Mixed-Precision Tomographic Reconstructor Computations on Hardware Accelerators

Mixed-Precision Tomographic Reconstructor Computations on Hardware Accelerators

Nicolas Doucet*†, Hatem Ltaief*, Damien Gratadour†, and David Keyes*

*Extreme Computing Research Center King Abdullah University of Science and Technology, Thuwal, Jeddah 23955
Email: Hatem.Ltaief, David.Keyes@kaust.edu.sa

†Paris Observatory - LESIA, Paris, France
Email: Nicolas.Doucet, Damien.Gratadour@obspm.fr

***Abstract*—The computation of tomographic reconstructors (ToR) is at the core of a simulation framework to design the next generation of adaptive optics (AO) systems to be installed on future Extremely Large Telescopes (ELT). In fact, it is also a critical component for their operation on sky. The goals of these instruments range from the detection of the light from the most distant galaxies to the analysis of the composition of exoplanets terrestrial atmospheres. Based on advanced AO techniques, the instrument MOSAIC relies on a computational framework to filter out the Earth atmospheric turbulence and eventually enhance the images quality out of ground-based telescopes. The ToR calculation is a compute-bound operation based on the Cholesky factorization. Due to its cubical algorithmic complexity, the ToR may represent a major bottleneck for the E-ELT when scaling up the large number of wavefront sensors used in the baseline MOSAIC design. To mitigate this increasing dimensionality overhead, this paper presents the implementation of a novel mixed-precision Cholesky-based dense matrix solver on hardware accelerators. The new algorithm takes into account the data-sparse structure of the covariance matrix operator and uses the tensor cores of NVIDIA V100 GPUs to leverage performance at an unprecedented scale. To our knowledge, this is the first computational astronomy application that exploits V100's tensor cores outside of the traditional arena of artificial intelligence. Experimental results demonstrate the accuracy robustness and the high performance of the mixed-precision ToR on synthetic datasets, which paves the way for future instrument deployments on the E-ELT.**

***Index Terms*—European Extremely Large Telescope, Tomographic Reconstructor, Mixed-Precision Algorithms, High Performance Computing, GPUs and Tensor Cores;**

I. Introduction

Looking back at four decades of microprocessor trend [1], the scientific community has come to the following striking observation: single-thread performance increase is over and so is perhaps the *free lunch*. Indeed, the semiconductor technology has reached physical limits due to power dissipation challenges, which resulted into the end of Dennard scaling. This has created a power wall with a plateau in processor clock frequency expansion during the last decade. To overcome these challenges, manycore architectures have come to the rescue, unleashing the computational power brought by high thread concurrency. This has been achieved at the expense of redesigning existing high performance software libraries and applications to take advantage of this unprecedented degree of parallelism [2]. Nevertheless, the quest toward exascale computing has still been slowed down and this has urged the community to further explore disruptive hardware and software solutions. This paper highlights both aforementioned possible solutions and their impacts in the context of a computational astronomy application.

Our application framework simulates and helps design the future Adaptive Optics (AO) instrumentation for Extremely Large Telescopes (ELT), such as the MOSAIC [3] instrument to be deployed on the European Extremely Large Telescope (E-ELT) [4], a.k.a., the biggest eye on Earth with a 39 meter mirror diameter. In particular, it is in charge of filtering out the atmospheric turbulence, captured by the wavefront sensors (WFS), on small islands of interest in a large Field of View (FoV). Using a multi-object adaptive optics (MOAO) approach [5], the core simulation of MOSAIC consists in computing a tomographic reconstructor (ToR) using covariance matrices generated from WFS measurements. Once the data is denoised after applying the ToR, the star light may be sent to MOSAIC with a significant improvement in resolution, from which the composition of their respective atmospheres may be identified. While initially developed in the context of MOAO, we have recently shown [6] that this framework can be generalized to other AO concepts such as Ground Layer AO (GLAO) or Multi-Conjugate (AO). MOSAIC actually provides a unique tool to enable tomographic wavefront reconstruction for a wide range of instruments for ELTs, including the E-ELT as well as the giant Magellanic Telescope [7] and the Thirty Meter Telescope [8].

The ToR executes on a symmetric dense covariance matrix, which contains the correlations between all the measurements of all the WFS. The ToR calculation then involves solving a large system of linear equations using the Cholesky factorization followed by a backward and forward substitution. The algorithmic complexity grows cubically as the number of measurements increases, making the ToR a computational challenge. Moreover, the time variation of the atmospheric turbulence imposes real-time constraints in order to precisely track the atmosphere natural evolution. These constraints necessitate integrating algorithmic innovations along with exploiting underlying hardware features to rise to the aforemen-

tioned challenge. We present herein a new high performance implementation of the ToR workflow to effectively decrease the time to solution.

The software solution comes from the discovery of the data-sparse structure of the covariance matrix, which may be typical for general covariance matrices. In other words, the matrix carries information which may not be relevant toward the final accuracy assessment, for instance, when considering the weak correlations between remote WFS. Therefore, one of the possible software solutions is to operate on these data-sparse off-diagonal data tiles with a lower precision arithmetic compared to other off-diagonal data tiles, which may carry more critical information. The resulting mixed-precision ToR algorithm (i.e., 32-bit and 16-bit floating-point arithmetic) should still ensure numerical accuracy and robustness above a certain application threshold to eventually meet the high quality of the image obtained with the optical instrument. The hardware solution consists in mapping this mixed-precision ToR algorithm into NVIDIA GPU V100 tensor cores, which are capable of performing half precision arithmetic up to eight times faster than single precision arithmetic. This interesting convergence of hardware and software solutions ultimately raises the achieved performance to an unprecedented scale. It further enables the MOAO application framework to render real-time computations possible, as the number of WFS increases for next generations of ground-based telescopes. Fine-grained computations, based on task-based programming model, are at the core of the framework workload. It provides the flexibility to orchestrate the mixed-precision algorithm at a tile level, while potentially exposing asynchronous executions to a dynamic runtime system of choice, e.g., StarPU [9].

Experimental results demonstrate the accuracy robustness of the ToR simulation framework on synthetic datasets, representative of E-ELT dimensions. We also report performance scalability using various hardware accelerator generations, including the latest NVIDIA V100 GPUs optimized for half-precision arithmetics. To our knowledge, this is perhaps the first astronomy application and one of the few scientific applications, which exploits NVIDIA V100 tensor cores outside of the traditional arena of artificial intelligence.

The remainder of the paper is as follows. Section II describes related work. Section III presents the main contributions of the paper. Section IV provides detailed information on the tomographic AO instruments to be deployed on the E-ELT. The ToR computational phase of the MOSAIC simulation is explained in Section V. Section VI introduces the novel fine-grained mixed-precision ToR algorithm, while implementation details are highlighted in Section VII. Numerical assessment and performance results are reported in Section VIII on synthetic datasets for the E-ELT using various hardware accelerator generations. Finally, Section IX summarizes the paper and presents future work.

II. Related Work

Half precision floating-point arithmetic has been widely used for Artificial Intelligence (AI) applications (e.g., image processing/segmentation, pattern recognition, etc.) within Deep Learning (DL) frameworks, including NVIDIA cuDNN [10], TensorFlow [11], Caffe [12], Theano [13], and PyTorch [14]. These frameworks translate most of the DL computational workloads to half precision general matrix-matrix (GEMM) multiplication function calls, which may be leveraged by tensor cores from NVIDIA V100 GPUs (i.e., 16-bit) or Tensor Processing Units (TPUs) from Google's custom-developed hardware accelerators (i.e., bfloat16). AI may typically be considered as the domain of predilection for half precision computations. However the convergence between Big Data and HPC [15] has pressured scientists to look for opportunities to leverage the performance of their applications by using half precision arithmetic. This paradigm shift may challenge default convention of computing at higher precisions.

Indeed, there are recent works toward democratizing half precision arithmetics for scientific computing, which allows the crossing for the first time of the symbolic exaflop barrier. For instance, the use of 16-bit has been studied for genome-wide association study in genetics [16], where the majority of the computations has been casted to 16-bit GEMM on NVIDIA V100 tensor cores. The same NVIDIA tensor cores have also been applied to climate applications for accelerating DL workloads [17] achieving significant performance speedups.

In fact, mixed-precision algorithms are not new and have been explored in the past. For instance, mixed-precision iterative refinement approaches have been studied for solving dense linear system of equations [18] using single and double precision arithmetics and recently in three precisions, adding half precision to the mix [19]. The theoretical speedup factor is up to two only, since floating-point units (FPUs) operate for the traditional two-precision iterative refinement on 32-bit operands. These algorithms represent possible software solutions at exascale, thanks to the additional hardware support for lower precision arithmetics. Indeed, a new mixed precision iterative refinement approach [20] has demonstrated significant performance improvement (speedup factor up to four) when using for each computational stage (a typical *coarse granularity* approach) different precision arithmetics , i.e., 16-bit, 32-bit, and 64-bit on NVIDIA V100 GPUs.

In this paper, we dive deeply into the design of a mixed-precision algorithm for solving a dense linear system of equations using task-based programming model, without the need of iterative refinement. Based on tile algorithms [21], we employ fine-grained computations to exploit the inherent covariance matrices' sparse structure at a tile level. This results in the development of a new Cholesky-based dense linear solver, customized herein for the computational astronomy application. This may actually create new opportunities for a wide range of scientific applications. For instance, a similar customized approach has already been successfully applied in the context of a climate and weather prediction application [22], though no hardware support for lower precision has been demonstrated.

III. Research Contributions

The main objective of the paper consists in trading-off arithmetic precision for performance by leveraging hardware and software features in the context of the simulation for advanced AO concepts such as multi-object of multi-conjugate AO to equip the future generation of giant ground-based astronomical telescopes. The crux of the paper consists of three contributions. We take advantage of the task-based programming model and employ a novel mixed-precision *fine-grained* Cholesky-based factorization and solve. The dynamic runtime system StarPU orchestrates the scheduling and the asynchronous executions of various computational tasks onto the underlying heterogeneous hardware resources, while monitor their data dependencies. We assess the numerical robustness and the performance impact of the new mixed-precision algorithm using synthetic datasets, which are proxies for the future deployment of the E-ELT's MOSAIC instrument. To our knowledge, this is perhaps the first astronomy application and one of the few scientific applications, which leverages both software and hardware solutions for mixed-precision support outside of traditional AI workloads.

IV. Understanding the Origin of the Universe with MOSAIC

For ground-based telescopes, the atmospheric turbulence is a limiting factor since it distorts the remote star light wavefront when it reaches the Earth. The concept of Adaptive Optics (AO) approach appeared in 1953 [23] to address this limitation, even though the enabling technologies were not mature enough. It was eventually further developed with the advent of large telescopes in the early 1990 [24]. This technique relies on a Deformable Mirror (DM) to effectively compensate for the distortions introduced by atmospheric turbulence.

As illustrated in Fig. 1, a typical AO loop incorporates one (or more) wavefront sensors (WFS) to measure the wavefront deformations, a Real-Time Controller (RTC) that computes the necessary compensation conveyed through a command vector, and a deformable mirror (DM) the surface of which distorts as a result of applying the command vector. This loop permits then to flatten the incoming wavefront accordingly. The DM's shape must constantly follow the evolution of the atmospheric turbulence, which requires the RTC module to provide the DM with commands at a high pace (i.e., the frequency of the AO loop is of the order of $1kHz$).

The new generation of instruments needs to support the capability of the future ELTs, which have unprecedented sizes, allowing to collect more photons and thus, observe the most distant galaxies or faintest exoplanets, unreachable up to now. For instance, the diameter of the European ELT (39m) is almost five times as large as the largest currently deployed ground-based telescopes. The complexity of the new AO instruments increases as the square of the diameter growth, i.e., around 25 times. MOSAIC [3], one of these new instruments, aims at studying the structure of the early universe (around 10 billion years ago) in order to understand its formation. To collect detailed information sample large enough to provide

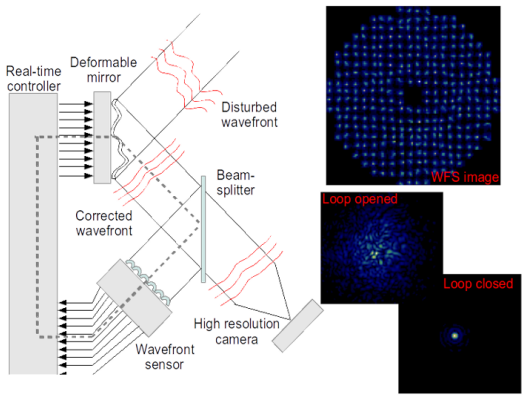

Fig. 1. The adaptive optic loop is composed by the deformable mirror (DM), the wavefront sensor (WFS), and the real-time controller (RTC).

astrophysicists with sufficient statistics for this study in reasonable time, multiple objects must be observed simultaneously. The AO loop described in Fig. 1 is modified for this purpose into a Multi-Object AO (MOAO) loop with more WFS probing a wide field of view and multiple DM to compensate for the turbulence on small patches in the different directions of interest. The RTC relies on a Tomographic Reconstructor (ToR), a change of basis matrix, to convert the (off axis) WFS measurements into theoretical on axis measurements used to determine the DM's shape. In the case of MOAO, one ToR is computed for each observation channel. Furthermore, the ToR has to follow the evolution of the atmosphere structure that varies over night and needs to be recomputed regularly (at the minute scale).

The ToR is obtained by solving a dense linear system, involving covariance matrices generated from the WFS measurements that can reach dimensions up to 100k in the case of the E-ELT. Beyond MOAO and the MOSAIC instrument, the same approach can be generalized to support other AO concepts relying on multiple wavefront sensing directions and turbulence tomography [6]. For instance, one can cite the MAORY MCAO instrument on the E-ELT [25] but also the GMTIFS and associated Laser Tomography AO (LTAO) module on the GMT [26] or the future MCAO module for the TMT [27]. Hence, this work is meant to have a critical impact on both the design phases of most ELTs AO instrumentation programs as well as their future operations on-sky. MOSAIC being one of the most challenging tomographic AO concept contemplated so far, we use it to benchmark our framework and prove its ability to deliver both the required accuracy and performance in terms of time-to-solution.

V. State-Of-The-Art ToR Computations

The computation of the Tomographic Reconstructor (ToR) is at the core of the operations of all the tomographic AO instruments concepts contemplated on ELTs. And even though it is not part of the hard real-time controller (RTC), which is in charge of pushing/pulling the DMs' actuators to compensate for the atmospheric turbulence, the ToR computation is still

subject to a soft real-time constraint and must be updated regularly to take into account the evolution of the atmosphere's structure. Previous works have demonstrated high performance implementations of the ToR, which consists in computing the Cholesky factorization of the dense symmetric covariance matrix followed by a backward and forward substitution. The most time-consuming kernel during the factorization and the solve phase is the general matrix-matrix multiplication (GEMM), which makes the compute-bound algorithm run close to system's sustained peak performance. Performance results have been reported on shared-memory systems equipped with hardware accelerators [5], [28], [29] as well as distributed-memory systems [30] using single precision floating-point arithmetic. In particular, tile algorithms [21] have been instrumental to get high performance on GPU-based systems [29]. The main idea is to divide these dense matrix operations into fine-grained computations using task-based programming model. The overall algorithm can then be expressed as a directed acyclic graph, where nodes correspond to tasks and edges represent data dependencies. The StarPU dynamic runtime system [9] coordinates the scheduling of these various computational tasks, both on CPUs and GPUs, by ensuring data dependencies are not violated. Not only this fine-grained computation enables asynchronous task execution, but also it permits to overlap data movement across the slow PCIe bus between the host and the device with useful computations. All in all, tile algorithms associated with a dynamic runtime system mitigate the overhead of bulk synchronous programming model, by weakening the artifactual synchronization points, while maximizing the occupancy on the underlying hardware resources (CPUs or GPUs).

However, all these approaches do not consider exploiting the numerical property of the matrix operator nor introducing mixed-precisions techniques. It turns out that the dense covariance matrix, which may be of size as large as 100k, has a data-sparse structure, due to weak interactions between some of the measurements taken by the WFS, as detailed in the next section.

VI. Leveraging Mixed-Precision Techniques for the ToR Computations

With the advent of hardware support for low precision floating-point arithmetics (e.g., Google TPU chip and NVIDIA GPU with tensor cores), the covariance matrix structure has been placed under scrutiny to identify opportunities for mixed precision computations. The measurements used to generate the matrix operator comes from the analog WFS cameras, which have to be converted to 32-bit in order to perform floating-point computations on the covariance matrix. Fig. 2(left) pictures *an X-ray* of a covariance matrix by performing an eigenvalue decomposition using 13248 measurements and eight WFS devices, which reveals an exponential decay. We divide the matrix into tiles of size 138 and run an eigenvalue decomposition on each individual tile. Fig. 2(right) reports the eigenvalue distribution on various matrix tiles by locations, starting from the main diagonal all the way up the top right corner of the symmetric matrix. This figure shows that tiles located around the diagonal carry useful information with eigenvalues reaching highest magnitude. This is expected, since measurements taken by WFS cameras physically located next to each other exhibit highest correlations. By contrast, as we move away from the main matrix diagonal tile, the magnitudes of the off-diagonal tiles' eigenvalues decrease. This corresponds to weak interactions between measurements taken by remote WFS cameras. These experiments provide insight into the matrix tile structure and exposes the overall data sparsity of the covariance matrix. Therefore, they pave the way for support of low precision computations. Furthermore, the task-based programming model gives the flexibility to decide at a fine-grained level, i.e., at the tile level, which precision arithmetic should be used before operating on it. This fine-grained precision control may provide a better impact in terms of performance, while maintaining the required instrument accuracy. The traditional coarse-grained approach with iterative refinement, as proposed in [20], not be numerically stable for the herein application, given the high condition number of the covariance matrix. Such matrix tile structure is

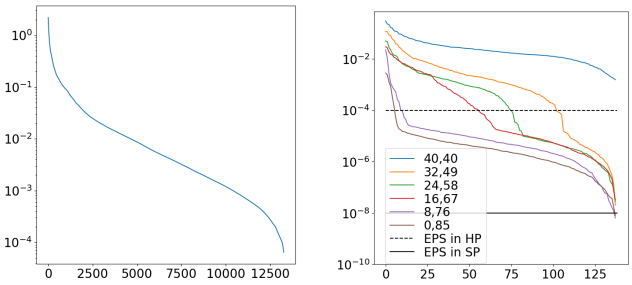

Fig. 2. Eigenvalue distribution of the global covariance matrix (left) and Eigenvalue distribution of individual tiles with coordinates (i, j), with i the row index and j the column index (right)

representative of the covariance matrices studied in adaptive optics for astronomy applications on a given ground-based telescope. Therefore, the eigenvalue decomposition may not need to be calculated every time a covariance matrix is generated to determine its structure, since the correlations between measurements of fixed WFS cameras may not drastically change. Instead, a priori knowledge on the covariance matrix structure may be sufficient to cherry-pick tiles candidate for low precision computations.

For enabling half-precision arithmetics in our application, the idea is to replace our original 32-bit tile GEMM CUDA function (i.e., 4^{th} variant) by one of the three 32/16-bit mixed-precision GEMM CUDA function, namely `cublasGemmEx`, as detailed in Table I. These three GEMM variants operate directly on NVIDIA V100 tensor cores. They provide incremental levels of support for mixed precision arithmetics. The 1^{st} variant supports solely 16-bit `muladd` operations. The 2^{nd} variant performs the `add` operation in 32-bit with 16-bit input/output operands. Besides doing the `add` operation in 32-bit, the 3^{rd} variant returns the output operand C in 32-bit precision arithmetic. The four variants are ranked from fastest

to slowest in terms of performance.

TABLE I
32/16-BIT MIXED PRECISION CUBLASGEMMEX SUPPORTED FUNCTION.

GEMM variants	A/B type	C type	Compute type	Alpha / Beta
HP3	16-bit	16-bit	16-bit	16-bit
HP2	16-bit	16-bit	32-bit	32-bit
HP1	16-bit	32-bit	32-bit	32-bit
SP	32-bit	32-bit	32-bit	32-bit

The first column will be used to specify the function's precision in the following.

These variants allow to precisely study the numerical resilience of the MOSAIC simulation framework, by trading off precision arithmetics for performance and identifying the point of no return. In our particular application, this point of no return is when the symmetric covariance matrix looses its positive definiteness, from which the Cholesky factorization fails and the solver cannot proceed anymore.

VII. IMPLEMENTATION DETAILS

To compute the ToR, we rely on the Chameleon library [31] with StarPU [9] as building blocks to implement the novel mixed-precision tile Cholesky factorization and its solver on accelerator-based systems.

a) StarPU Dynamic Runtime System: StarPU asynchronously schedules the various tasks in parallel, according to their data dependencies, onto available CPU and GPU hardware resources. In case the GPU runs out of memory, StarPU employs an out-of-core strategy and uses the CPU memory to compensate for the lack of GPU memory at runtime. StarPU maintains the data coherency between the CPU and GPU by using a heuristic similar to the cache coherency protocol, while aggressively prefetching data to ensure high occupancy on the device.

b) Mixed-Precision Cholesky-based Solver: Since Chameleon does not provide a `spotrf` GPU-resident task implementation, we integrate the MAGMA [32] GPU kernel implementation. For the remaining Level-3 BLAS kernels, i.e., `strsm-ssyrk-sgemm`, we rely on the `cuBLAS` library. We run on the NVIDIA V100's tensor cores for the half-precision operation, therefore only the matrix-matrix multiplication (GEMM) can be performed in half precision (`hgemm`). The cuBLAS `hgemm` implementation for the Tensor Cores (i.e., `cublasGemmEx`) allows various combination of the precision of the input/output data tile and tensor core operation (see Table I). The direct approach is to create `hgemm` tasks for half precision GEMM and replace the corresponding operation in the ToR computation. This approach thus requires an additional GPU-resident task to convert the 32-bit tiles to 16-bit, as soon as the tile is available for the `hgemm` operation. Then, depending on the precision variant used, we convert the tile back to 32-bit. We cannot replace all `sgemm` by `hgemm` calls since the covariance matrix may loose its positive definiteness. The Cholesky factorization will then fail and the overall ToR computation cannot further proceed. To prevent such situations from happening, we add a mechanism to choose the precision of each GEMM with the help of a functor. It is a structure for which the user defines the operator () as an heuristic, taking as input argument the position of the tile in the matrix and returning the precision variant of the GEMM operation to be used for this tile.

c) Heuristic for Mixed-Precision Computations: The decision of swapping with a lower precision computational kernel can be made at runtime via a heuristic formulae. The actual mechanism is non-intrusive: users can define its own heuristic through the functor to feed the mixed-precision Cholesky factorization and the corresponding solver functions with information to operate on the proper tile coordinates. The implementation of the ToR itself does not need to be modified. As a result, users can instantiate functors and define a suitable heuristic to choose the precision of the GEMM operation on a given tile, without being intrusive into the underlying linear algebra routine. This implementation can, therefore, be used by other applications [22] and still exploit the matrix specific structure at hand, simply by customizing the heuristic. A similar mechanism can be enforced on the solve stage, i.e., the backward and forward substitution, but in our case, all GEMMs of the substitutions can be computed with HP1, without hindering the solution accuracy. In this paper, the heuristic we used is fairly simple: we choose the precision of the GEMM based on the distance of the output tile to the diagonal, using different thresholds to address the different variants of the half precision. Basically, the closer a tile is to the diagonal, the more accurate the operation is. Typically, in our application, the first tiles around the diagonal must be treated in 32-bit, while the precision arithmetic for the far off-diagonal tiles can be relaxed. Fig. 3 illustrates this heuristic. The current heuristic is general enough, and thus, amenable to capture more complex matrix structures with parametrized regions' shape.

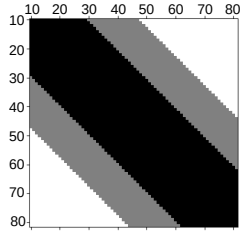

Fig. 3. Heuristic based on the studied data-sparse matrix structure. The color depends on the GEMM's accuracy, the tiles operated in SP, HP1 and HP2 (see Table I) are respectively in black, grey and white.

d) Pseudo-code: The ToR is obtained by solving $X \cdot A = B$, where X is the $m \times n$ unknown ToR matrix, A is an $n \times n$ symmetric definite positive matrix and B is the $(m \times n)$ the right hand side, with n and m are the number of WFS measurements and the number of measurements of the true sensors, respectively. The matrices are split into MT and NT number of tiles, using the tile size nb. Algorithm 1 introduces the mixed-precision cholesky factorization, which decomposes $A = L \cdot L^T$. Algorithm 2 shows the mixed-precision backward substitution $y \cdot L^T = B$. The forward substitution can be easily derived from Algorithm 2. We implement a GPU-resident kernel $SP2HP$ to perform the precision conversion from 32-

bit to 16-bit.

Algorithm 1 Mixed-Precision Cholesky factorization.

```
1: for k=0..NT-1 do
2:     A[k][k] ← spotrf(A[k][k])
3:     for m=k+1..NT-1 do
4:         A[m][k] ← strsm(A[k][k], A[m][k])
5:         H[m][k] ← SP2HP(A[m][k])
6:     end for
7:     for n=k+1..NT-1 do
8:         A[n][n] ← ssyrk(A[n][k], A[n][n])
9:         for m=n+1..NT-1 do
10:            precision ← heuristic(m, n)
11:            if precision == float then
12:                A[m][n] ← sgemm(A[m][k], A[n][k], A[m][n])
13:            else if precision == half then
14:                if variant == 1 or 2 then
15:                    H[m][n] ← SP2HP(A[m][n])
16:                    H[m][n] ← hgemm(H[m][k], H[n][k], H[m][n])
17:                    A[m][n] ← SP2HP(H[m][n])
18:                else
19:                    A[m][n] ← hgemm(H[m][k], H[n][k], A[m][n])
20:                end if
21:            end if
22:        end for
23:    end for
24: end for
```

Algorithm 2 Mixed-precision backward substitution.

```
1: for k=0..NT-1 do
2:     for m=0..MT-1 do
3:         B[m][k] ← trsm(A[k][k], B[m][k]
4:         Hb[m][k] ← SP2HP(B[m][k])
5:         for n=k..NT-1 do
6:             Ha[n][k] ← SP2HP(A[n][k])
7:             B[m][n] ← hgemm(Hb[m][k], Ha[n][k], B[m][n])
8:         end for
9:     end for
10: end for
```

VIII. Experimental Results

a) Environment Settings: We test two shared-memory systems, each with two sockets 20-core Intel(R) Xeon(R) Broadwell CPU E5-2698 v4 @ 2.20GHz, that are connected to four GPU through PCIe (10GB/s). They differ on the GPU they host. The first one has four Nvidia P100 interconnected to each other with a 20GB/s NVLink, as depicted in Fig. 4. The second one has four Nvidia V100, 3 GPU pairs are connected with a double NVLink of 25GB/s (for a total of 50GB/s), and the remaining pairs are connected with a single NVLink, as seen in Fig. 4. A single P100 GPU has a theoretical peak

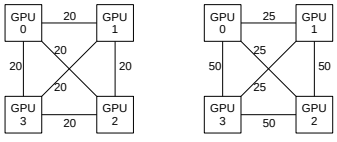

Fig. 4. Peer-to-Peer bandwidth for the P100 (left) and V100 (right) GPU-based systems.

performance of 10.8 TFlop/s in single precision, whereas a V100 GPU performance goes as high as 15 TFlop/s. We refer to these systems as P100 and V100. The half precision's theoretical peak performance of the V100-based system (since the P100 is not equipped with Tensor Cores) is 500 TFlop/s (125 TFlop/s per GPU). In the case of our mixed-precision ToR algorithm, the sustained peak performance depends on the ratio of GEMM performed in 32-bit versus 16-bit, as well as the volume of data motion among the GPU pairs.

b) Numerical Accuracy: We test the numerical robustness of the mixed-precision ToR. We compare the AO performance using the Strehl Ratio (SR) obtained with the end-to-end simulation tool COMPASS [33] on long exposure images. The SR corresponds to the ratio of the maximum value of the image of a point source over its theoretical maximum (1 being the best achievable image). In other words, the SR provides the quality of the image obtained with the optical instrument. We use the SR of the SP as a reference to compare against the mixed precision approaches. In Fig. 5(left) and 5(right), we test two different AO systems: an eight and a forty meter diameter telescopes with a total of $17k$ and $50k$ measurements, respectively. In addition to the SR, we specify for each bar the proportion of each GEMM variant used (i.e., SP, HP1, HP2, and HP3) among all the tiles involved in a GEMM operation, there is a total of 153 and 1326 of these tiles for each case respectively. Due to excessive low precision computations, the covariance matrix may loose its positive-definiteness and the Cholesky factorization may fail. Then, the ToR cannot be computed and the SR cannot be provided. Fig. 5 shows

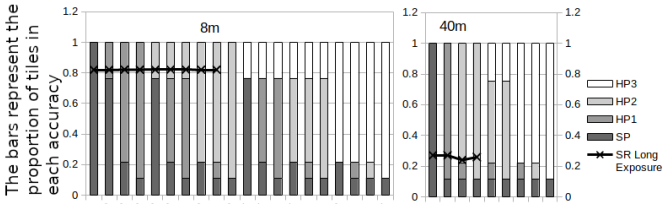

Fig. 5. Proportion of the different GEMM precision and associated ToR accuracy on a eight meter telescope with $17k$ measurements (left) and an forty meter telescope with $50k$ measurements (right).

that if the Cholesky factorization succeeds, the ToR has an AO performance equivalent to a single precision approach. Moreover, only a small amount of SP tiles are required when mixed with HP1 or HP2 to achieve a reasonable SR. However, the configurations using HP3 constantly fails getting a proper AO performance.

c) Performance Results: We report in Fig. 6 the performance results in Tflop/s obtained on the two systems described in VIII-0a in SP and mixed precision for the V100. In the case of the mixed precision, the chosen heuristic uses SP GEMM for the tiles that are near the diagonal and only HP1 cublasGemmEx for the others. The threshold that determines the operation accuracy is set to maximize the

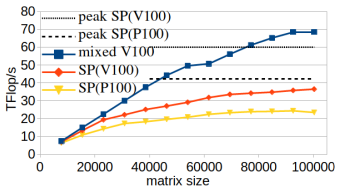

Fig. 6. ToR performance with four GPUs.

number of tasks performed with the tensor cores while producing a valid ToR, as highlighted in Section VIII-0b.

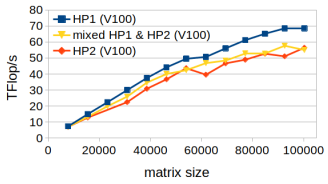

Fig. 7. ToR performance for various mixed precision cases with four GPUs

This heuristic achieves in fact the best performance among the tested heuristics, as shown in Fig. 7.

On both systems, the ToR computation achieves around 60% of the theoretical peak performance in single precision. However, the mixed-precision approach only attains a third of the theoretical peak performance. The sustained performance of the Cholesky factorization only was observed to be almost identical to the one of the ToR computation The performance obtained when executing only the factorization reflects this behavior, suggesting the performance issue comes from the factorization itself and more precisely, the GEMM operation. This assumption is confirmed as we remove from the Cholesky factorization workflow all operation that is not a GEMM.

The performance obtained for the GEMM execution of the factorization, as presented in Fig. 8, shows that the SP approach achieves respectively 70% and 80% of the peak performance for the P100 and V100, respectively. The ratio gets down to 42% for the mixed-precision and 25% for the full half precision approach. However, the performance loss from the Cholesky's GEMM to the full ToR computation is better for the mixed-precision case, showing that the StarPU runtime system schedules the additional tasks efficiently.

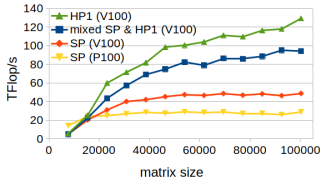

Fig. 8. GEMMs-only of the Cholesky factorization with four GPUs

The scalability graph in Fig. 9(left) shows that the performance almost doubles while moving from two to four GPUs, but the performance obtained for a single GPU stays low. This low performance can be explain by the fact that a single GPU does not have enough memory to store the full matrix, forcing the out-of-core feature of StarPU to be activated in order to use the host memory additionally. This is visible when one observes the data transfers from Fig. 9(right): as GPUs are added, the portion of transfer between the host and the GPU decreases significantly. For this mixed precision implementation, tiles must be store both in SP and HP increasing the memory usage, therefore increasing the host to device communication. Last but not least, the communication volume between host to device in single precision is higher with P100-based than V100-based system. This is related to a StarPU runtime decision which may favor time-to-time the offloading to P100 host memory system given the higher peer-to-peer bandwidth achieved on the V100 system, as shown in Fig. 4.

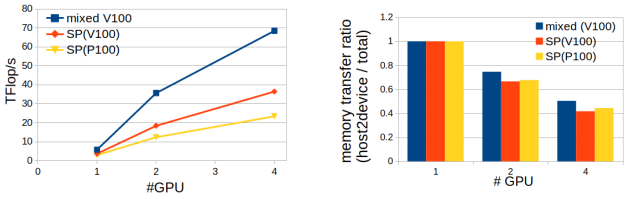

Fig. 9. Scalability for 1 to 4 GPUs for a 100k matrix (left), Memory transfer ratio: out-of-core (right)

IX. Conclusions and Future Work

This paper explores the use of NVIDIA V100 tensor cores for mixed-precision GEMM to speed up the ToR computation achieving almost 70 TFlop/s. This brings the overall ToR computation time of the mixed-precision ToR at the E-ELT scale below 8s as opposed to 14s for 32-bit ToR. This comes though at the price of additional data movement compared to the naive 32-bit ToR, which prevents reaching even higher performance. We plan to improve the runtime decisions within StarPU to better overlap data traffic with useful computations. This work represents a major milestone in assessing such challenging instruments and has a critical impact on both the design phases of most ELTs AO instrumentation programs as well as their future on-sky operations, beyond the MOSAIC instrument studied herein.

References

[1] K. Rupp, "42 Years of Microprocessor Trend Data," https://www.karlrupp.net/2018/02/42-years-of-microprocessor-trend-data/, Feb. 2018.

[2] H. Meuer, E. Strohmaier, J. Dongarra, and H. Simon", "The Top500 List," Jun. 2019, http://www.top500.org.

[3] F. Hammer, B. Barbuy, J. G. Cuby, L. Kaper, S. Morris, C. J. Evans, P. Jagourel, G. Dalton, P. Rees, M. Puech, M. Rodrigues, D. Pearson, and K. Disseau, "MOSAIC at the E-ELT: A multi-object spectrograph for astrophysics, IGM and cosmology," in *Ground-based and Airborne Instrumentation for Astronomy V*, ser. Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 9147, Aug. 2014, p. 914727.

[4] A. McPherson, J. Spyromilio, M. Kissler-Patig, S. Ramsay, E. Brunetto, P. Dierickx, and M. Cassali, "E-ELT update of project and effect of change to 39m design," in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, ser. Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 8444, Sep. 2012.

[5] É. Gendron, A. Charara, A. Abdelfattah, D. Gratadour, D. Keyes, H. Ltaief, C. Morel, F. Vidal, A. Sevin, and G. Rousset, "A novel fast and accurate pseudo-analytical simulation approach for MOAO," in *Adaptive Optics Systems IV*, ser. Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 9148, Aug. 2014, p. 91486L.

[6] N. Doucet, R. Kriemann, E. Gendron, D. Gratadour, H. Ltaief, and D. Keyes, "Scalable soft real-time supervisor for tomographic AO," in *Adaptive Optics Systems VI*, ser. Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 10703, Jul 2018, p. 107034L.

[7] A. H. Bouchez, G. Z. Angeli, D. S. Ashby, R. Bernier, R. Conan, B. A. McLeod, F. Quirós-Pacheco, and M. A. van Dam, "An overview and status of GMT active and adaptive optics," in *Adaptive Optics Systems VI*, ser. Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 10703, Jul 2018, p. 107030W.

[8] C. Boyer, "Adaptive optics program at TMT," in *Adaptive Optics Systems VI*, ser. Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 10703, Jul 2018, p. 107030Y.

[9] C. Augonnet, S. Thibault, R. Namyst, and P. Wacrenier, "StarPU: A unified platform for task scheduling on heterogeneous multicore architectures," *Concurrency Computat. Pract. Exper.*, vol. 23, pp. 187–198, 2011.

[10] S. Chetlur, C. Woolley, P. Vandermersch, J. Cohen, J. Tran, B. Catanzaro, and E. Shelhamer, "cudnn: Efficient primitives for deep learning." *CoRR*, vol. abs/1410.0759, 2014. [Online]. Available: http://arxiv.org/abs/1410.0759

[11] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[12] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional Architecture for Fast Feature Embedding," *arXiv preprint arXiv:1408.5093*, 2014.

[13] Theano Development Team, "Theano: A Python Framework for Fast Computation of Mathematical Expressions," *arXiv e-prints*, vol. abs/1605.02688, May 2016. [Online]. Available: http://arxiv.org/abs/1605.02688

[14] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch," in *NIPS-W*, 2017.

[15] M. Asch, T. Moore, R. Badia, M. Beck, P. Beckman, T. Bidot, F. Bodin, F. Cappello, A. Choudhary, B. de Supinski, E. Deelman, J. Dongarra, A. Dubey, G. Fox, H. Fu, S. Girona, W. Gropp, M. Heroux, Y. Ishikawa, K. Keahey, D. Keyes, W. Kramer, J.-F. Lavignon, Y. Lu, S. Matsuoka, B. Mohr, D. Reed, S. Requena, J. Saltz, T. Schulthess, R. Stevens, M. Swany, A. Szalay, W. Tang, G. Varoquaux, J.-P. Vilotte, R. Wisniewski, Z. Xu, and I. Zacharov, "Big data and extreme-scale computing: Pathways to Convergence-Toward a shaping strategy for a future software and data ecosystem for scientific inquiry," *The International Journal of High Performance Computing Applications*, vol. 32, no. 4, pp. 435–479, 2018. [Online]. Available: https://doi.org/10.1177/1094342018778123

[16] W. Joubert, D. Weighill, D. Kainer, S. Climer, A. Justice, K. Fagnan, and D. Jacobson, "Attacking the opioid epidemic: Determining the epistatic and pleiotropic genetic architectures for chronic pain and opioid addiction," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage, and Analysis*, ser. SC '18. Piscataway, NJ, USA: IEEE Press, 2018, pp. 57:1–57:14. [Online]. Available: http://dl.acm.org/citation.cfm?id=3291656.3291732

[17] T. Kurth, S. Treichler, J. Romero, M. Mudigonda, N. Luehr, E. Phillips, A. Mahesh, M. Matheson, J. Deslippe, M. Fatica, Prabhat, and M. Houston, "Exascale deep learning for climate analytics," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage, and Analysis*, ser. SC '18. Piscataway, NJ, USA: IEEE Press, 2018, pp. 51:1–51:12. [Online]. Available: http://dl.acm.org/citation.cfm?id=3291656.3291724

[18] A. Buttari, J. Dongarra, J. Langou, J. Langou, P. Luszczek, and J. Kurzak, "Mixed Precision Iterative Refinement Techniques for the Solution of Dense Linear Systems," *The International Journal of High Performance Computing Applications*, vol. 21, no. 4, pp. 457–466, 2007. [Online]. Available: https://doi.org/10.1177/1094342007084026

[19] E. Carson and N. Higham, "Accelerating the solution of linear systems by iterative refinement in three precisions," *SIAM Journal on Scientific Computing*, vol. 40, no. 2, pp. 817–847.

[20] A. Haidar, S. Tomov, J. Dongarra, and N. J. Higham, "Harnessing GPU Tensor Cores for Fast FP16 Arithmetic to Speed Up Mixed-precision Iterative Refinement Solvers," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage, and Analysis*, ser. SC '18. Piscataway, NJ, USA: IEEE Press, 2018, pp. 47:1–47:11. [Online]. Available: http://dl.acm.org/citation.cfm?id=3291656.3291719

[21] E. Agullo, J. Demmel, J. Dongarra, B. Hadri, J. Kurzak, J. Langou, H. Ltaief, P. Luszczek, and S. Tomov, "Numerical linear algebra on emerging architectures: The PLASMA and MAGMA projects," *Journal of Physics: Conference Series*, vol. 180, 2009.

[22] S. Abdulah, H. Ltaief, Y. Sun, M. G. Genton, and D. E. Keyes, "Geostatistical Modeling and Prediction Using Mixed-Precision Tile Cholesky Factorization," in *Proceedings of the 26th IEEE International Conference on High Performance Computing, Data, and Analytics*. Hyderabad, India: IEEE Computer Society, 2019.

[23] H. W. Babcock, "The possibility of compensating astronomical seeing," *Publications of the Astronomical Society of the Pacific*, 1953.

[24] P. Y. Kern, P. J. Lena, P. Gigan, F. J. Rigaut, G. Rousset, J.-C. Fontanella, J.-P. Gaffard, C. Boyer, P. Jagourel, and F. Merkle, "Adaptive optics prototype system for infrared astronomy, i: system description," 1990. [Online]. Available: https://doi.org/10.1117/12.20411

[25] P. Ciliegi, E. Diolaiti, R. Abicca, G. Agapito, M. Aliverti, C. Arcidiacono, N. Auricchio, A. Balestra, A. Baruffolo, M. Bellazzini, M. Bonaglia, G. Bregoli, O. Brissaud, L. Busoni, A. Carlotti, E. Cascone, J.-J. Correia, F. Cortecchia, G. Cosentino, V. D'Orazi, M. Dall'Ora, V. De Caprio, A. De Rosa, A. Delboulbé, I. Di Antonio, G. Di Rico, M. Dolci, S. Esposito, D. Fantinel, P. Feautrier, G. Fiorentino, I. Foppiani, E. Giro, L. Gluck, P. Grani, D. Greggio, F. Hénault, L. Jocou, P. La Penna, S. Lafrasse, M. Lauria, E. Le Coarer, M. Le Louarn, M. Lombini, Y. Magnard, D. Magrin, E. Maiorano, F. Mannucci, E. Marchetti, D. Maurel, L. Michaud, E. Moraux, G. Morgante, T. Moulin, S. Oberti, G. Pariani, M. Patti, C. Plantet, L. Podio, A. Puglisi, P. Rabou, R. Ragazzoni, E. Redaelli, M. Riva, S. Rochat, F. Roussel, A. Roux, B. Salasnich, P. Saracco, L. Schreiber, M. Spavone, E. Stadler, M.-H. Sztefek, L. Terenzi, A. Valentini, N. Ventura, C. Vérinaud, and S. Zaggia, "MAORY for ELT: preliminary design overview," in *Adaptive Optics Systems VI*, ser. Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 10703, Jul 2018, p. 1070311.

[26] R. Sharp, G. Bloxham, R. Boz, D. Bundy, J. Davies, B. Espeland, B. Fordham, J. Hart, N. Herrald, J. Nielsen, A. Vaccarella, C. Vest, P. Young, and P. McGregor, "GMTIFS: The Giant Magellan Telescope integral fields spectrograph and imager," in *Ground-based and Airborne Instrumentation for Astronomy VI*, ser. Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 9908, Aug 2016, p. 99081Y.

[27] J. Crane, G. Herriot, D. Andersen, J. Atwood, P. Byrnes, A. Densmore, J. Dunn, J. Fitzsimmons, T. Hardy, B. Hoff, K. Jackson, D. Kerley, O. Lardière, M. Smith, J. Stocks, J.-P. Véran, C. Boyer, L. Wang, G. Trancho, and M. Trubey, "NFIRAOS adaptive optics for the Thirty Meter Telescope," in *Adaptive Optics Systems VI*, ser. Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 10703, Jul 2018, p. 107033V.

[28] A. Charara, H. Ltaief, D. Gratadour, D. E. Keyes, A. Sevin, A. Abdelfattah, E. Gendron, C. Morel, and F. Vidal, "Pipelining Computational Stages of the Tomographic Reconstructor for Multi-Object Adaptive Optics on a Multi-GPU System," in *SC'14*. IEEE, 2014, pp. 262–273. [Online]. Available: http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=7012142

[29] H. Ltaief, D. Gratadour, A. Charara, and E. Gendron, "Adaptive Optics Simulation for the World's Largest Telescope on Multicore Architectures with Multiple GPUs," in *Proceedings of the Platform for Advanced Scientific Computing Conference*, ser. PASC '16. New York, NY, USA: ACM, 2016, pp. 9:1–9:12. [Online]. Available: http://doi.acm.org/10.1145/2929908.2929920

[30] H. Ltaief, A. Charara, D. Gratadour, N. Doucet, B. Hadri, E. Gendron, S. Feki, and D. Keyes, "Real-Time Massively Distributed Multi-object Adaptive Optics Simulations for the European Extremely Large Telescope," in *2018 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, May 2018, pp. 75–84.

[31] "The Chameleon Project," http://project.inria.fr/, INRIA Bordeaux, 2018.

[32] MAGMA, "Matrix Algebra on GPU and Multicore Architectures. Innovative Computing Laboratory, University of Tennessee. Available at http://icl.cs.utk.edu/magma/," 2009.

[33] D. Gratadour, M. Puech, C. Vérinaud, P. Kestener, M. Gray, C. Petit, J. Brulé, Y. Clénet, F. Ferreira, E. Gendron, M. Lainé, A. Sevin, G. Rousset, F. Hammer, I. Jégouzo, M. Paillous, S. Taburet, Y. Yang, J.-L. Beuzit, A. Carlotti, M. Westphal, B. Epinat, M. Ferrari, T. Gautrais, J. C. Lambert, B. Neichel, and S. Rodionov, "COMPASS: an efficient, scalable and versatile numerical platform for the development of ELT AO systems," in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, ser. Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 9148, Aug. 2014, p. 6.