Green Flash: Exploiting future and emerging computing technologies for AO RTC at ELT scale

Gratadour D.[a], Andrighettoni M.[c], Angerer G.[c], Basden A.[b], Bernard J.[a], Biasi R.[c], Bitenc U.[b], Buey T.[a], Dembet R.[a], Deneux H.[d], Dipper N.[b], Doucet N.[a], Dufourc G.[d], Durand S.[a], Ferreira F.[a], Flaischer A.[d], Gendron E.[a], Geng D.[b], Jenkins D.[b], Lain M.[a], Leclere J.B.[d], Le Ruyet B.[a], Osborn J.[b], Patauner C.[c], Perret D.[a], Pescoller D.[c], Porta F.[c], Pretet D.[d], Reeves A.[b], Rouad C.[d], Sevin A.[a], Staykov L.[b], Townson M.[b], and Younger E.[b]

[a]LESIA - Observatoire de Paris/ Université Paris Diderot (France)
[b]CfAI, University of Durham (United Kingdom)
[c]Microgate (Italy)
[d]Accelize / PLDA (France)

ABSTRACT

Controlling sophisticated AO systems on the future ELTs is a challenge which is yet to be resolved. As the ELTs are adaptive telescopes, this is critical to all ELT instrumentation. Green Flash is an international EU funded joint industrial and academic project intended to study and exploit future and emerging computing technologies for ELT scale AO real-time control. This includes the hard real-time data pipeline, the soft real-time supervisor module as well as a real-time capable ELT-scale simulation to test and verify proposed solutions. To date, we have initiated in-depth studies on GPUs, MICs and FPGAs. Moreover, beyond computing capabilities, it is critical for these future AO RTC to address the data flow challenges both in terms of the large rate of streaming data from sensors and in terms of heterogeneous data streams in the system. A strong emphasis is thus made in the project around interconnect strategies using dedicated hardware, middleware and software. With this R&D program we aim at feeding the E-ELT AO systems preliminary design studies, led by the selected first-light instruments consortia, with technological validations supporting the designs of their RTC modules. The culmination of the project will be an on-sky demonstration of a chosen solution. However, all proposed solutions will include a detailed study of capabilities and scalability. Here we review the project goals and the results from the studies at the mid-point of this three year endeavor and we describe the downselection process that will lead to the design of a full featured prototype to be eventually implemented in the lab and tested with the real-time simulator.

Keywords: Adaptive Optics, Real-time computer, GPU, FPGA, High Performance Computing

1. INTRODUCTION

The main goal of Green Flash[1] is to design and build a prototype for a Real-Time Controller (RTC) targeting the European Extremely Large Telescope (E-ELT) Adaptive Optics (AO) instrumentation. This initiative responds to a critical challenge in the astronomical community. Scaling up the real-time control solutions of AO instruments in operation or currently planned for the 8-10m telescopes to the specifications of the AO modules at the core of the next generation of extremely large telescopes is not a viable option. New technological developments are needed to implement HPC facilities supporting this future extremely large scale scientific equipment, for which the construction has begun.

With Green Flash, we are proposing technical solutions, assessing these enabling technologies through prototyping and assembling a full scale demonstrator to be validated with a simulator and tested on sky.

This work is sponsored through a grant from project 671662, a.k.a. Green Flash, funded by European Commission under program H2020-EU.1.2.2 coordinated in H2020-FETHPC-2014

Our strategy is based on a strong interaction between academic and industrial partners. Components specifications and system requirements are derived from our main application. Industrial partners lead the development of enabling technologies aiming at innovative tailored solutions with potential wide application range. The academic partners provide the missing links in the ecosystem, with possible wider collaborations with specialized research teams, through generic tools supporting their own application but in mainstream open source projects.

A standard platform, SPARTA,[2] has been implemented by the European Southern Observatory (ESO) to fulfill the needs of the Very Large Telescope (VLT) instrumentation. This successful project has delivered a standard platform used to develop the real-time control system of 10 VLT instruments in operation or in development. However, scaling up the current SPARTA implementation to the size of E-ELT AO system appears to be very challenging considering cost and obsolescence issues of some components.[3] Because the E-ELT AO instruments RTC have to be based on a standard platform for maintainability issues, one of our primary goal is to provide the building blocks of the next generation of standard RTC platform. Incidentally, our philosophy and part of our specifications are greatly inspired by the initial guidelines for the SPARTA concept developed more than 10 years ago. However, the output of the Green Flash project is not an evolution of SPARTA in itself. We propose to develop new core components and an architecture consistent with a possible evolution of this platform.

The project has several objectives in various areas :

- prototype a cluster, able to reach a sustained performance of 1.5 TMAC/s of computing power and processing 150 Gb/s of streaming data with a maximum jitter of 100 μs over 1s of operation
- provide a Smart interconnect solution, based on the FPGA technology, compatible with existing high performance switch solutions, relying on standard serial protocols (TCP/UDP through 10G Ethernet), and supported in mainstream non proprietary middleware
- assess the performance of control matrix computation algorithms and the control matrix upload to the controller with minimal introduced latency and jitter in the control process
- prototype a main board, hosting a high cell density FPGA with an integrated ARM-based HPS, including a PCIe Gen3 rootport as an internal interface, and using 10G Ethernet and 40G Infiniband as network interface and back plane
- build a small scale cluster by interconnecting several prototype microservers through a standard network protocol (10G Ethernet)
- assemble a full functionality prototype for an AO RTC, scalable to the dimensioning of the E-ELT first light instrumentation, including a real-time core and a supervisor module
- implement a real-time simulator, designed to emulate the AO system I/O with various levels of accuracy, using existing AO sub-systems models
- fully characterize the AO RTC prototype performance with several configurations (single and multi-conjugate AO) and propose a strategy for its integration on sky.

In the following, we review the progresses made in each of these areas during the first half of the project.

2. NEW CORE COMPONENTS BASED ON HIGH DENSITY FPGA

Microgate is responsible for the development of new solutions based on high density FPGA to be implemented in AO RTC. The goal is to develop a solution for the hard real-time data pipeline with FPGA based boards for realizing a stackable, energy efficient micro-server for data-intensive applications. More details on this work are given in.[4] The main results are outlined below.

During the preliminary design phase (performed before the granting of the Green Flash project by the EC) we defined the internal requirements and the architecture of the micro-server system. The solution implemented comprises the design and prototyping of two different boards based on FPGAs.

One board, called μXComp, will act as a computational board that can perform the real-time computation in a deterministic way with low latency and low jitter. To guarantee these performances, the board will be based on the Arria 10 FPGA by Altera, embedding >1500 DSP cores, each capable of performing a full MAC operation in one cycle. Moreover, the board is equipped with the Hyper Memory Cube technology that allows memory transfer rates up to 10x faster than SDRAM DDR4 technology.

The second board, called μXLink, will have an FPGA with an hard-wired ARM processor in the same chip (SoC) and is used as an interface and control board to connect to several computational boards and to WFSs and DMs. These board will also be based on the Arria 10 device, but in its SoC version.

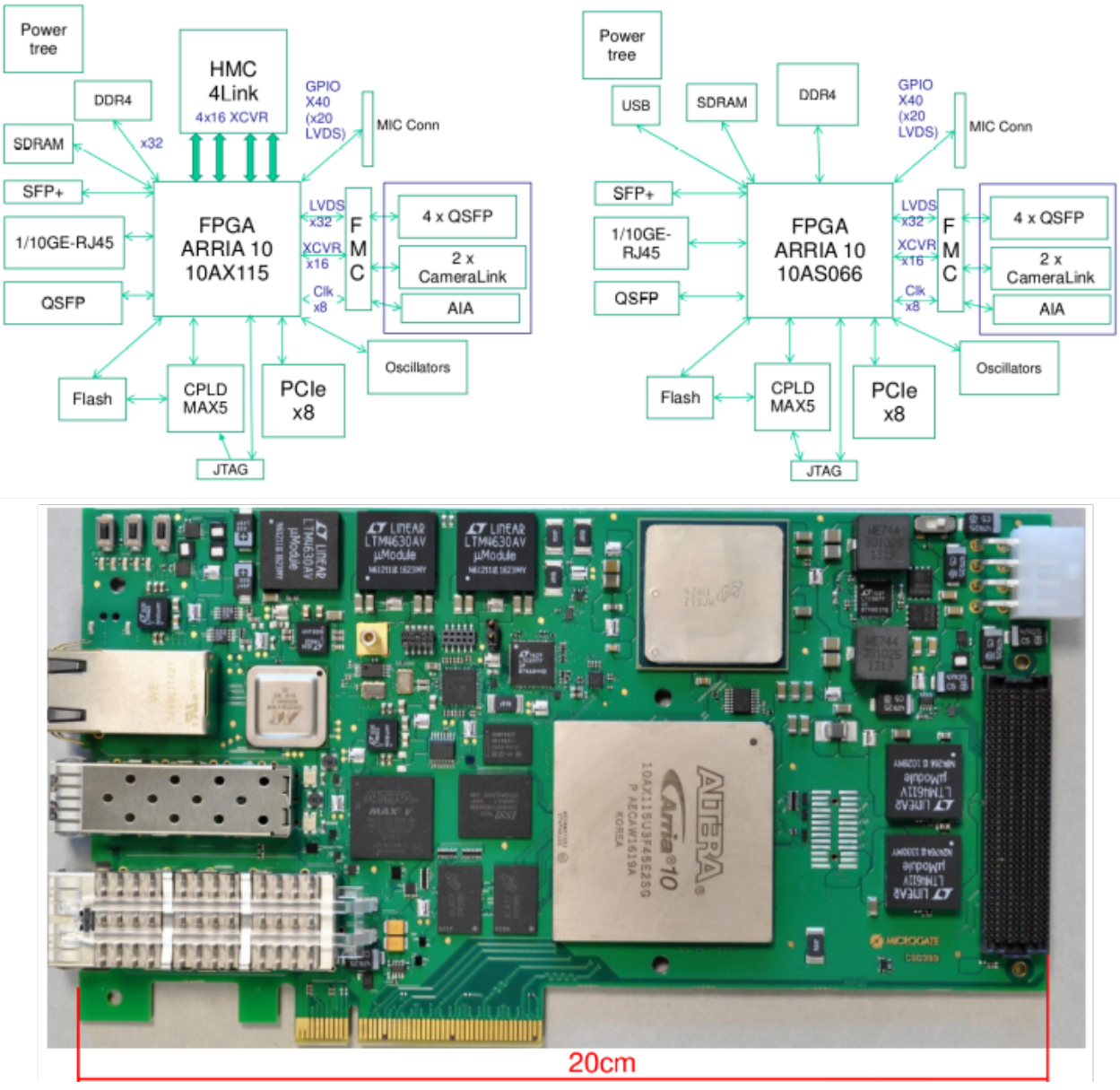

Figure 1: Block diagrams for the two boards developed at Microgate and picture of the first μXComp board.

During the first half of the prototyping period, the HW design of the μXComp board was completed; the routing of the board was finalized as well as the component procurement to fabricate the first prototypes. The

first prototype of the μXComp was also produced by Microgate and is one of two types of FPGA boards to realize the hard real-time data pipeline in a micro-server. The first prototype of the two FPGA boards, the μXComp board is manufactured and is currently under test.

On the firmware side, a preliminary version of the PCIe board drivers for the Linux OS was completed and some sample AO designs at low level were implemented, i.e. not using yet the Quickplay tool by PLDA. We also implemented preliminary input and output interfaces for the real-time pipeline.

The test of the μXComp board has confirmed that the hardware design performs as expected or better. In particular all tests related to the signal integrity show excellent results, which is clear indication of the design level of the printed circuit board and of the quality of the assembly. Initially the FPGA/HMC data transfer throughput did not perform at all as expected. After hardware problems have been excluded, and Microgate has interacted at lot with Intel and Micron support to investigate the problem and solved it. The μXComp board is now a versatile computational board with large interconnect capabilities and high end computational performance.

3. HARD REAL-TIME WITH HARDWARE ACCEERATORS

The goal of this part of the project is to investigate a classic accelerated architecture for AO real-time control where the system is based on a standard CPU server architecture accelerated by various alternative technologies: GPU, MIC and FPGA. The advantage of such a system is that it is based entirely on commercial off-the-shelf components and the accelerator hardware can be abstracted from the software and upgraded with new technology as it emerges. The advantages and disadvantages of the three technologies are being assessed in terms of throughput, latency and jitter along with ease of programming.

3.1 GPU technology

Following the recent evolution of the HPC market, relying on off-the-shelf hardware accelerators such as GPUs as a replacement for DSP boards seems to be a good option. The main concerns with this strategy are:

- the number of accelerators required (related to the application specific performance of a single unit) and the real achievable performance in a distributed memory configuration
- the GPU I/O data transfer latency
- the long term maintainability considering the current new HPC boards release rate and more generally the long term strategy of these hardware vendors with respect to the HPC market

Our goal in Green Flash is to provide

- a study of the compute performance achievable on the AO real-time pipeline in both a single and a multiple GPU environments and of actual pixel data transfer bandwidth achievable from an external device, with very high timing accuracy measured using a FPGA board
- a strategy to optimize the overall pipeline in a single and multiple nodes environment
- a study of up-scaling strategies and foreseen performance with several implementation options

In order to meet the specifications, the goal is as much to maximize performance on a single iteration as to minimize the jitter on this peak performance, including data transport. We chose a very low level approach using a FPGA based network and used persistent kernels to handle all the computation steps that include pixel calibration, slopes and command vector computation. This approach simplifies the latency management by reducing the communication but leads us to re-implement an entire AO control loop and some GPUs standard features : communication mechanisms (guard, peer-to-peer), algorithms (generalized matrix-vector multiplication, reduce/all reduce) and new synchronization mechanisms on a multi node - multi GPU system. Thanks to the use of this strategy coupling custom (direct) FPGA-GPU data transfer and persistent kernels on the GPU,

we were able to demonstrate very low jitter on a realistic pipeline, dimensioned to the SCAO case on the E-ELT. The figure below shows achieved jitter levels of the order of tens of μs for communication and synchronization between the GPUs, which is well within specifications. More details on this work is described in Bernard et al. (this conference).[5]

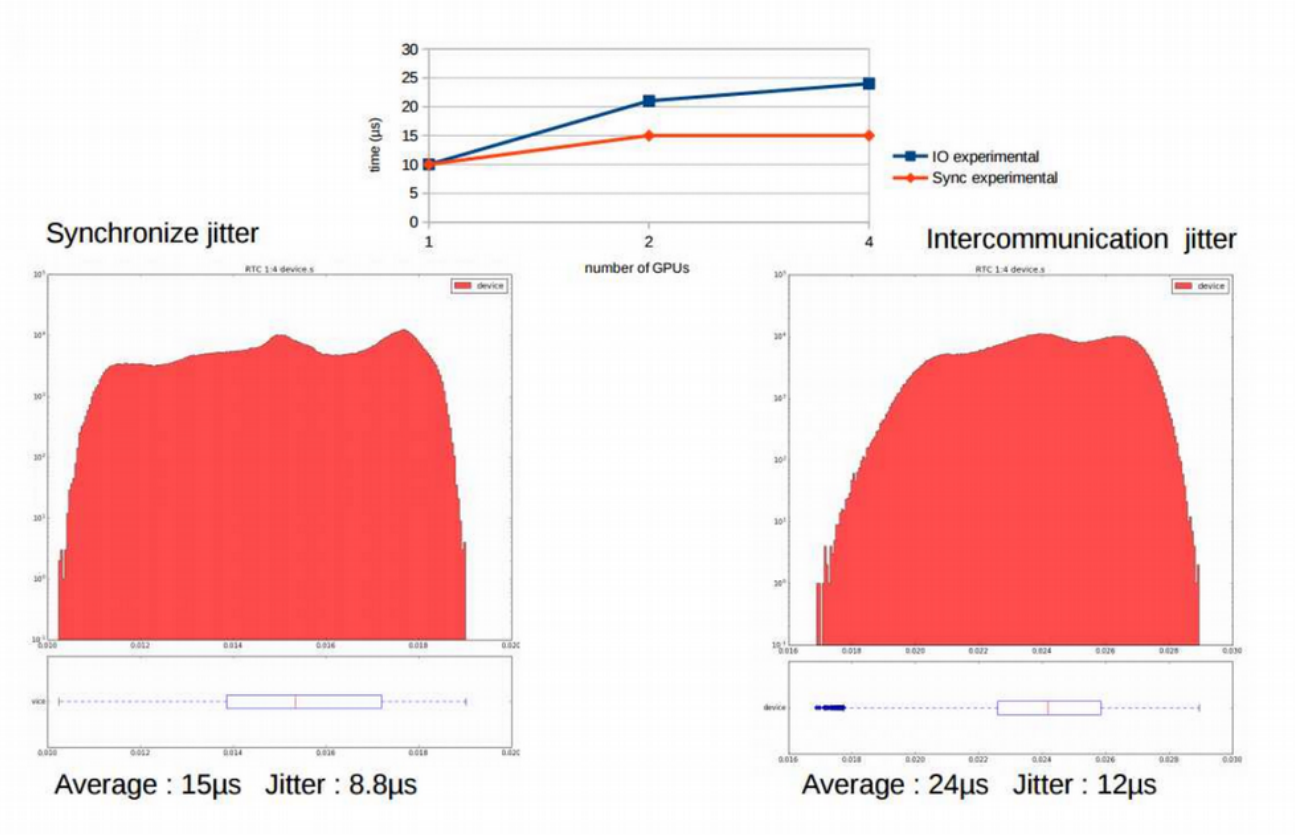

Figure 2: Jitter levels obtained with synchronization and GPU-GPU intercommunication in a multi-GPU pipeline.

3.2 Many Integrated Cores technology

The initial results for the MIC device (Xeon Phi Knights Corner) demonstrated the practicality of using this technology in terms of latency, but were less encouraging in terms of the jitter which is extensive even when a real-time version of Linux is run on the host CPU system.

While this first series of results were obtained with the first generation of Xeon Phi processors (Knigths Corner), additional tests were conducted using the newest generation of processors (Knights Landing, a.k.a. KNL). An important difference between the Knights Corner and the Knights Landing is that the Knights Landing system is not an accelerator, but rather, the core CPU of a server is a Knights Landing. Therefore we do not need to consider aspects such as data offload to an accelerator, and the system design is greatly simplified. The Xeon Phi KNL has been tested with a simple code performing basic RTC operations including image calibration, slope calculation and reconstruction. For an SCAO ELT case (74x74 subapertures) a frame rate of 1.2 kHz has been achieved (800s computation time). This simple case is to provide a maximum performance benchmark but is not pipelined and is therefore not suitable for real RTCS operation.

Figure 3 shows the MVM computation time for various matrix sizes. We see that this scales quadratically with matrix size as expected, though (probably due to the internal implementation of the MKL library used), there is some variation from a quadratic scaling depending on exact matrix size.

In addition to optimizing existing hardware for AO RTC we have also ordered the hardware for a full prototype system. This system consists of Xeon Phi server including 4 x 7210 and 4 x 7250. We also have a 10Gb/s camera which will be interfaced to the system over 10Gb/s ethernet. This solution will be assessed during the second half of the prototyping period.

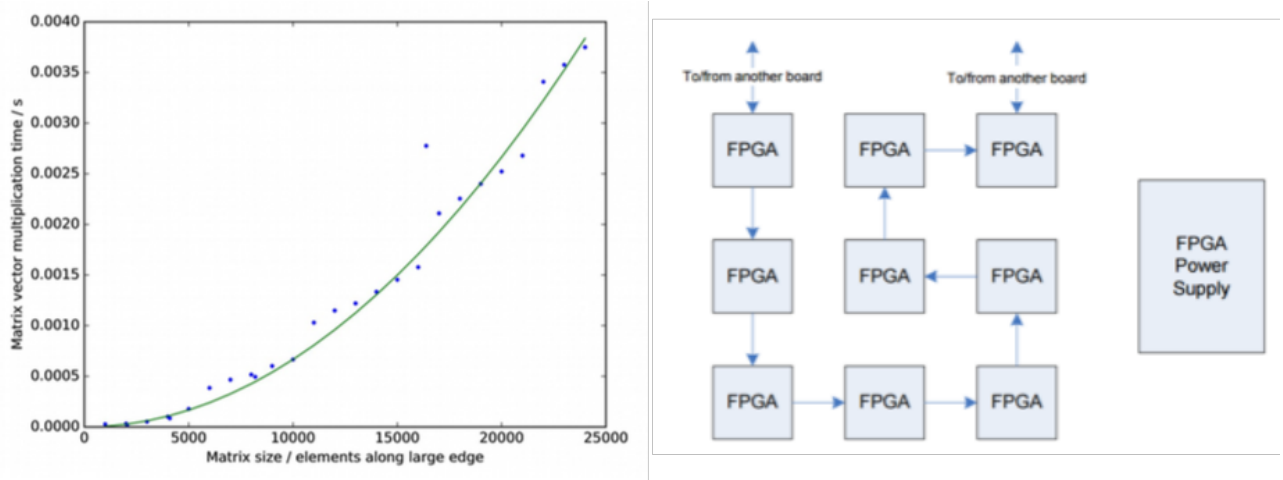

Figure 3: Left, matrix-vector multiplies performance on Xeon Phi KNL, versus various matrices sizes and right, the daisy chained FPGA concept based on COTS FPGA boards.

3.3 COTS FPGA accelerators technology

The FPGA accelerator option is of great interest as it is capable of providing a system with close to zero jitter which is a requirement for some proposed E-ELT instruments. The main barrier to such systems is the length of the FPGA firmware development cycle. The nature of the software produced is substantially different to that for GPU and MIC devices. However, we are designing an FPGA cluster that will be capable of competing with those technologies.

The HPC world is already using arrays of FPGA to provide rapid application throughput. However, this technology is particularly applicable to Green Flash where determinism (low jitter) and low latency are critical. The existing SPARTA real-time system from ESO does all camera pixels handling in FPGA.

A COTS FPGA cluster is studied for the real-time data pipeline, especially the demanding wavefront reconstruction algorithm. The cluster is designed to have a scalable multiple FPGA architecture. The performance can be improved by adding more FPGA hardware and each FPGA node runs identical firmware. The FPGA will provide low latency and very low jitter, ideal for systems that require such time-deterministic characteristics. A standard protocol such as UDP based on 10GbE can be used for communication to the rest of the system.

This FPGA cluster is designed as an accelerator for the AO control loop data processing, especially for the reconstruction. The main design is in line with the Green flash project requirement. Some features are highlighted as :

- Scalability: both the communication bandwidth and the processing power can be adjusted for different computing requirement or different budget;
- Upgradeability: When new hardware is available, the FPGA application can be upgrade to the new platform with relatively small effort;
- Programmability: An FPGA application can be constructed without much concern about the running hardware or low level signal timing, which should dramatically shorten the development time and/or cost.

Other cluster criteria, such as reliability and manageability, would still remain as key specification.

This design of a FPGA cluster consists of an access node and many internal work nodes. The single access node provides an external interface for accessing the cluster computing resource. The communication of such interface is standard UDP, preferably through a 10GbE Ethernet port. This node translates the external request to internal command and communicates with the internal working nodes. The access node and work node are logical nodes, which means an access node and a work node can possibly coexist on the same piece of hardware, especially considering that many AO applications are memory bandwidth constrained, while the translation between the external request and the internal command does not require large buffer. The daisy chain cluster is simple to implement and manage. Although the processing power has good scalability, its communication bandwidth is almost fixed. The proposed FPGA solution is presented in Figure 3.

4. SUPERVISION PIPELINE

The AO loop optimization aiming at providing the real-time AO data pipeline with regular tomographic reconstructor matrix update is the most computationally challenging part of the AO RTC. In the first half of the project, our team has produced a supervision design & test report, which includes :

- the definition of the supervisory pipeline in terms of input / output, critical algorithms and data movement, following the Learn & Apply approach for tomographic AO
- the definition of a new optimized multi-GPU approach for the Learn step (i.e. turbulence parameters identification) and some preliminary tests
- a study of the tomographic reconstructor computation (i.e. Apply step) complexity and achievable time to solution on currently available hardware
- a study of up-scaling strategies to meet the E-ELT requirements in terms of problem dimensioning and time-to-solution for the supervision pipeline, and several implementation options

The main conclusion of this comprehensive study of the supervision pipeline if ported to GPUs, is the large number of GPUs required to achieve adequate performance at the E-ELT scale.

A full pipeline, following the so-called learn & apply approach, including the experimental covariance matrix generation using noisy data from the instrument, includes the identification of critical turbulence parameters through a fitting process of the latter on a theoretical model and the computation of the corresponding tomographic reconstruction The supervisor module is thus a mix of cost function optimization for parameters identification (*Learn* process) and linear algebra for reconstructor matrix computation (*Apply* process). The output of this work is described in details in Doucet et al. (this conference).[6]

The parameters identification stage (*Learn* process) is intended to fit the measurements covariance matrix on a model including system and turbulence parameters. To do so, we use a score function optimized using the Levenberg-Marquardt algorithm. A multi-GPU implementation of the latter has been produced during the first phase of the project. It consists of a dual stage process. In the first pass, only a limited number of turbulence layers are considered (5 layers) and in the second pass more layers are added to meet the system specifications (up to 40 layers). Initial performance analysis was done on a realistic E-ELT MCAO case on the multi-GPU DG-X1 platform.

The time-to solution reached is 240s (4 minutes) including 25s for the first pass and 213s for the second pass. The weak and strong scaling of this process have been assessed and show an excellent behavior, very close to the perfect case for weak scaling and an impressive ¿90% efficiently for strong scaling on multiple GPUs. This is very encouraging for the supervision strategies on the E-ELT since the time-to-solution is already very close to the initial instrument specifications (with a reconstructor update every minute).

Concerning the reconstructor matrix computation (*Apply* process), it aims at computing the tomographic reconstructor matrix using covariance matrix between truth sensor and other WFS and invert of this covariance matrix. Several methods have been assessed: LU or Cholesky factorizations or *brute force* using a direct solver.

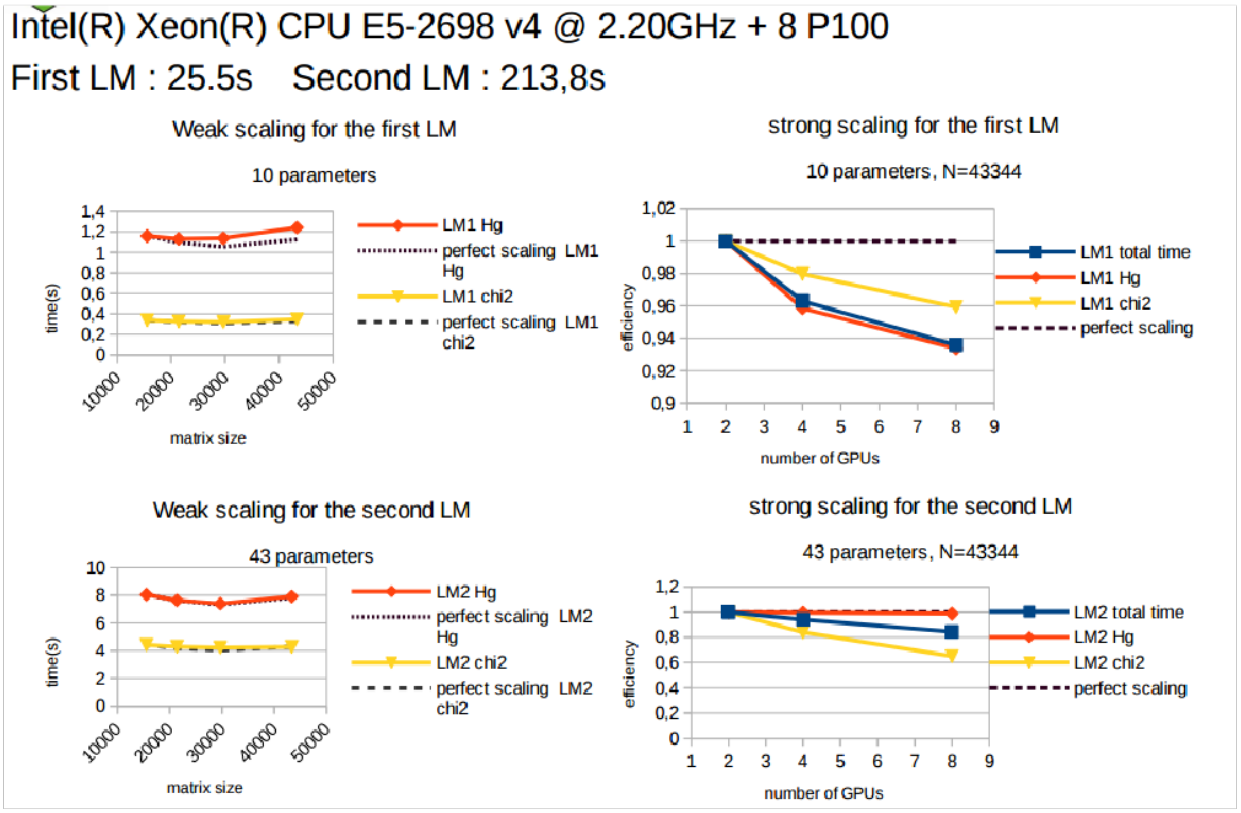

Figure 4: Performance scalability of the Learn pipeline.

The latter seems to be the most efficient one since it is mostly compute-bound and exposes a high level of scalability. In the context of the Green Flash project, and in collaboration with the Extreme Computing Research Center at KAUST, we have developed an efficient implementation of the control matrix computation for AO on multicore system with multiple GPUs using high-performance numerical library for solving large dense linear algebra problems. The code was built to be highly portable, so as to explore various architectures by using standard vendor provided maths libraries.

Our results show that GPUs can deliver better peak performance. The saturation is not reached and we expect >2.5 or better with larger matrices. Moreover, the NVlink interconnect seems to perform very well in this multi-GPU platform. Finally, record time-to-solution is obtained on the DGX-1 server from NVIDIA, as a MAORY / HARMONI full scale process (100k x 100k matrix) is addressed in 25sec to compute tomographic reconstructor, which is well within the system specifications.

The final implementation should be consistent with the required rate of operation and its performance should be assessed on various architectures for future down-selection as the instruments enter the final design phase.

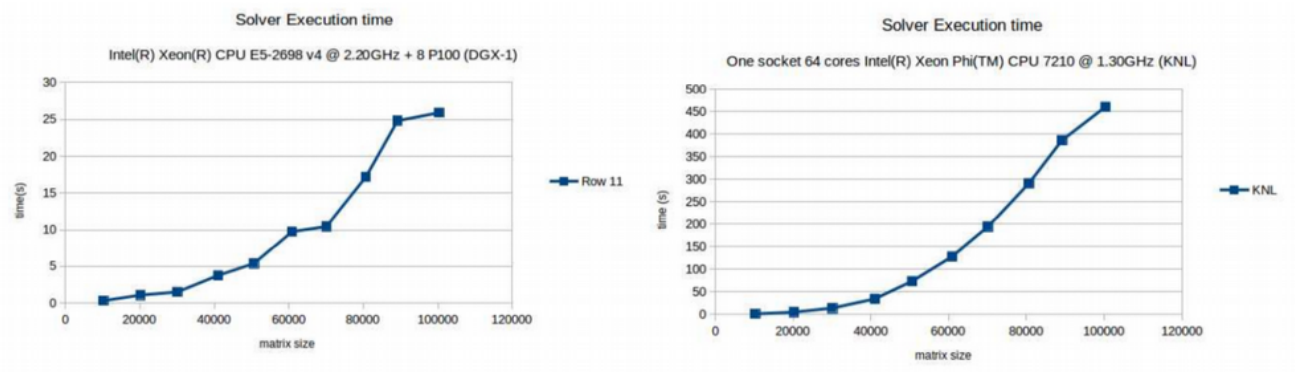

Figure 5: Performance obtained for the tomographic reconstructor computation on both NVIDIA GPUs and Intel Xeon Phi accelerators.

5. SMART INTERCONNECT CONCEPT

Meeting the real-time constraints with COTS accelerators involves not only efficient implementations of algorithms on the accelerators but also an efficient interconnect strategy to minimize intra- and inter-nodes data transfer latency. A holistic approach in the system design is required to handle both the data streams from sensors and the inter-nodes communications and to maintain adequate overall throughput with deterministic performance.

We have proposed to develop a concept of smart interconnect, tailored to the AO application, enabling new features at the Network Interface Controller (NIC) level in order to:

- Locally reduce the bandwidth requirement in the cluster by performing simple data reduction tasks on the NIC
- Optimize data broadcasting / re-routing depending on environment topology and occupancy
- Handle both data streams from sensors and nodes inter-communications in a unified approach

This work aims at providing a high performance low latency interconnect solution, based on standards and exploring the use of new smart features in the context of real-time control for AO. The NIC could be defined as an actor of the data flow processing, available at the programming model level to leverage both RDMA capabilities for efficient data routing or smart broadcasting and generic computing blocks for acceleration.

Enabling smart features on the NIC can be done statically, by providing the user access to a predefined collection of features, included in the NIC design, through a driver and API interoperable with standard middleware. We propose to follow another approach in which the user can build, in a unified and simplified framework, his own data flow design on the NIC (including communication protocols and computing blocks) and the driver and end-user API to provide access to these features in the programming model. This can be made possible through the use of an integrated development environment for the FPGA board including FPGA design, High Level Synthesis (HLS) for custom computing blocks, simulation, hardware emulation and end-user API development platform. This is done through complementing the ecosystem of an existing integrated FPGA development environment (QuickPlay from Accelize / PLDA), by providing data handling and computational blocks tailored to our application, and support for several FPGA options and board designs.

During the first half of the prototyping period the unified architecture of the Smart Interconnect system has been defined, with respect to following criteria:

- Genericity
- a unique architecture shall cover the whole set of interconnection requirements

- adaptability is relying on the use of FPGA technology and advanced FPGA application programming environment
- Scalability achievable day 1 using COTS elements
- smooth migration to state of the art technology is enabled

At development environment level, the following achievements were reached on the QuickPlay tool:

- Improved maturity through: improved development flow, easing and speeding up IPs and boards integration and allowing integration of 3rd party IP providers.
- Improved Performances and features through: improved proprietary HLS through directives usage, allowing higher computing performances, Vivado HLS integration, improved SDK performances (PCIe and TCP layers), improved emulation and configuration (static or dynamic) of C and HDL Kernels (
- Improved Tool accessibility, allowing SW engineer to easily target application on FPGA
- Improved genericity through increased target families (Altera Stratix V / Arria 10, Xilinx Kintex-7 / Kintex-US, Virtex-US) and BSP delivery for an increased set of boards (Reflex XpressGX5, Reflex XpressGXA10, Reflex - XpressK7 160, Reflex - XpressK7 325, Reflex XpressKUS, Xilinx KC705, Xilinx KCU105, Bittware A10PL4, Microgate - XComp)

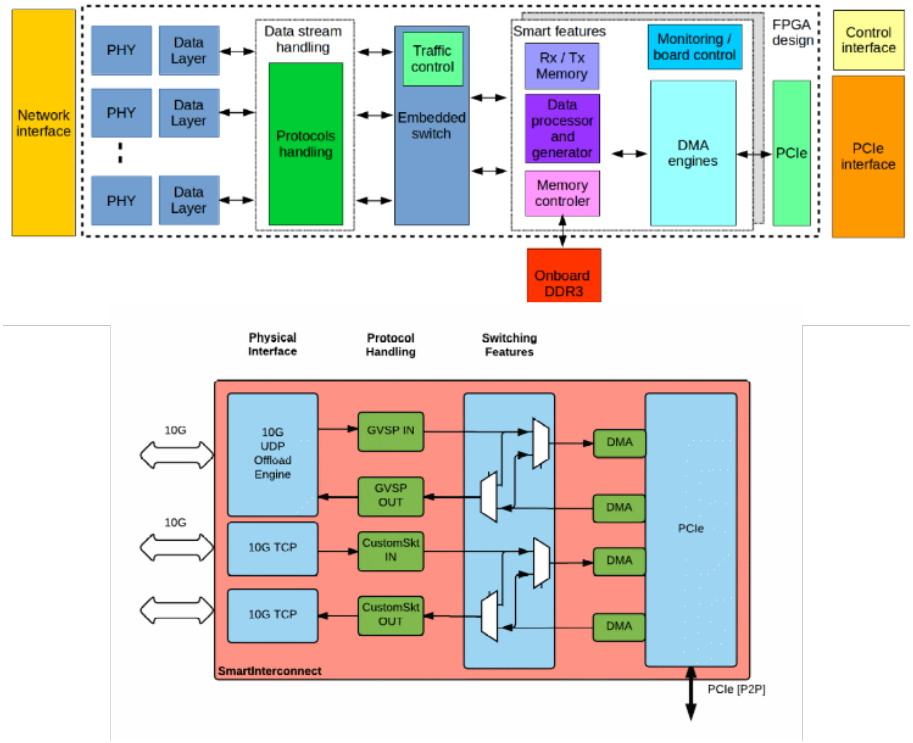

Figure 6: The smart interconnect generic design.

The Development of a High Bandwith FPGA NIC card was driven by QuickPlay tool enhancement, and came to reality through Smart Interconnect development, which is a perfect application example since it requires all features proposed for such smart FPGA NIC.

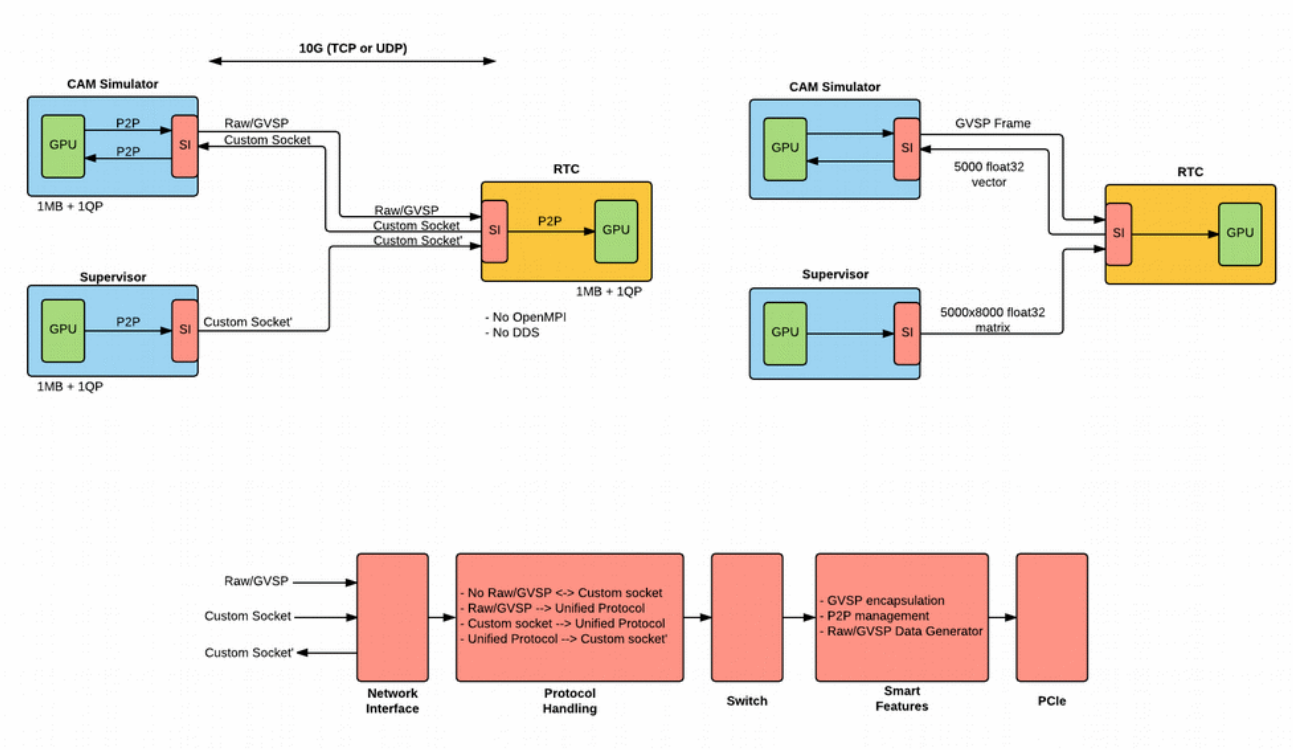

Figure 7: SCAO scale prototype using smart interconnects (SI).

Indeed, the Smart Interconnect development required: BSP availability for multiple boards; integration of specific IPs and features with TCP/UDP offload engines at 10 Gbps, PCIe gen3, DDR4 memory (and HMC to come); data processing capability, intrinsic to FPGA and made simpler through QuickPlay and C-Kernels usage

The Smart Interconnect prototypes allowed to demonstrate QuickPlay Genericity, allowing design to be targeted on any of the supported board with performance difference and maximal TCP/UDP performances, with 9 Gbps of effective data transported and a latency and jitter fully compliant with Green Flash project requirements.

To date, the following features and performances of the Smart Interconnect have been demonstrated:

- Gen3 x8 PCIe End-Point with DMA and Peer-to-Peer capability (tested with Tesla K40 GPU)
- 10G Ethernet interfaces with full TCP/UDP bandwidth (9 Gbps of effective data carried)
- Real-time GeV (GVSP) and matrix (CSKT) encoding/decoding through C-Kernels
- Simple switching features through HDL Kernel integration

6. MIDDLEWARE EVALUATION

One of the lessons learned from the existing ESO real-time system middleware development is the requirement for abstraction and virtualisation in the system. The key to efficiency in what will be a distributed system, and to abstraction and portability, is middleware. In standard HPC super computers this is often MPI. However, we have additional requirements on latency and jitter, which vary between the different system modules.

Various commercial middleware solutions exist and we have now completed an initial down-selection of these, based on the system requirements. There are three identified middleware domains : Control, Telemetry and Low-latency pipeline.

The three middleware domains each have their own requirements.

- Control: Request/reply pattern, Service discovery/location transparency
- Telemetry: Publish/subscribe pattern, Hard throughput, weak latency/determinism requirements
- Low-latency pipeline: Hard latency, determinism, throughput requirements, Fan-out/fan-in pattern distribute workloads

The following evaluations are in progress:

- Control: DDS, ICE Telemetry
- Telemetry: DDS, ZeroMQ/Google protocol buffers
- Real-time pipeline: Real-time pipeline ZeroMQ, MPI

For the real-time pipeline the goal is to limit the total RTCS latency to one frame, at a frame rate of 1kHz this equates to a total latency of 1000s. The goal for jitter on the latency is 100s in any 1 second period. The latency in the middleware must therefore be significantly less than these values.

ZeroMQ was evaluated for the real-time pipeline. The mean latency obtained, even for small message sizes, is unacceptable. The jitter on the latency is also unacceptably high. A similar study was led with MPI, that shows that for small message sizes the mean latency is less than 50s with acceptable jitter. More tests will be conducted in the remainder of the project to assess these various middleware technologies and their applicability to each of the middleware domains.

7. CONCLUSION

This paper reports on the results obtained at mid-term thanks to the various developments that are being led in the context of the Green Flash project, towards the study of a generic and scalable architecture for future E-ELT AO RTC. We are now entering the final phase of the project in which a down-selection process will be led between the various technologies contemplated for each sub-system, on the basis of the prototyping results. It will lead to the definition of a final prototype architecture to be implemented and tested with a simulator as the final objective of the project.

REFERENCES

[1] Gratadour, D., Dipper, N., Biasi, R., Deneux, H., Bernard, J., Brule, J., Dembet, R., Doucet, N., Ferreira, F., Gendron, E., Laine, M., Perret, D., Rousset, G., Sevin, A., Bitenc, U., Geng, D., Younger, E., Andrighettoni, M., Angerer, G., Patauner, C., Pescoller, D., Porta, F., Dufourcq, G., Flaischer, A., Leclere, J.-B., Nai, A., Palazzari, P., Pretet, D., and Rouaud, C., "Green FLASH: energy efficient real-time control for AO," in [*Adaptive Optics Systems V*], *Proc. SPIE* **9909**, 99094I (July 2016).

[2] Fedrigo, E., Bourtembourg, R., Donaldson, R., Soenke, C., Valles, M. S., and Zampieri, S., "Sparta for the vlt: status and plans," (2010).

[3] Enrico Fedrigo, R. D., "Sparta roadmap and future challenges," (2010).

[4] Patauner, C., Biasi, R., Andrighettoni, M., Angerer, G., Pescoller, D., Porta, F., and Gratadour, D., "FPGA based microserver for high performance real-time computing in Adaptive Optics," in [*Adaptive Optics for Extremely Large Telescopes V (AO4ELT5)*], xx (Oct. 2017).

[5] Bernard, J., Lainé, M., Perret, D., Sevin, A., and Gratadour, D., "A GPU based RTC for E-ELT Adaptive optics : Real Time Controller prototype," in [*Adaptive Optics for Extremely Large Telescopes V (AO4ELT5)*], This conf. (Oct. 2017).

[6] Doucet, N., Gratadour, D., LTaief, H., Gendron, E., Sevin, A., Ferreira, F., Vidal, F., Kriemann, R., and Keyes, D., "Efficient supervision stragtegy for tomographic AO systems on E-ELT," in [*Adaptive Optics for Extremely Large Telescopes V (AO4ELT5)*], This conf. (Oct. 2017).

[7] Melani, A., Serrano, M. A., Bertogna, M., Cerutti, I., Quiñones, E., and Buttazzo, G., "A static scheduling approach to enable safety-critical openmp applications," in [*Design Automation Conference (ASP-DAC), 2017 22nd Asia and South Pacific*], 659–665, IEEE (2017).