-r4-web.pdf
https://resources.nvidia.com/en-us-tensor-core/nvidia-tensor-core-gpu-datasheet
https://resources.nvidia.com/en-us-grace-cpu/grace-hopper-superchip

Genome-Wide Association Study

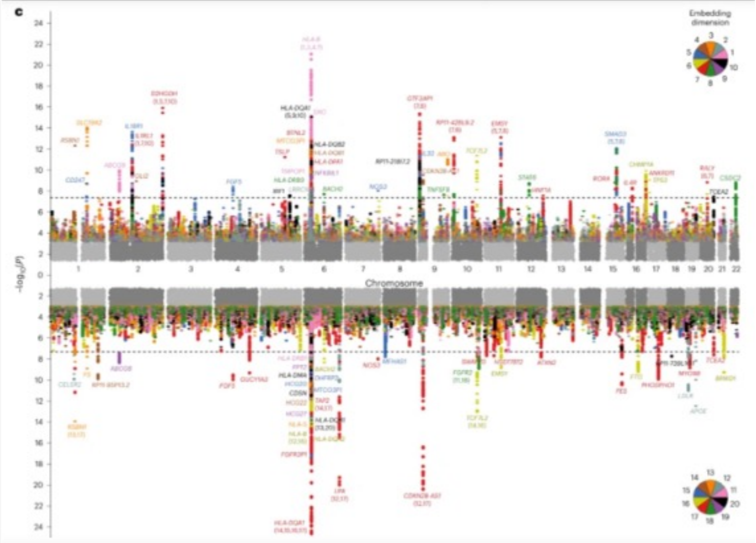
Population



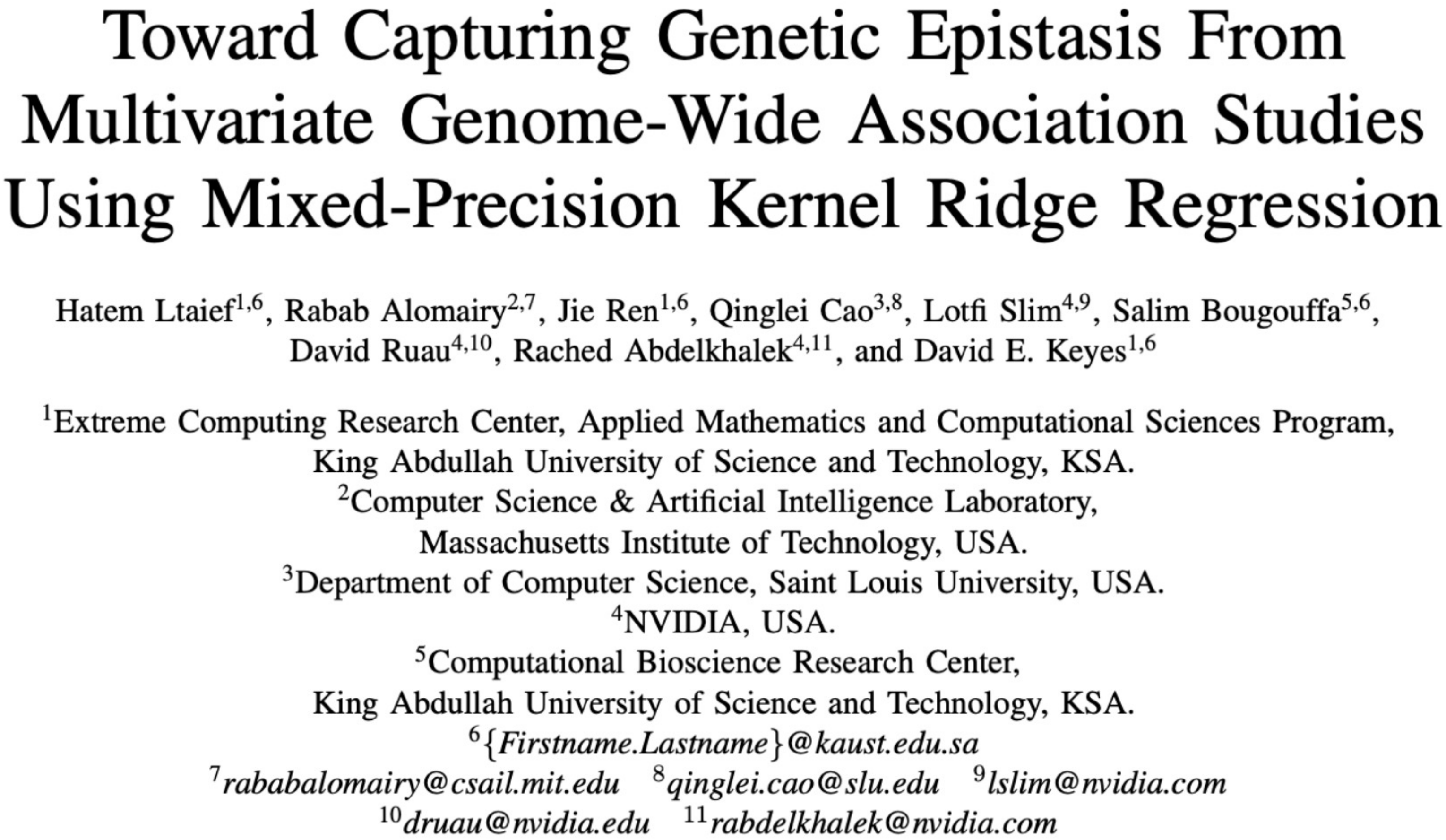
Genotyping



Statistical association



Genome-Wide Association Study: GB entry @ SC24

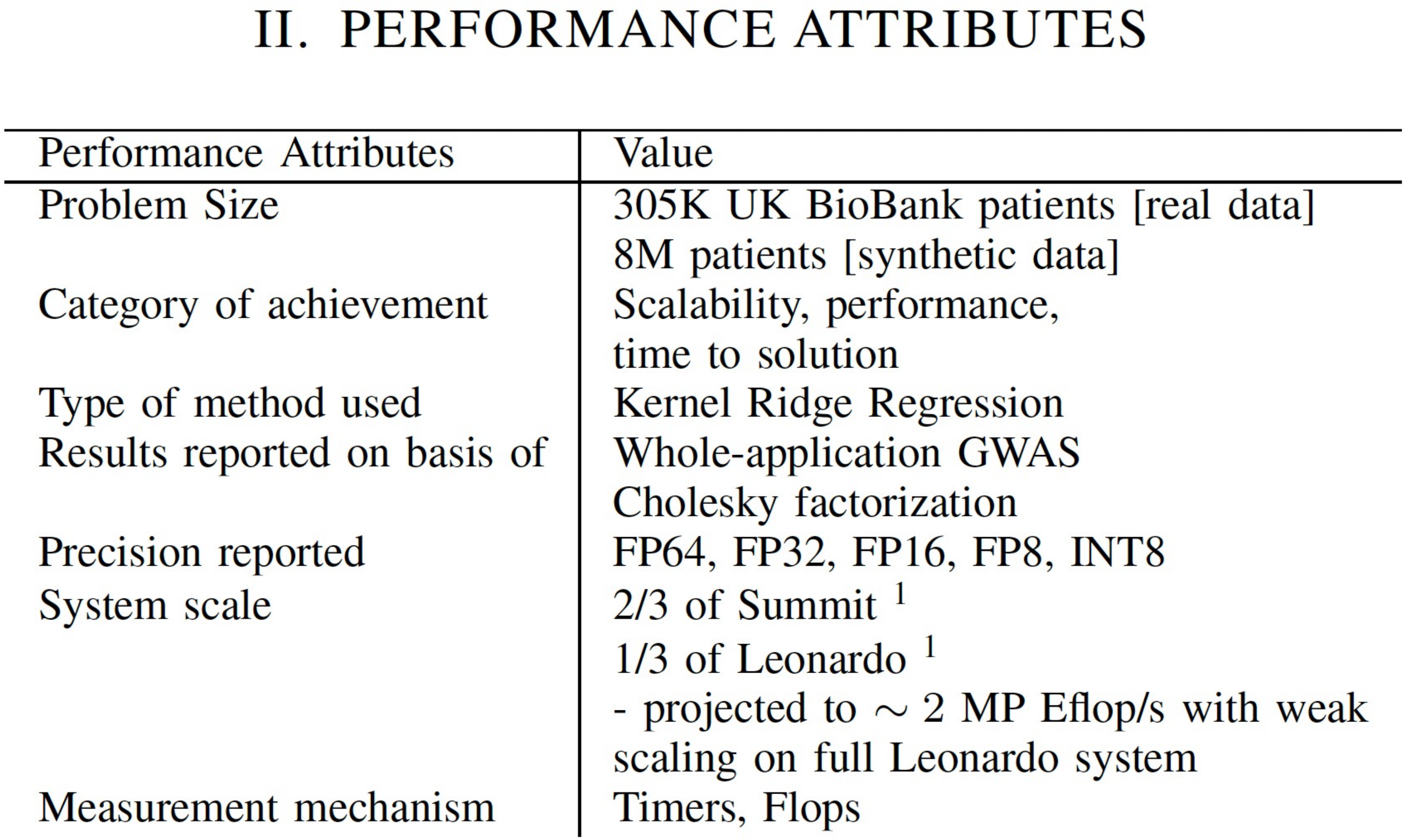


Toward Capturing Genetic Epistasis From Multivariate Genome-Wide Association Studies Using Mixed-Precision Kernel Ridge Regression

Hatem Ltaief[1,6], Rabab Alomairy[2,7], Jie Ren[1,6], Qinglei Cao[3,8], Lotfi Slim[4,9], Salim Bougouffa[5,6], David Ruau[4,10], Rached Abdelkhalek[4,11], and David E. Keyes[1,6]

[1]Extreme Computing Research Center, Applied Mathematics and Computational Sciences Program,
King Abdullah University of Science and Technology, KSA.
[2]Computer Science & Artificial Intelligence Laboratory,
Massachusetts Institute of Technology, USA.
[3]Department of Computer Science, Saint Louis University, USA.
[4]NVIDIA, USA.
[5]Computational Bioscience Research Center,
King Abdullah University of Science and Technology, KSA.
[6]*{Firstname.Lastname}@kaust.edu.sa*
[7]*rababalomairy@csail.mit.edu* [8]*qinglei.cao@slu.edu* [9]*lslim@nvidia.com*
[10]*druau@nvidia.edu* [11]*rabdelkhalek@nvidia.com*

Genome-Wide Association Study: GB entry @ SC24



II. PERFORMANCE ATTRIBUTES

Performance Attributes	Value
Problem Size	305K UK BioBank patients [real data] 8M patients [synthetic data]
Category of achievement	Scalability, performance, time to solution
Type of method used	Kernel Ridge Regression
Results reported on basis of	Whole-application GWAS Cholesky factorization
Precision reported	FP64, FP32, FP16, FP8, INT8
System scale	2/3 of Summit [1] 1/3 of Leonardo [1] - projected to $\sim$ 2 MP Eflop/s with weak scaling on full Leonardo system
Measurement mechanism	Timers, Flops

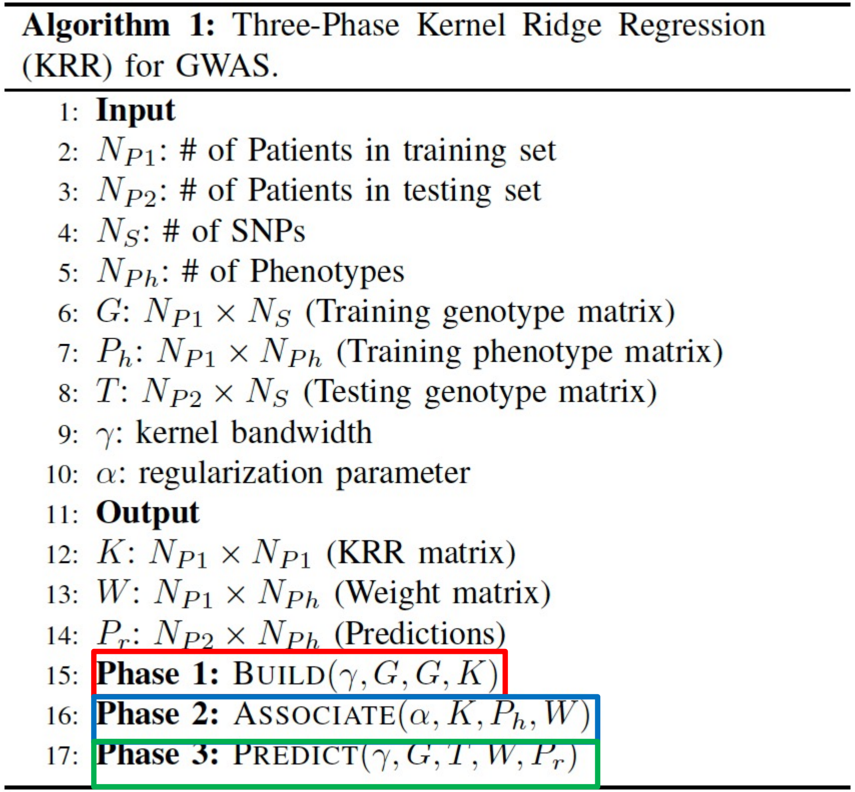
Overview of the GWAS Problem

- Analyze DNA sequence variations spanning an entire genome
- Identify genetic risk factors for common diseases or other traits within a population
- Use genetic factors to make predictions about individuals at risk and to identify the biological underpinnings of disease
- Expose big data challenges: Genotypes (million of SNPs) >> Phenotypes (hundreds of diseases)

State-Of-The-Art

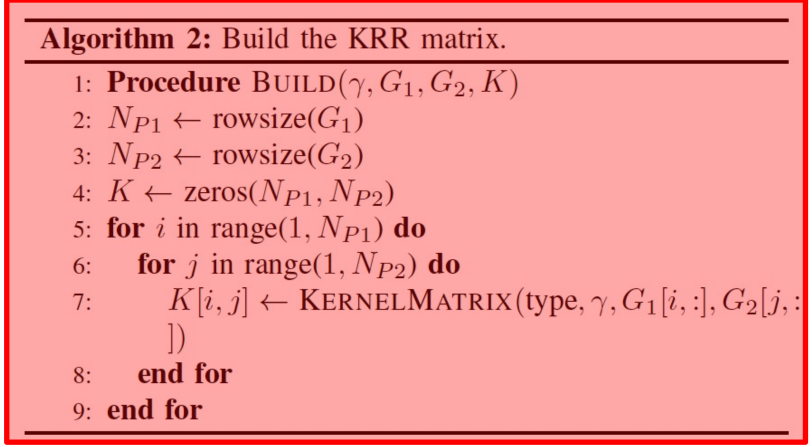
- Use linear models: overfitting issues, accuracy (ill-conditioned matrix). Penalized regression approaches come to rescue, e.g., ridge regression and LASSO
- Capture the nonlinear nature of genotype-phenotype relationships, i.e., epistasis (interactions between distant loci), gene-environment interactions, and non-additive genetic effects
- Transform the input data into a higher-dimensional feature space where nonlinear relationships can be more effectively captured and modeled
- Democratize Kernel Ridge Regression (KRR) for GWAS

General Algorithms



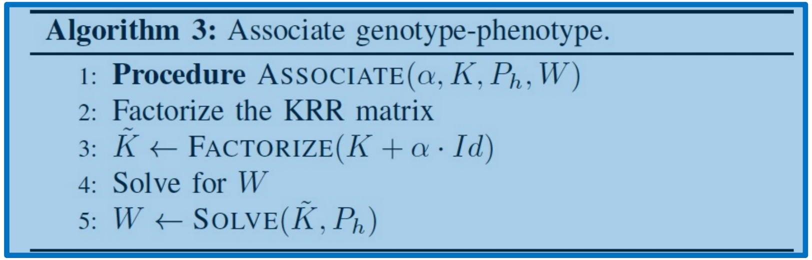
Algorithm 1: Three-Phase Kernel Ridge Regression (KRR) for GWAS.

```
**Input**
$N_{P1}$: # of Patients in training set
$N_{P2}$: # of Patients in testing set
$N_S$: # of SNPs
$N_{Ph}$: # of Phenotypes
$G$: $N_{P1} \times N_S$ (Training genotype matrix)
$P_h$: $N_{P1} \times N_{Ph}$ (Training phenotype matrix)
$T$: $N_{P2} \times N_S$ (Testing genotype matrix)
$\gamma$: kernel bandwidth
$\alpha$: regularization parameter
**Output**
$K$: $N_{P1} \times N_{P1}$ (KRR matrix)
$W$: $N_{P1} \times N_{Ph}$ (Weight matrix)
$P_r$: $N_{P2} \times N_{Ph}$ (Predictions)
**Phase 1:** BUILD$(\gamma, G, G, K)$
**Phase 2:** ASSOCIATE$(\alpha, K, P_h, W)$
**Phase 3:** PREDICT$(\gamma, G, T, W, P_r)$
```



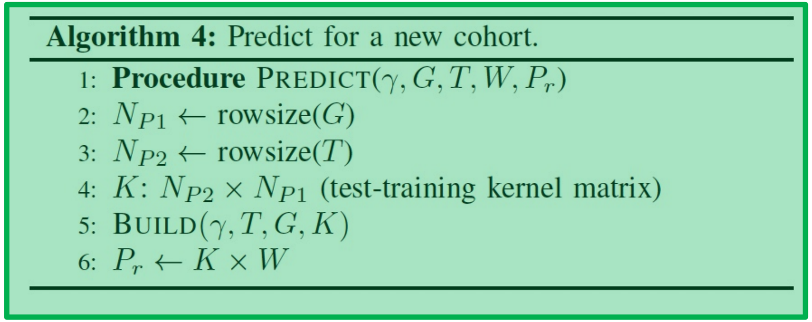
Algorithm 2: Build the KRR matrix.

```
**Procedure** BUILD$(\gamma, G_1, G_2, K)$
$N_{P1} \leftarrow$ rowsize$(G_1)$
$N_{P2} \leftarrow$ rowsize$(G_2)$
$K \leftarrow$ zeros$(N_{P1}, N_{P2})$
**for** $i$ in range$(1, N_{P1})$ **do**
    **for** $j$ in range$(1, N_{P2})$ **do**
        $K[i,j] \leftarrow$ KERNELMATRIX(type, $\gamma, G_1[i,:], G_2[j,:$
        ])
    **end for**
**end for**
```



Algorithm 3: Associate genotype-phenotype.

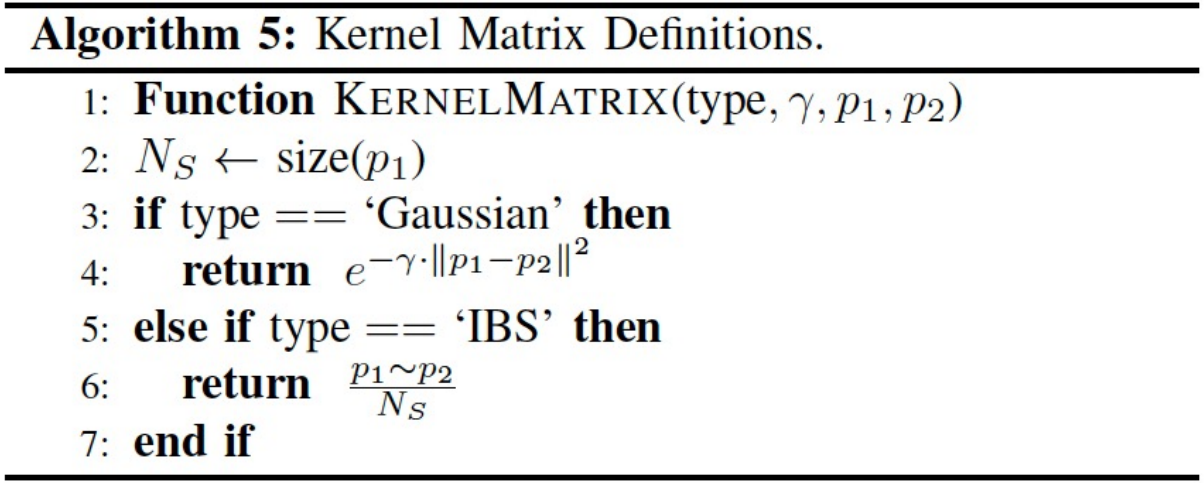
```
**Procedure** ASSOCIATE$(\alpha, K, P_h, W)$
Factorize the KRR matrix
$\tilde{K} \leftarrow$ FACTORIZE$(K + \alpha \cdot Id)$
Solve for $W$
$W \leftarrow$ SOLVE$(\tilde{K}, P_h)$
```



Algorithm 4: Predict for a new cohort.

```
**Procedure** PREDICT$(\gamma, G, T, W, P_r)$
$N_{P1} \leftarrow$ rowsize$(G)$
$N_{P2} \leftarrow$ rowsize$(T)$
$K$: $N_{P2} \times N_{P1}$ (test-training kernel matrix)
BUILD$(\gamma, T, G, K)$
$P_r \leftarrow K \times W$
```

The Build Phase



Algorithm 5: Kernel Matrix Definitions.

```
Function KernelMatrix(type, $\gamma$, $p_1$, $p_2$)
$N_S \leftarrow$ size($p_1$)
if type == 'Gaussian' then
    return $e^{-\gamma\cdot\|p_1-p_2\|^2}$
else if type == 'IBS' then
    return $\frac{p_1 \sim p_2}{N_S}$
end if
```

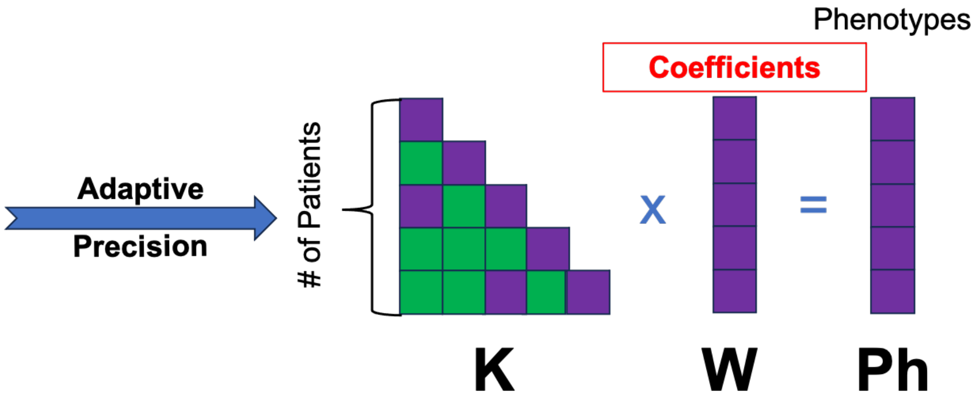
- Compute Euclidean distance between each pair of individual (slow)
- Exponent the results
- Generate the covariance matrix

The Associate Phase

🟩 FP16/FP8

🟪 FP64/FP32

Mixed-Precision Cholesky-based Solver

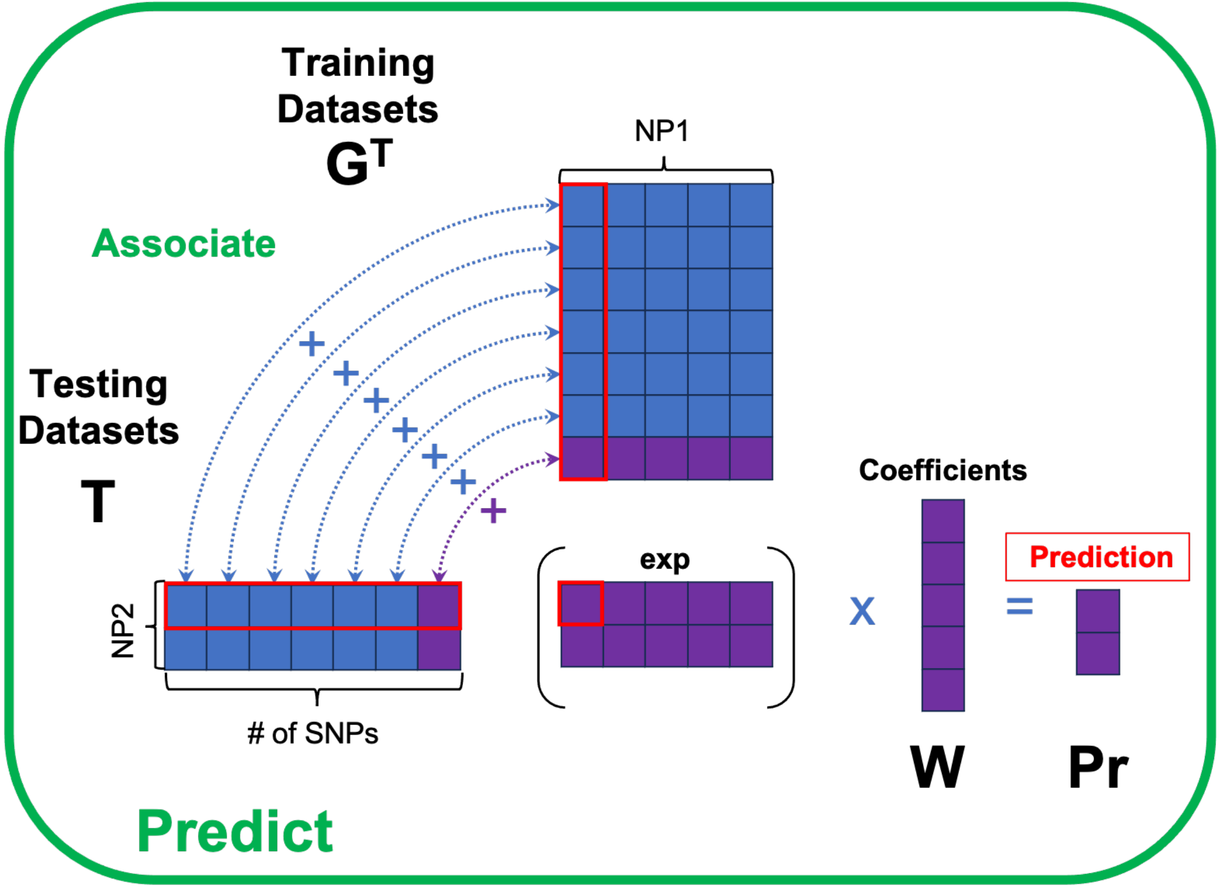


Associate

The Predict Phase

🟩 FP16/FP8

🟪 FP64/FP32



Training Datasets G^T

NP1

Associate

Testing Datasets

T

NP2

# of SNPs

exp

Coefficients

Prediction

X

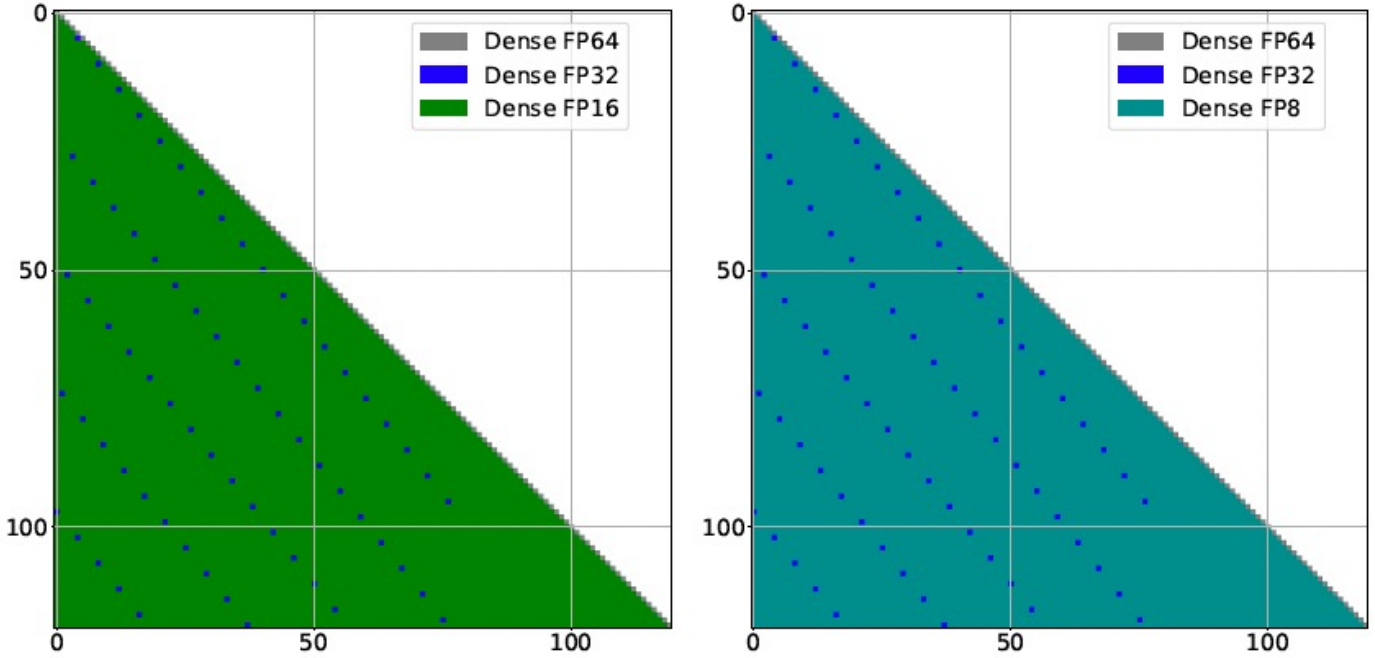
=

W

Pr

Predict

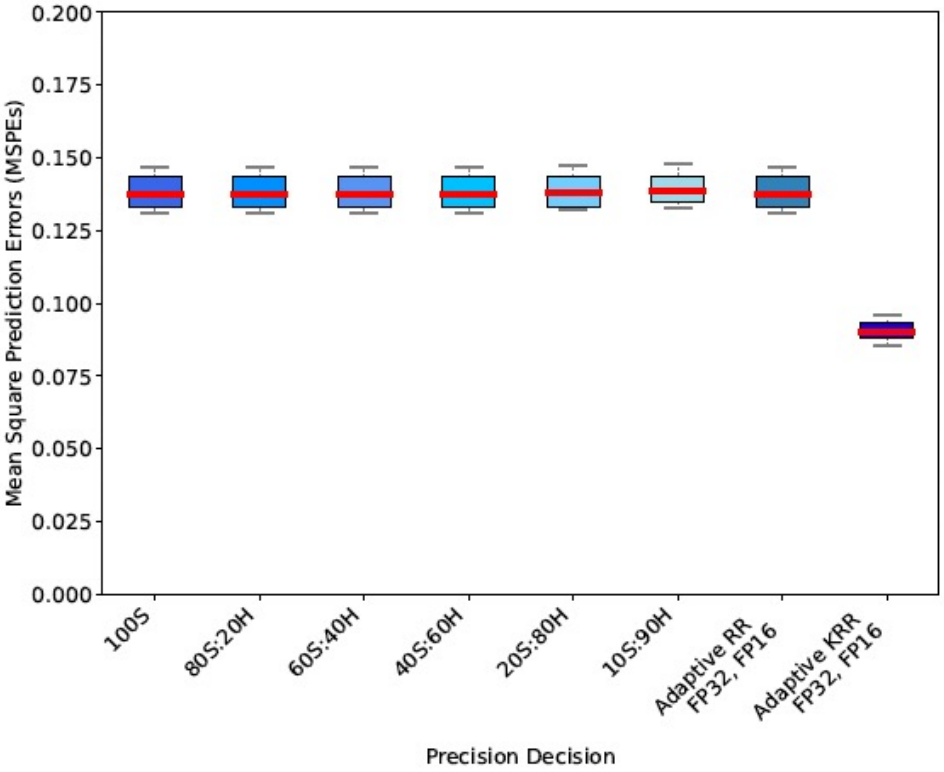
GWAS surfing the AI wave w/ low precision arithmetics



(a) Activating FP16 with A100. (b) Activating FP8 with H100.

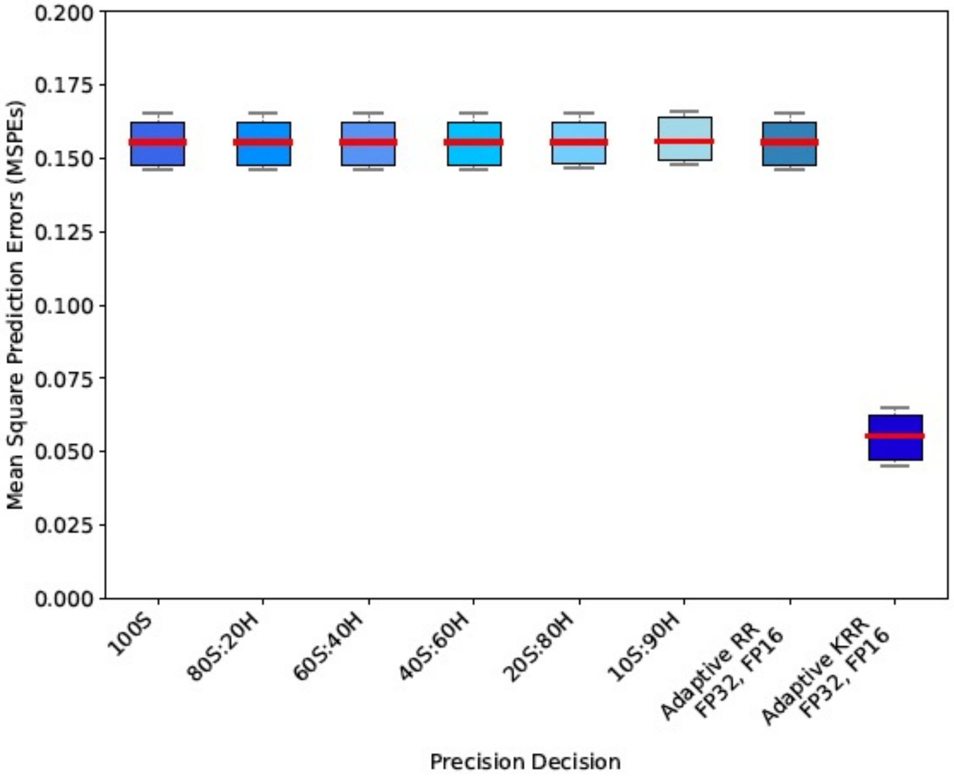
Fig. 4: Precision heatmaps.

300K Patients from UK BioBank: MSPE assessment



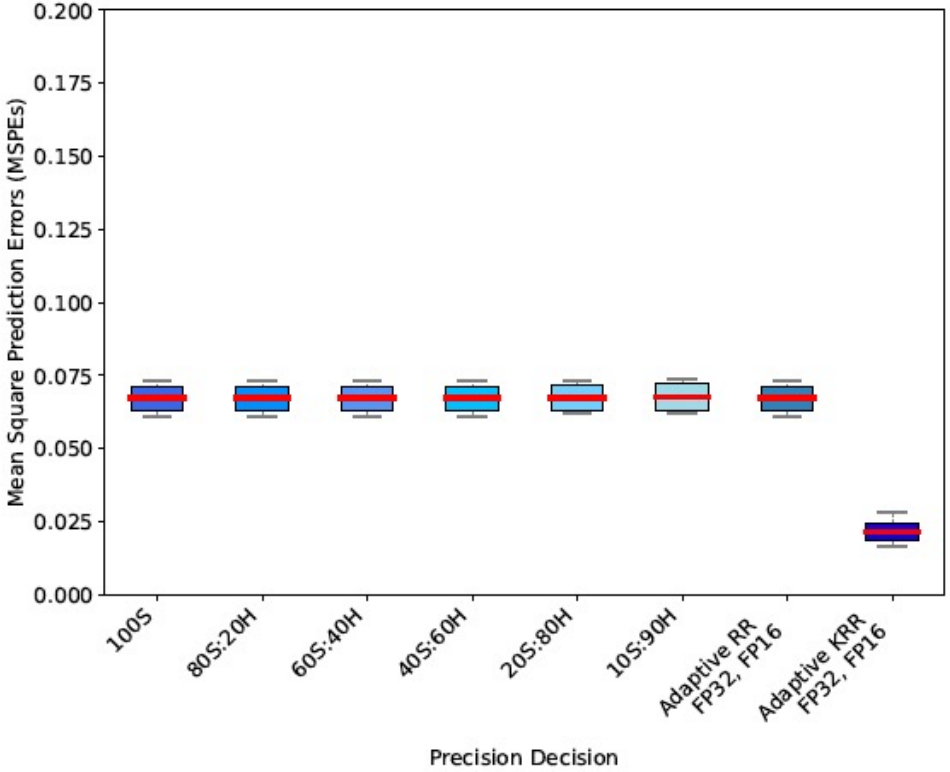
(a) RR vs KRR for Hypertension.

300K Patients from UK BioBank: MSPE assessment



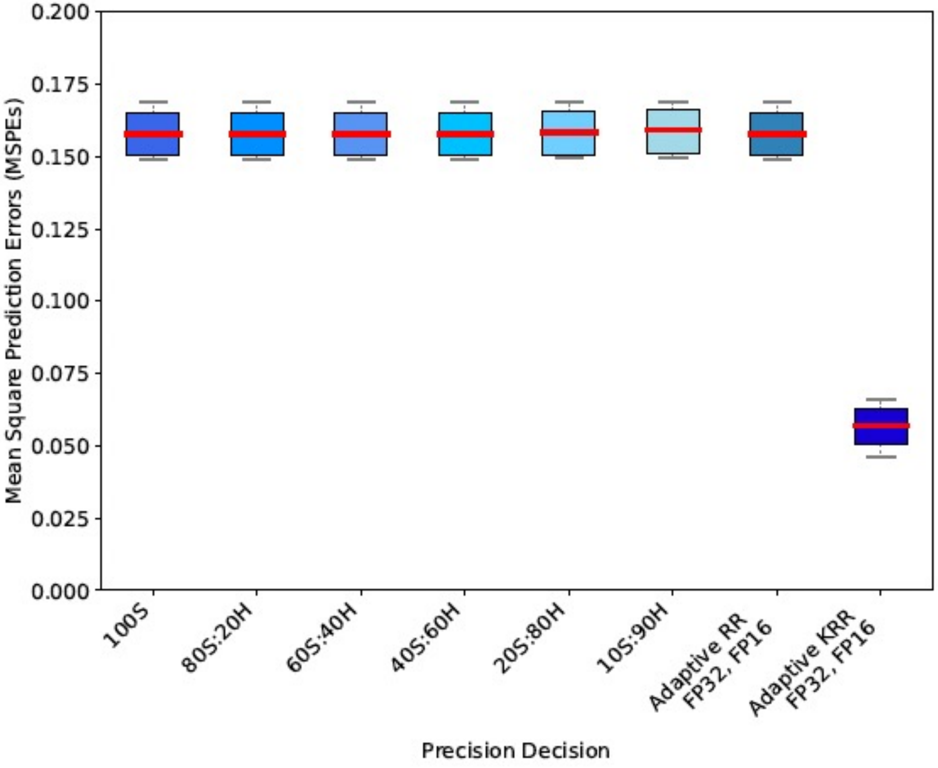
(b) RR vs KRR for Asthma.

300K Patients from UK BioBank: MSPE assessment



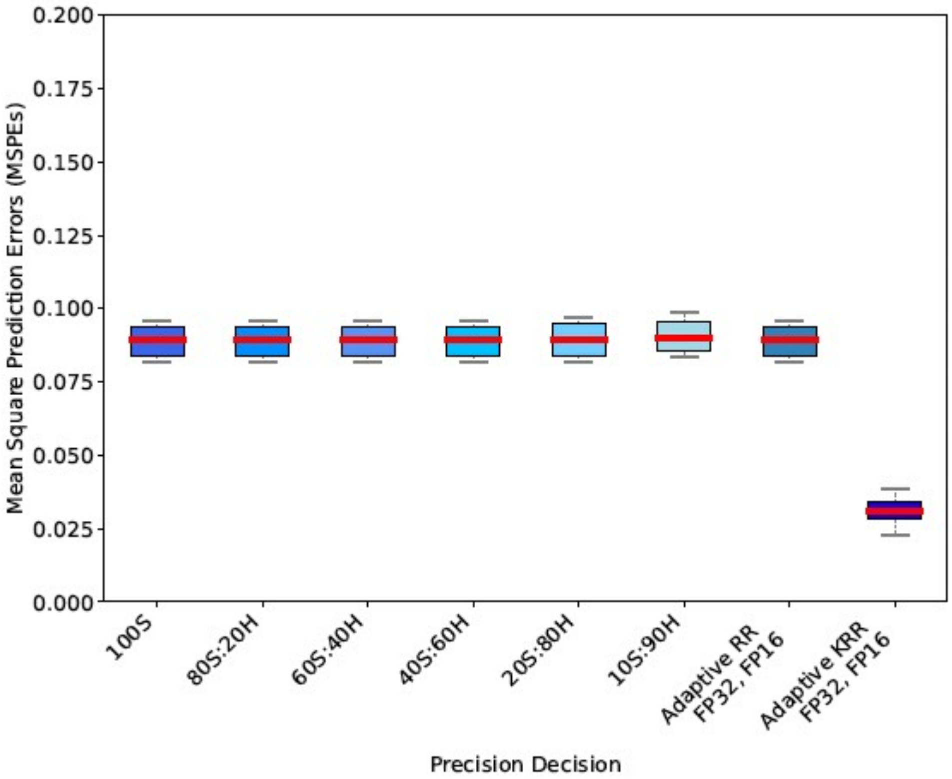
(c) RR vs KRR for Allergic Rhinitis.

300K Patients from UK BioBank: MSPE assessment



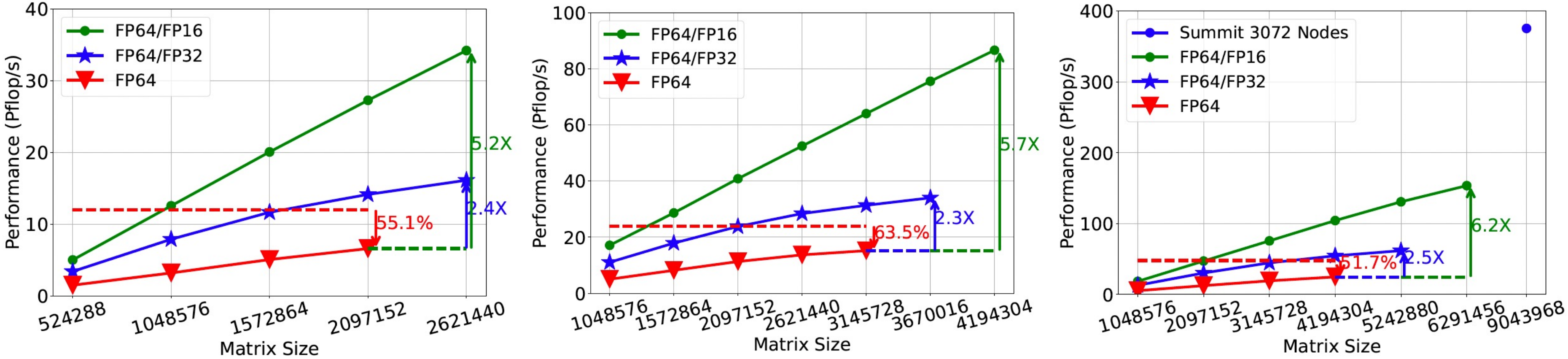
(d) RR vs KRR for Osteoarthritis.

300K Patients from UK BioBank: MSPE assessment



(e) RR vs KRR for Depression.

Performance Results: multi-node, multi GPU



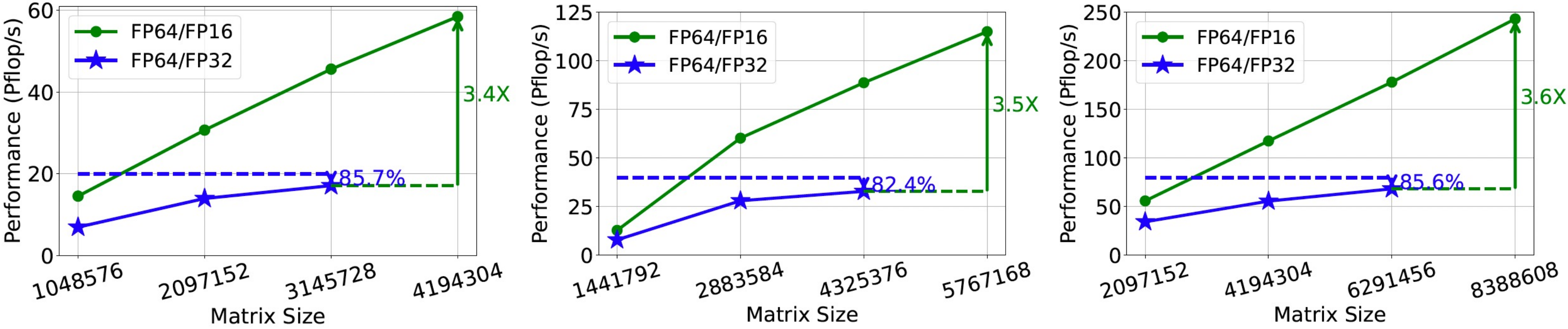
(a) 256 nodes.

(b) 512 nodes.

(c) 1024 nodes.

Fig. 9: Performance scalability of the `Associate` phase of the KRR-based GWAS ($N_P = N_S$) on `Summit`.

Performance Results: multi-node, multi GPU



(a) 256 nodes.

(b) 512 nodes.

(c) 1024 nodes.

Fig. 8: Performance scalability of the `Associate` phase for the KRR-based GWAS ($N_P = N_S$) on `Leonardo`.

Performance Results: strong scaling

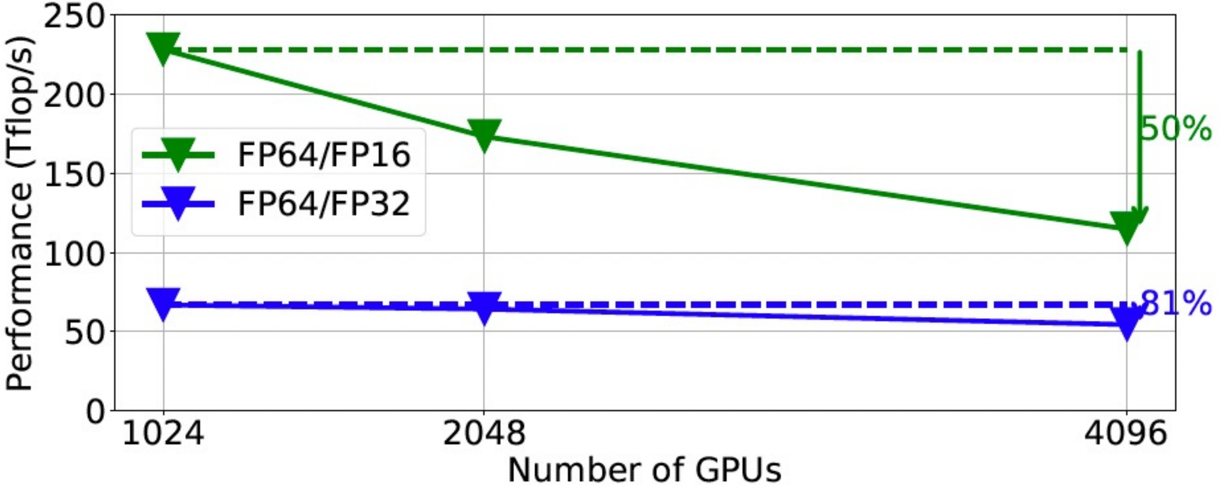


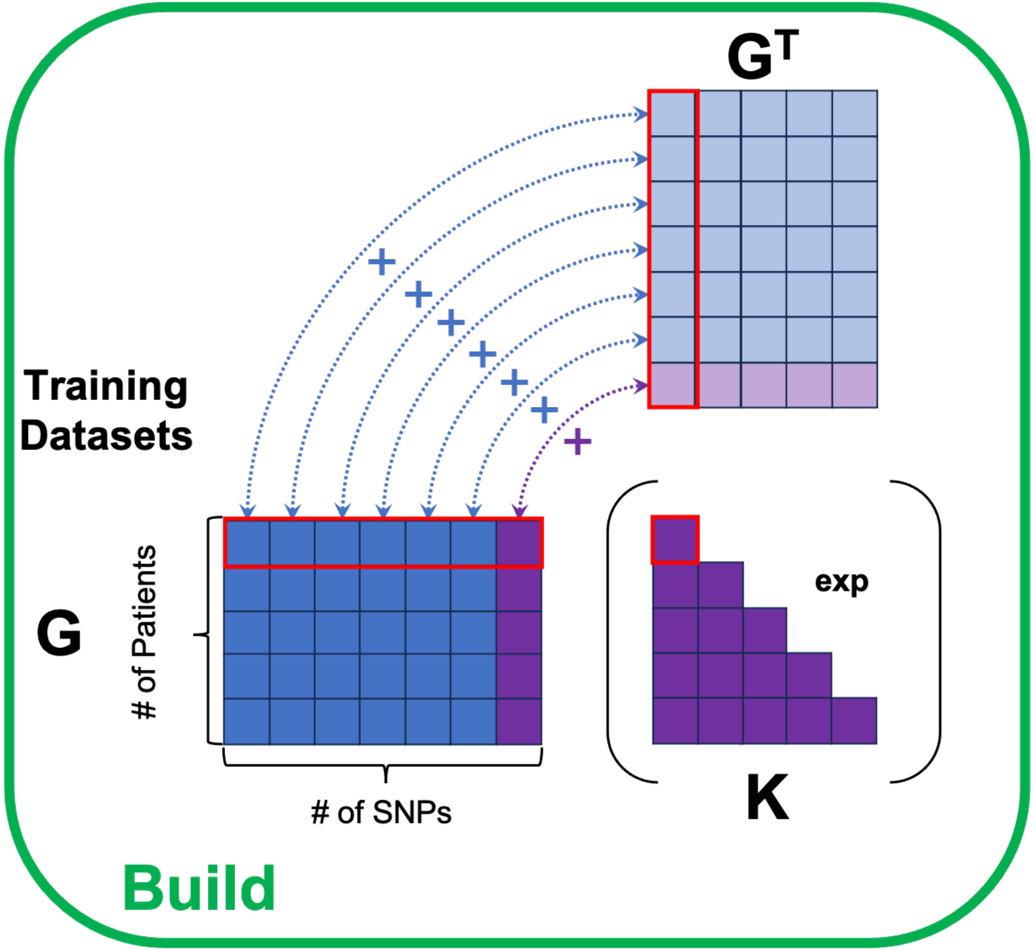
Fig. 11: Strong Scaling on Leonardo using various precision configurations, i.e., FP64/FP16 and FP64/FP16.

The Build Phase

🟦 INT8

🟩 FP16/FP8

🟪 FP64/FP32



B. Gallet and M. Gowanlock. Leveraging GPU Tensor Cores for Double Precision Euclidean Distance Calculations. IEEE HiPC, 2022.

Performance Results: single-node, multiple GPUs

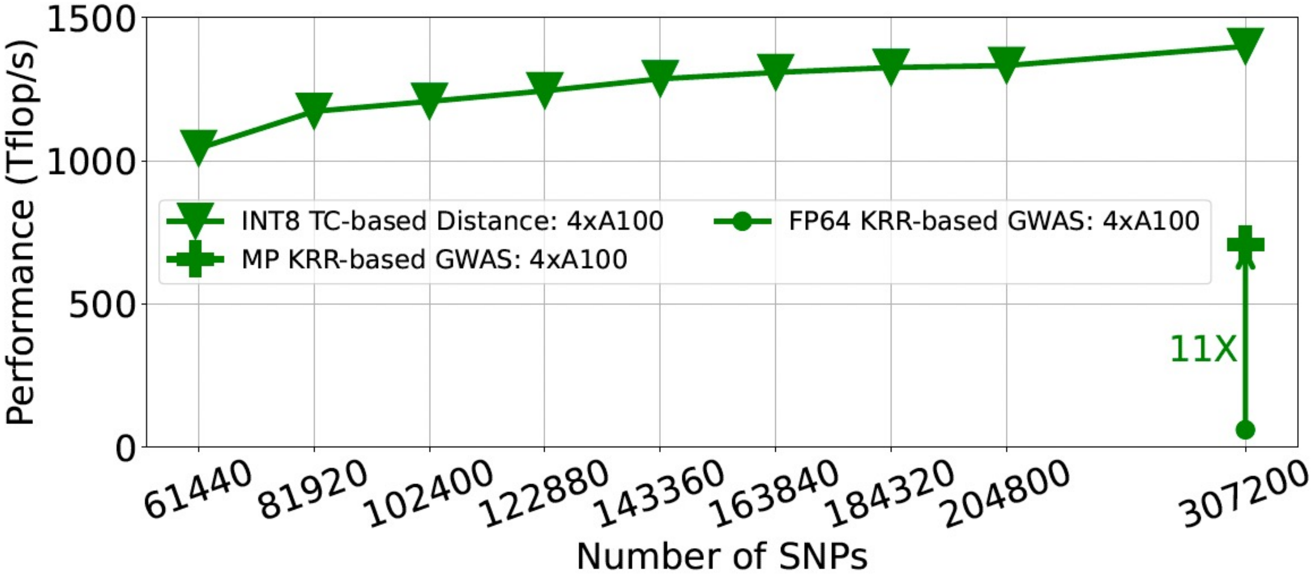


Fig. 6: Impact of # SNPs on distance kernel performance.

Performance Results: weak scaling

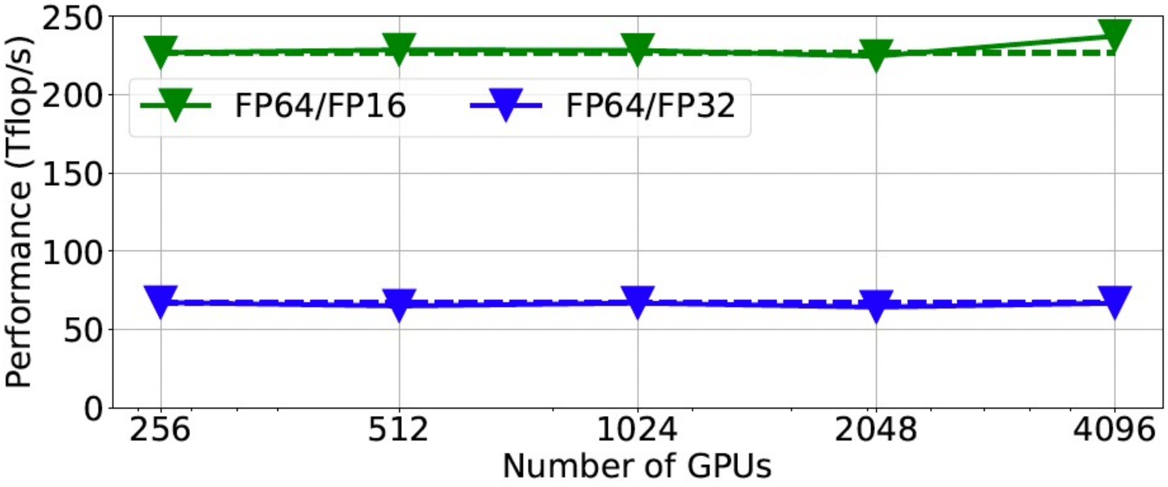


Fig. 10: Weak Scaling on Leonardo using various precision configurations, i.e., FP64/FP16 and FP64/FP16.

We expect 2 Eflop/s of sustained performance on fullscale Leonardo

Performance Results: single-node, single GPU

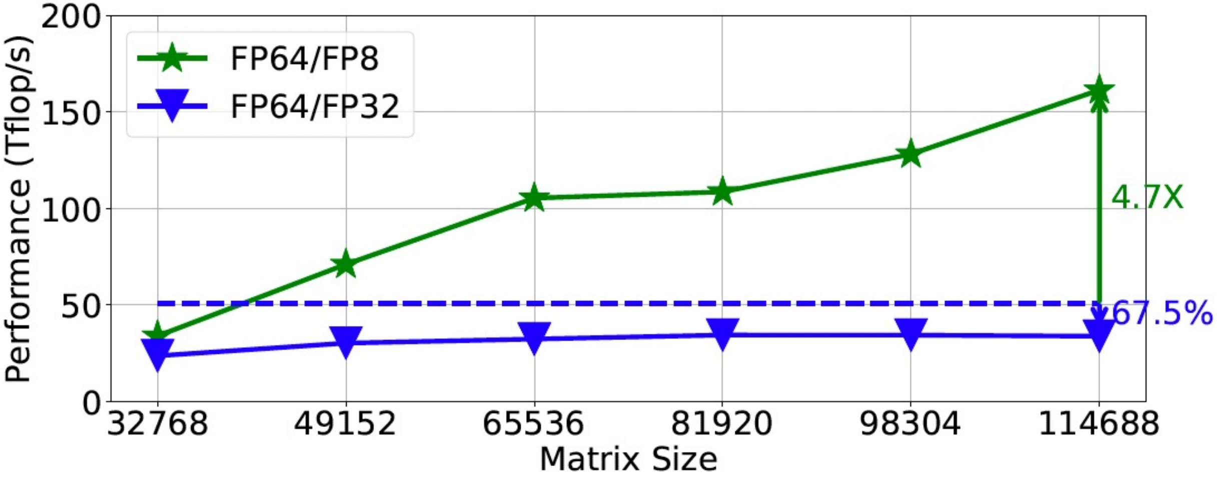
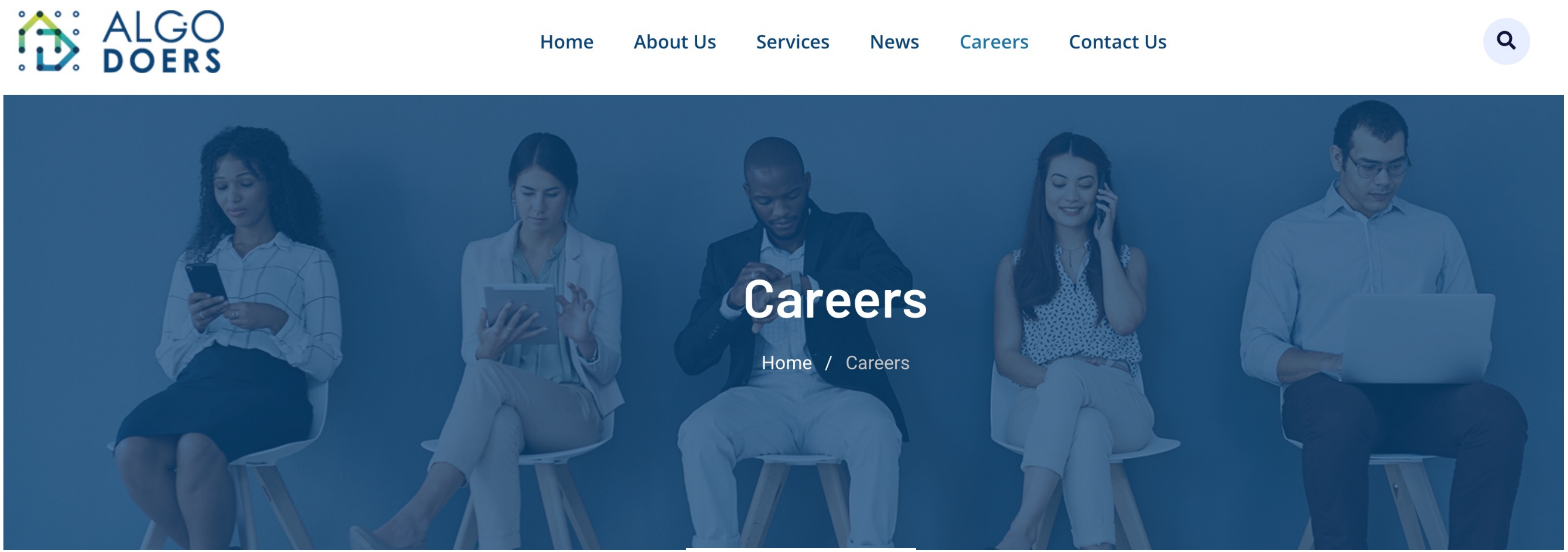


Fig. 7: Performance of FP64/FP8 and FP64/FP32 on H100 PCIe.

We hope to have access to NVIDIA EOS System

We are recruiting! Check it out @ www.algodoers.com



ORACLE THALES Shell TotalEnergies أرامكو السعودية saudi aramco VIRIDIEN

Hewlett Packard Enterprise NVIDIA intel AMD FUJITSU PASQAL EVIDEN

Thanks,

QUESTIONS?