GTCAO Real Time Control System software design

J. Marco de la Rosa[a], J. Rosich[a], M. Núñez Cagigal[a], O. Tubío[a], J.C. López[a], L. F. Rodríguez-Ramos[a], A. Basden[b], J. Patrón[a], I. Montilla[a], R. L. López[a], M. Puga Antolín[a], R. Simoes[a], F. Tenegi[a], J. Sánchez-Capuchino[a], M. Reyes[a], V.J.S. Béjar[a].
[a]Instituto de Astrofísica de Canarias (IAC).
[b]Durham University.

ABSTRACT

The GTC Adaptive Optics (GTCAO) instrument will provide single-conjugate natural guide star adaptive optics capabilities to the FRIDA instrument installed in the GTC telescope. The instrument implements a real time closed loop with a Shack-Hartmann wavefront sensor and a deformable mirror. Besides that standard operation, the instrument sends information to GTC for performing fine control of the telescope primary and secondary mirrors for desaturating the deformable mirror and optimizing its performance. This paper presents the software design for the Real Time Control System in charge of implementing all the functionality related with the real time control loop.

Keywords: GTCAO, GTC, adaptive optics, real time, software.

1. INTRODUCTION

GTCAO is the instrument that provides single-conjugate Adaptive Optics (AO) to the FRIDA [1] instrument designed for the GTC telescope in La Palma. From the software point of view, the instrument is controlled by the AOCS ({GTC} Adaptive Optics Control System). This software is in charge of controlling and coordinating the functioning of a set of mechanisms, primarily necessary for managing the opto-mechanical interfaces with the telescope, and the functioning of the AO real time control loop. The wavefront sensor (WFS) and the deformable mirror (DM) are pieces of hardware that need to work in a tightly coupled manner in order to comply with the real time (RT) requirements. Besides, the computation needed for the RT control loop is highly demanding, so a hardware providing high computing power needs to be also coupled with them. For ease of organization within the project and given the high coupling and complexity of coordinated functioning of all these components, they are treated as a complete system themselves. This system is called RTCS (RT Control System) and is in fact a subsystem of the AOCS.

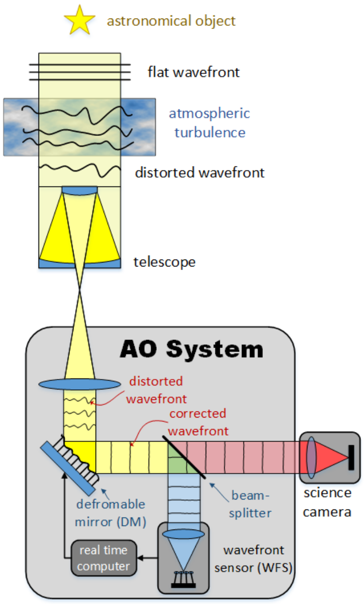

Figure 1. Schematic for sinlge-conjugate adaptive optics system [2].

1.1 The GTC Control System

GTC implements a distributed control system called GTC Control System (GCS). GCS software is based on CORBA and provides many features commonly needed by the software components that control devices and instruments integrated with the telescope. In a simplistic way, from the GCS point of view, an instrument is a set of coordinated devices. The GCS provides an instrument sequence execution application called the Sequencer, standardized GUI development tool, logs, alarms, automatic periodic variable monitoring, a standard set of libraries and a set of software components for interacting with the telescope. All those features will be used when applicable in order to simplify development and build robust software components that are coherent with the rest of software in GTC.

In particular, the *ObservingEngine* software component in the GCS is key for the GTCAO system. This component manages communication between different software components and the telescope mirrors and axes. It also performs coordinate system conversions and other related functions. GTCAO needs to send specific information for deformable mirror desaturation and for guiding to the *ObservingEngine*, which will distribute the different commands to the appropriate mechanisms in the telescope.

1.2 DARC

DARC is the Durham AO Real Time Controller [3], [4]. It is a complete and modular open source software that implements a real time control system (RTCS) for adaptive optics, developed by the Durham University. This software is powerful, flexible and extendable, and the support from the development team at Durham is outstanding. A great amount of the functionality related with the AO closed loop needed by GTCAO is implemented in DARC out of the box. Specific modules need to be developed for interfacing with the camera and the DM, as well as for fine tuning the loop controller. That aside, all the main loop structure, commanding, telemetry, command line UIs and GUIs are already implemented and in a very sensible way, adequate for use in RT architectures and with high performance computing in mind. Durham University also develops The Durham AO Simulation Platform (DASP) [5] which implements many simulations, including WFSs and DMs that can be attached to DARC in order to test the AO closed loop control.

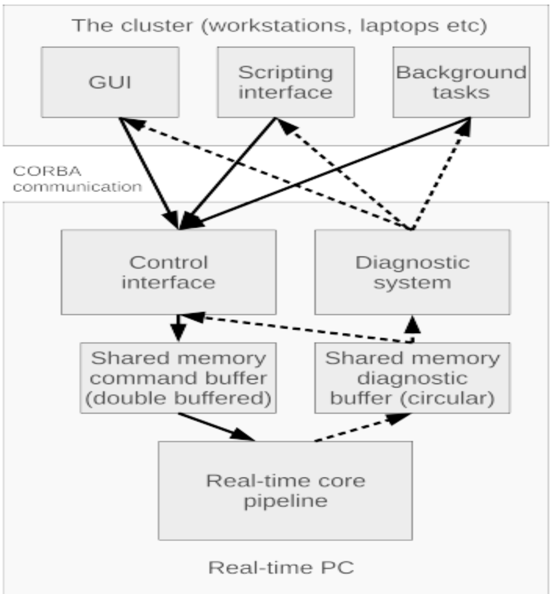
The cluster (workstations, laptops etc)
GUI
Scripting interface
Background tasks
CORBA communication
Control interface
Diagnostic system
Shared memory command buffer (double buffered)
Shared memory diagnostic buffer (circular)
Real-time core pipeline
Real-time PC

Figure 2. DARC software architecture overview.

The telemetry streams provided by DARC out of the box include: raw and calibrated pixels, centroids, mirror reconstruction results, mirror demands sent to actuators, loop status; and others specific to the loop configuration. The telemetries can be extended by programming new modules or changing the standard ones.

1.3 Main hardware

1.3.1 WFS Camera [6], [7]

Figure 3. OCAM2 camera by First light.

Model	OCAM2
Brand	First light
Sensor model	EM CCD220
Resolution	240x240 pixels
Pixel size	24 microns
max FPS	1500 (2067 in new K model)
Interface	Cameralink (Matrox Radient ECVl, Pcie2.0, x8, 4GB/s)
Plate scale	0.35 arcsec/pixel
FoV	3.5”x3.5

1.3.2 Deformable Mirror (DM) and Deformable Mirror Drive Electronics (DMDE) [8]–[10]

Figure 4. SAM 373 DM + DMDE by Cilas.

Model	SAM 373 DM + DMDE
Brand	Cilas
Actuators number	373 piezoelectric actuators
Resolution	< 3 nm (12 bits)
Actuators disposition	24 microns
Space between actuators	7 mm
Interfaces	sFPDP (Curtis Wright FiberExtreme SL100 (2.5Gbit/s)), RS485
Total stroke	±3,8 microns

1.3.3 RTC host

Figure 5. Superserver 4U SYS-6X8IRD-XQR604U by Supermicro.

Model	Superserver 4U SYS-6X8IRD-XQR604U
Brand	Supermicro
Processor Model	Intel Xeon Processor E5-2650 v3
Number of processors	2
Number of cores	20 (10 each microprocessor)
Processor base clock frequency	2.3GHz
Cache	25 MB SmartCache
Theoretical peak performance	320 GFLOPS
Memory type	DDR4 2133
Memory size	64GB
Max memory bandwidth	68 GB/s

1.3.4 Abbreviations used in this document

AO	Adaptive Optics
DM	Deformable Mirror
DMDE	Deformable Mirrors Drive Electronics
FPS	Frames Per Second
GCS	GTC Control System
GTCAO	GTC Adaptive Optics instrument
LCU	Local Control Unit
M1	GTC primary Mirror
M2	GTC secondary Mirror

MA	GTC Main pointing Axes
r_0	Fried parameter [2], [12]
RT	Real Time
RTC	Real Time Control
RTCS	Real Time Control System
SH WFS	Shack-Hartmann wavefront sensor [13]
τ_0	Coherence time [2], [14]
WF	Wavefront
WFS	Wavefront Sensor

2. RTCS FUNCTIONALITY AND RT REQUIREMENTS

As introduced in the previous section, the main task that RTCS performs is coordinating the WFS sensing with the DM acting in order to correct the sensed WF. This task is called the AO loop and is standard to every AO system. There are multiple different ways of implementing this loop depending on the physical models used and the WF sensing technology used. The approach adopted in GTCAO can be summarized in the following steps:

- Acquire WFS image: read the image from the camera in the SH WFS.
- Preprocess WFS image: perform different preprocesses to the images read from the WFS for improving later processing. Typical examples of such preprocesses are: dark noise subtraction, flat field multiplication, background subtraction, application of threshold, multiplication by pixel weighting, application of power factor apply a threshold.
- Compute slopes: in the SH WFS, a set of subapertures is processed. Each subaperture is normally a square or rectangle inside which a spot of light, corresponding to the projection of the reference star, moves depending on the WF deformation. The centroid of each subaperture is calculated. The difference in position from that centroid to the reference centroid where the spot should be if the WF was not corrugated is called slope [13].
- Reconstruct WF: the slopes obtained in the previous steps are the input to an algorithm that computes how the WF can be corrected. Several of these algorithms exist and this is still an active field of research. In our case we are using two approaches, one for zonal and another for modal reconstruction, both based in the MVM algorithm [15].
- Calculate DM actuation: the output of the previous step is transformed into a physical actuation to the DM actuators. This step can use several types of filters, apply constraints, arithmetic transformations and corrections due to the actual physical state of the DM actuators (offsets, differences in gains, broken actuators, etc.).

All that can be thought of as the standard AO loop functionality. Nonetheless, there are issues that also need to be addressed by the software. These issues can depend on the actual opto-mechanical design of the instrument and on practical operation needs.

In the case of GTCAO, the telescope can improve the DM performance with an appropriate commanding of the primary (M1) and secondary mirrors (M2), and the telescope main axes (MA). M1 mainly provides active optics [16], M2 mainly provides tip-tilt correction [17] and MA act in the pointing. If commanded appropriately, they can relieve the DM of part of the corrections to be applied to the WF and improve the DM dynamic range usage (desaturate). M1, M2 and MA operation frequencies are lower than that of the DM. The reconstruction algorithm can be tuned to use a set of filters in order to separate information going to M1, M2 and MA from the information going to the DM, thus dividing the corrections performed by each element. There will be a set of low command rate (~20-100Hz) corrections (slow tip-tilt and other modes) that will be used to correct the WF, performed by M2; and a set of corrections at even lower command rate (~0.1-

1Hz) that will be used to desaturate the DM, coordinately performed by M2, M1 and MA. Figure 6 refines Figure 1 adding the GTCAO specific control flows just described in and the hardware described in section 1.3.

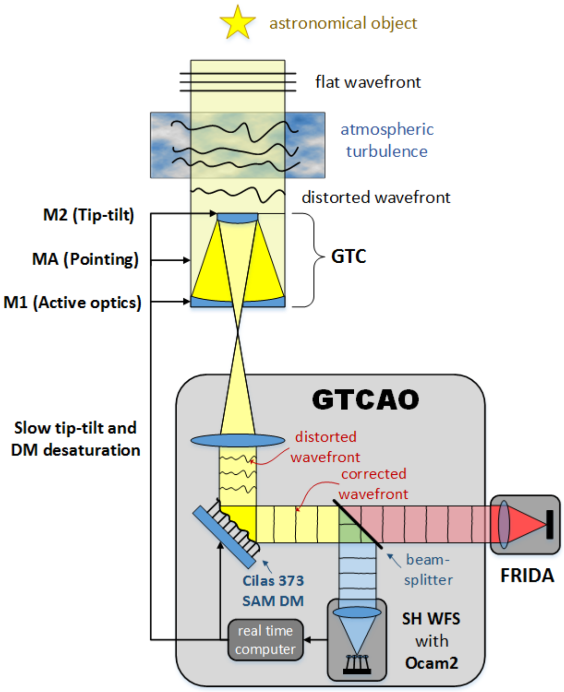

Figure 6. Schematic for the GTCAO adaptive optics system including hardware and addictional control flows.

During observation, the instrument will rotate and the illumination and optical paths of the system will vary. Some of the subapertures can be obscured and the relationship of the centroids displacement with mirror actuation will change. This pupil rotation effect [18] creates the need to monitor the subaperture illumination and rotation of the instrument in order to decide which subapertures to use and to change the control matrixes that are used for WF reconstruction if needed.

There are also practical considerations concerning the closed AO loop operation. Apart from running the loop, the system needs to monitor how the loop is performing. This monitoring should allow detection of anomalous situations and also provide pertinent information to the final user.

An initial reference estimation of the performance needed for the RTC host in order to comply with the AO closed loop RT requirements was calculated. This estimation included: bias correction flat field correction, SH spots processing, control loop computation, reconstruction process and application of the correction to the DM. The main control loop parameters are shown in Table 1 and the results are shown in Table 2. There should be taken into account that there are performance needs that were not considered. Other functionalities that the RTC implements that were not considered include: monitoring WFS subaperture flux, r_0 computation, update of control loop parameters and general monitoring and telemetry functionalities (displaying/sending WFS camera images, DM actuation telemetry, control loop status information, etc.).

Table 1. Main parameters of the AO control loop for required performance estimation.

SH WFS number of subapertures	400
Subaperture size in pixels	10x10
WFS camera FPS	1000
Number of DM actuators	373

Table 2. Estimated performance required for the AO closed control loop execution.

Number of operations per second	2.5GFLOPS
Memory bandwidth	7.3GB/s

3. RTCS SOFTWARE COMPONENTS

This section describes the main software components and their main interfaces. Due to space constraints for this work, a very high level view is presented here so the details of the components and interfaces are not shown. The objective of this view is presenting an appropriate conceptual component division and gathering all the main information flows between components. The presented components and interfaces are intended to support *all* the functionality required for the system. This section includes a global component view and detailed component views for the main components, *GTCAORTCSController* and *DARC*.

3.1 Global component view

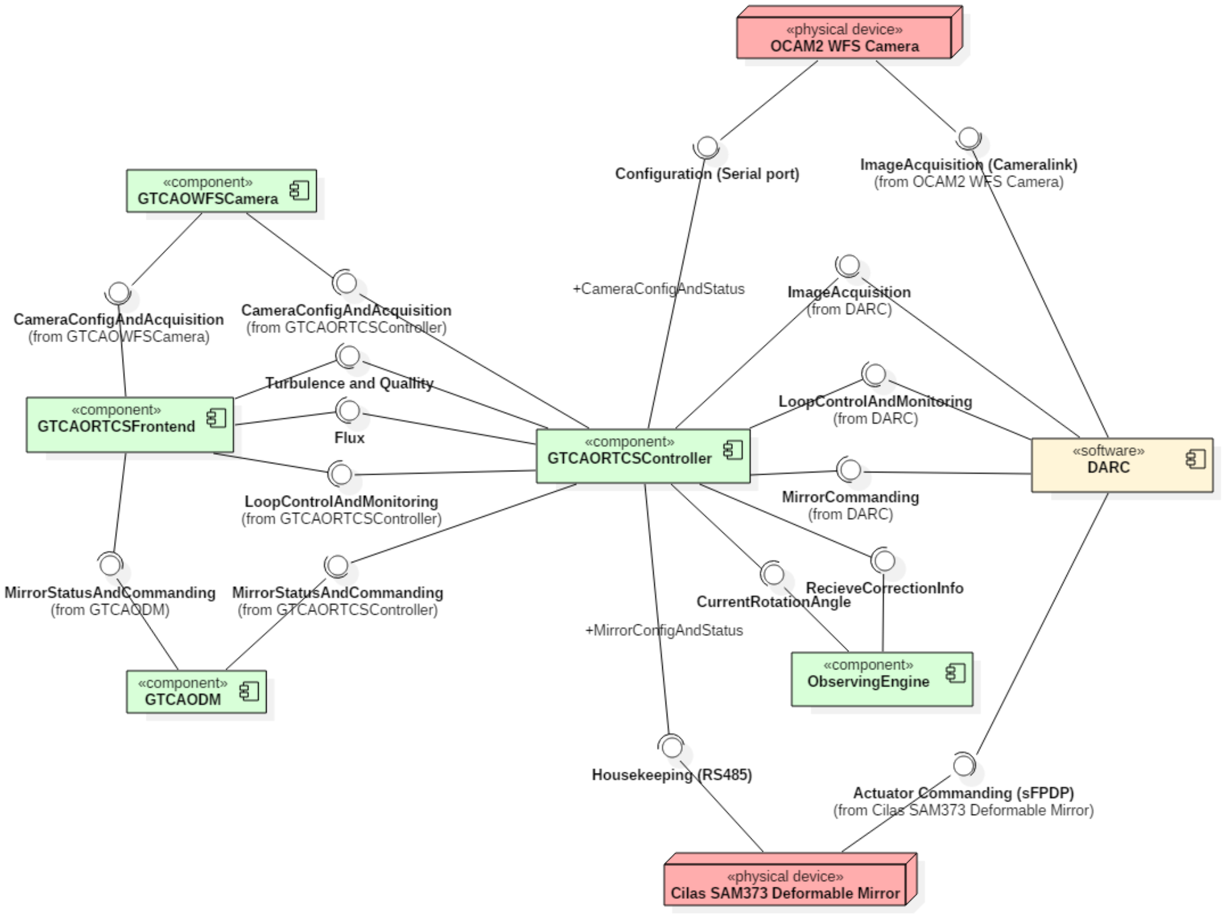

Figure 7. Global RTCS software components and interfaces view.

Note that the independent GCS components (Devices) for the WFS camera and the DM are only frontends for the *GTCAORTCSController* component exposing the appropriate interfaces. This design architecture is due to the restrictions on WFS camera and DM interaction depending on the RT loop status. Centralizing communication with the WFS camera and the DM in the *GTCAORTCSController* eases the coordination of communication with the camera and DM. The *GTCAORTCSFrontend* component will expose the *GTCAORTCSController* interfaces regarding all aspects other than the WFS camera and the DM.

The *ObservingEngine* component is part of the official GCS release. From the GTCAO instrument point of view, the *ObservingEngine* is in charge of receiving the M1 and M2 correction, and MA guiding information generated by the RTCS; and providing information about the telescope alt/az position for pupil rotation correction. Specific software implementing low level functionality and embedded in the *ObservingEngine* need to be implemented by the GTCAO team in order to meet requirements.

The *GTCAORTCSController* component, together with the *DARC* component, are the main components in the RTCS so detailed views for them are also presented.

3.2 *GTCAORTCSController*

GTCAORTCSController (Figure 8) implements all the loop control and monitoring functionalities. It is responsible for coordinating the functioning of all the subcomponents in the RTCS in order to support all the observing modes and the functionalities needed for calibration, configuration and diagnostics. It relies on DARC for the actual implementation and control of the AO RT loop. This component coordinates access to the hardware directly related with the RT loop and also all the non RT loops that handle the AO RT loop parameter tuning during operation.

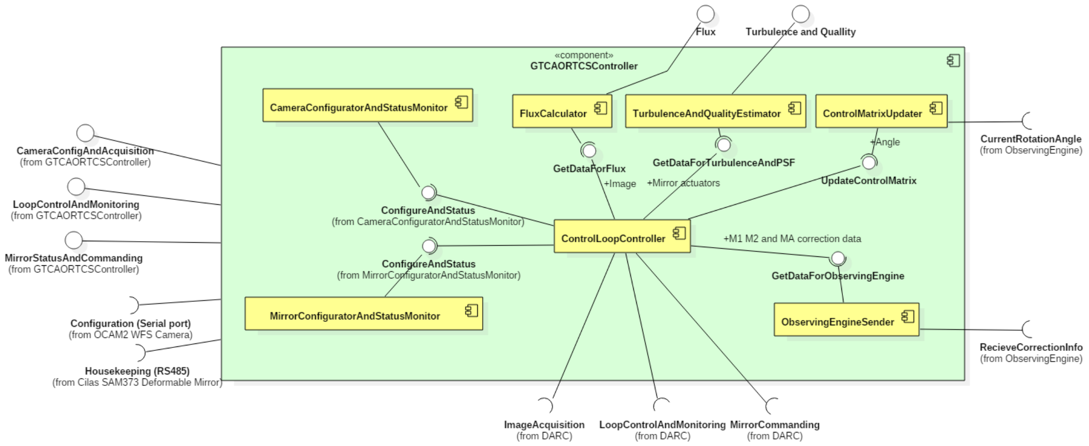

Figure 8. GTCAORTCSController subcomponents and interfaces view.

3.3 *DARC*

The *DARC* component (Figure 9) is a software application developed by the Durham University. DARC stands for “The Durham Adaptive Optics Real-time Controller” and it provides a series of applications and tools to manage an AO RT control loop. DARC follows a flexible and extendable modular design. Some specific modules need to be developed by the GTCAO software team [3], [19]. In particular, the WFS camera, DM and reconstruction modules need to be developed by the GTCAO team in order to implement all the GTCAO specific functionalities.

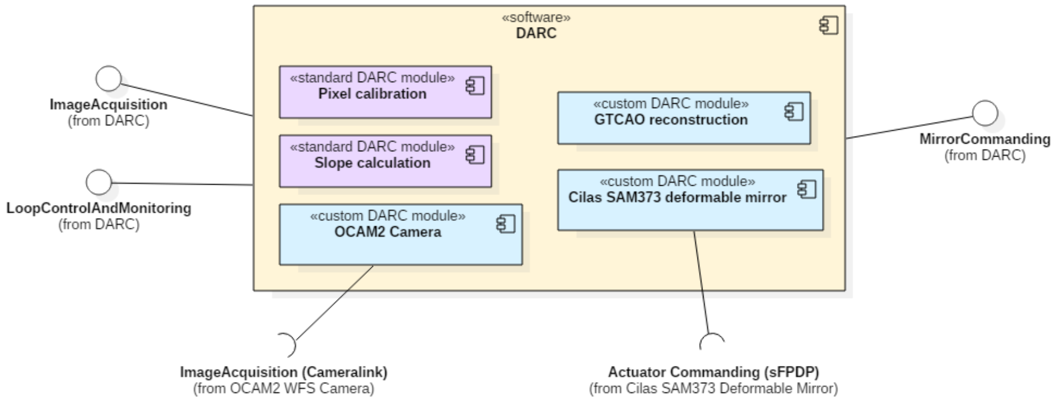

Figure 9. DARC subcomponents and interfaces view.

4. RTCS PROCESSES

This section presents the high level views of the main processes implemented by the RTCS.

4.1 Global process

The following figure shows the process followed in a typical RTCS working session.

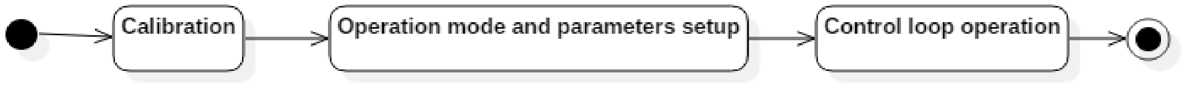

Figure 10. Typical RTCS working session.

After system startup, calibration procedures will be carried out if needed. After that, the operation mode and all the necessary parameters should be set. Once the setup is finished, the control loop operation can start. Loop operation will be open or close loop depending on the given observing modes and parameters. Also, M1 and M2 correction, MA guiding and supervisory loops activation will depend on the specific operation modes selected.

4.2 Calibration

Several types of calibrations are required for GTCAO which involve the RTCS. The RTCS is not responsible for implementing the different calibration procedures, although it must provide all the RT functionality needed in those procedures. The proposed calibration procedures will be implemented using the GCS Sequencer (see section 1.1) connected to the GTCAO software components.

4.3 Operation mode and parameters setup

A GUI, yet to be defined, will allow setting the operation mode and parameters for the control loop. This GUI will be developed with GTC's GUI development tools.

4.4 Control loop operation

The following figure summarizes the control loop operation.

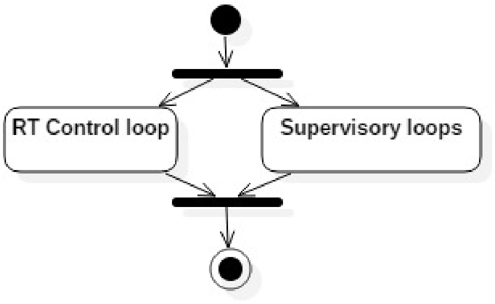

Figure 11. Control loop operation.

During control loop operation, both the RT control loop and the so called supervisory loops should run in parallel at different frequencies. Further detail for each subprocess is shown in the following subsections.

4.5 Real time control loop

The following diagram summarizes the high level steps followed during the RT control loop execution. A detailed description of the control loop functioning can be found in [19] for the general AO loop and in [20] for the wavefront reconstruction subprocess.

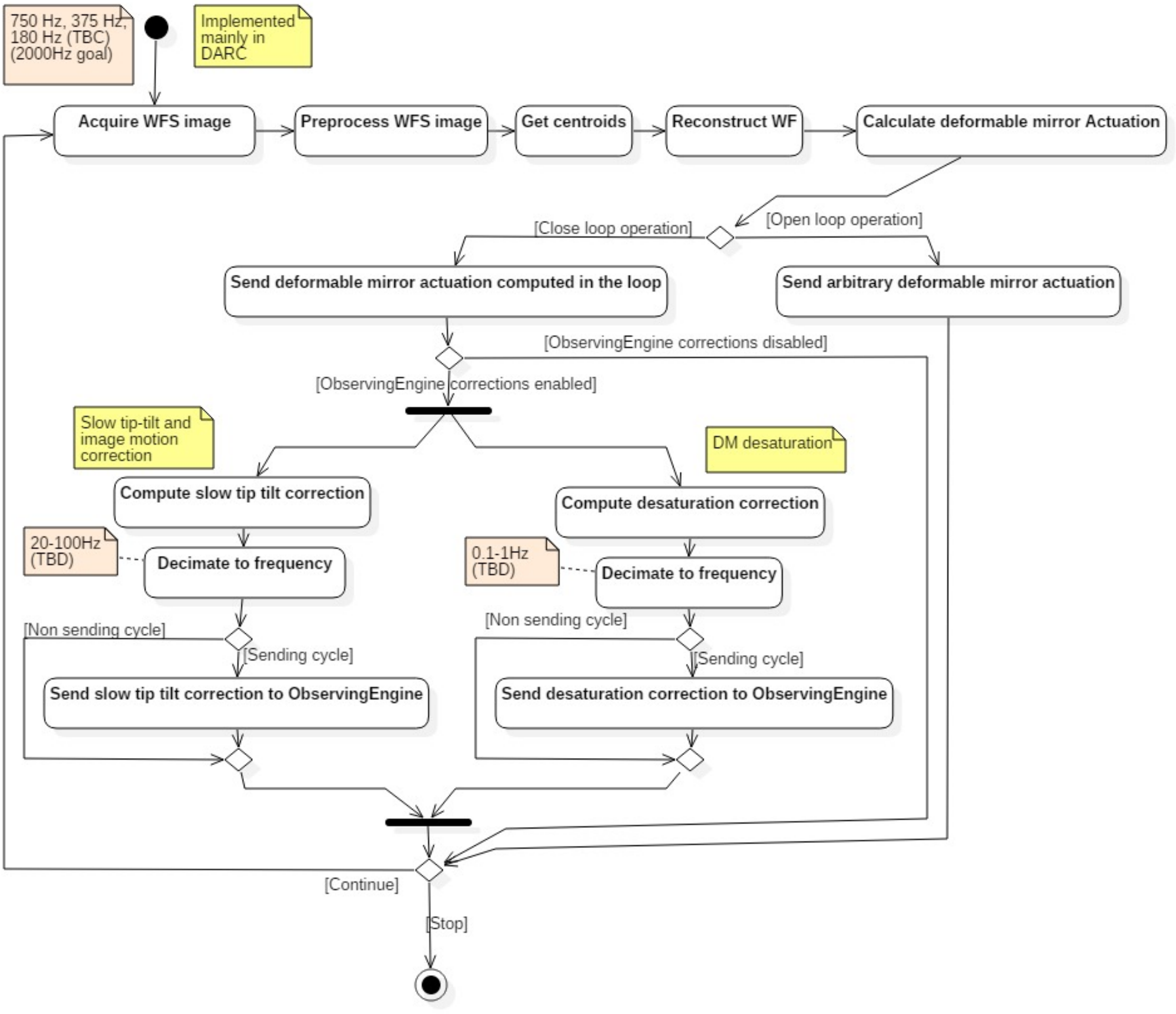
750 Hz, 375 Hz, 180 Hz (TBC) (2000Hz goal)
Implemented mainly in DARC
Acquire WFS image
Preprocess WFS image
Get centroids
Reconstruct WF
Calculate deformable mirror Actuation
[Close loop operation]
[Open loop operation]
Send deformable mirror actuation computed in the loop
Send arbitrary deformable mirror actuation
[ObservingEngine corrections disabled]
[ObservingEngine corrections enabled]
Slow tip-tilt and image motion correction
DM desaturation
Compute slow tip tilt correction
Compute desaturation correction
20-100Hz (TBD)
Decimate to frequency
0.1-1Hz (TBD)
Decimate to frequency
[Non sending cycle]
[Sending cycle]
[Non sending cycle]
[Sending cycle]
Send slow tip tilt correction to ObservingEngine
Send desaturation correction to ObservingEngine
[Continue]
[Stop]

Figure 12. Real time control loop process.

Each cycle of the AO control loop a set of computations is performed. These computations include the ones needed to generate a mirror actuation for the DM based on the image acquired from the SH WFS camera, and also the corrections needed for the telescope M1 and M2 mirrors, and for guiding (if guiding with the WFS is enabled). Depending on the type of loop operation, the calculated mirror actuation will be sent to the DM (close loop) or an arbitrary configuration will be sent to the DM (open loop). Open loop operation allows setting the DM flat or to an actuation that compensates non-common aberration paths [21], calculating actuator interaction matrixes or calculating the DM transfer function for example. Corrections for M1, M2 and MA should be computed over time and sent to the *ObservingEngine* at the appropriate rate. The main corrections to perform are slow tip-tilt correction and DM desaturation (see sect. 2).

4.6 Supervisory loops

As stated before, during control loop operation, both the RT control loop and the so called supervisory loops should run in parallel at different frequencies. The following figure summarizes the steps followed by the supervisory loops.

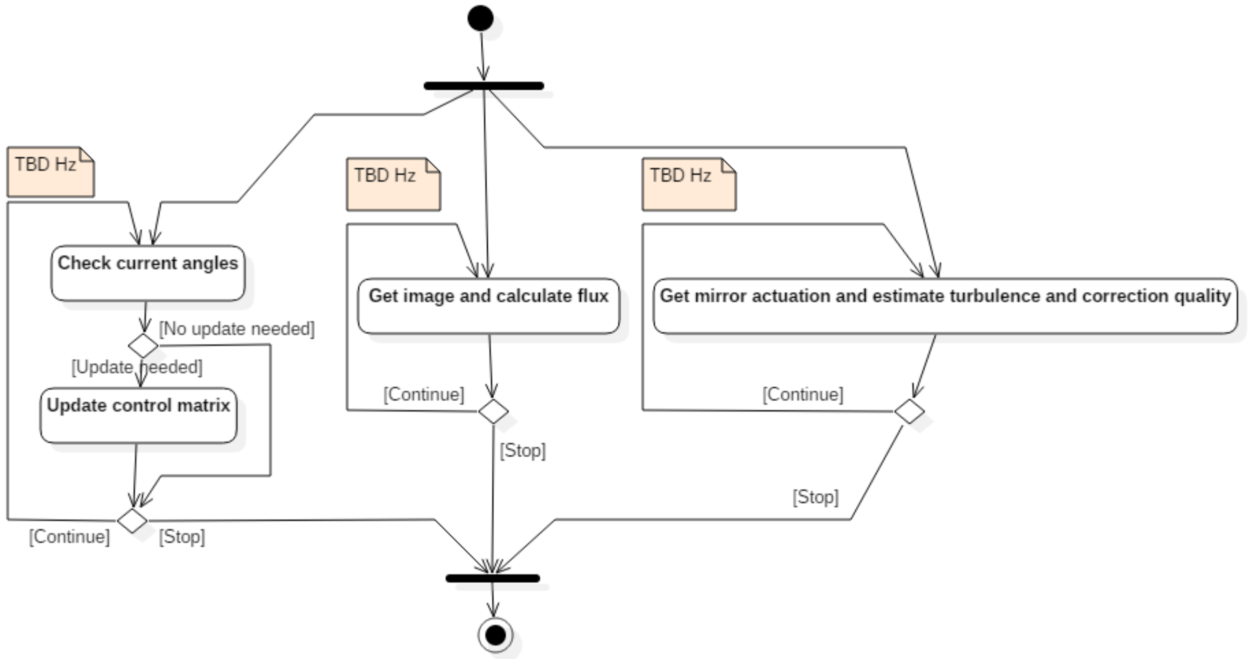

Figure 13. Supervisory loops process.

The tasks for each supervisory loop are:

- Update the control matrix depending on the alt/az position of the telescope.
- Calculate the flux for the subapertures in the WFS image.
- Estimate r_0 and coherence time (τ_0) [2] turbulence parameters, and image correction quality estimation.

5. RTCS DEPLOYMENT

The deployment of the main components regarding the RT control loop is conditioned by the high performance computing, low latency and low jitter required for correct functioning. All critical RT components must run in the physical host hosting the interface cards for the camera and the mirror as well as the high performance processors. Only the communications essential for the loop functioning should be performed by this host in order to isolate it from unexpected communication requests possibly affecting the system performance.

The software architecture and component deployment of the different GCS components related with the *ObservingEngine* need to be taken into account. In the same fashion as the *ObservingEngine* component has a class that manages high speed communication with M2, that is deployed directly in the M2 LCU, GTCAO will need another class or software component (probably part of the *ObservingEngine*) to manage high speed communication with M2 and directly deployed in the M2 LCU.

Figure 14 depicts the proposed deployment strategy for the components in the RTCS.

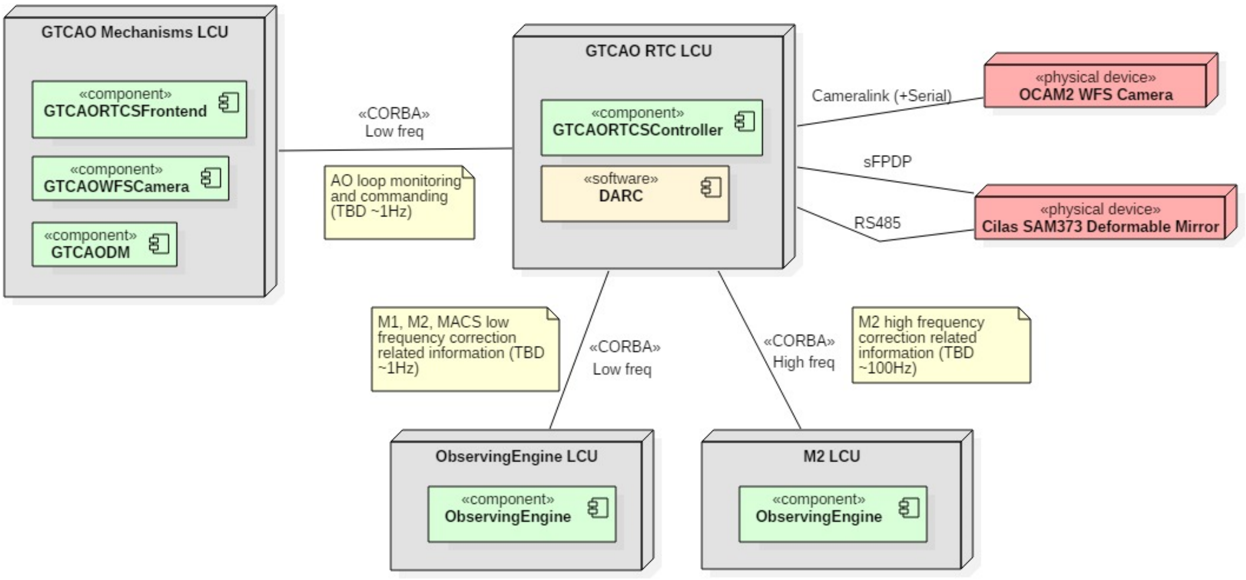

Figure 14. RTCS deployment view.

The proposed solution deploys the *GTCAORTCSFrontend* in the mechanisms LCU in order to relieve the RTC LCU of the processing power and communications required by this component. Communications involving the RTC LCU will be minimized in order to keep the RTC system jitter as stable as possible. CORBA interfacing has been selected as that is the interface used for communication with the *ObservingEngine*.

6. CONCLUSIONS AND FUTURE WORK

This work presented a feasible proposal for the GTCAO RTCS based on previous GTC instrument development experiences and successful software solutions like DARC. The concurrency of different control loops, both open and close, running at different frequencies and with different communication requirements, strongly constrains the system architecture and deployment. The presented design was ultimately driven by the need for high cycle frequencies and low jitter, the (consequential) need to minimize communications without hard real time constraints, and actual GCS physical implementation.

The immediate future work is the actual development of all the software described in this paper, which is already work in progress. Code compatible with the GCS and DARC modules are essential, but also multiple software tools for testing specific system requirements need to be developed and system configuration tasks need to be performed.

Future work not covered in this paper includes the GUIs design and development. Several GUIs will be needed to control and monitor the GTCAO RTCS. Both final and engineering GUIs will be needed and some of them will be developed in parallel with the rest of the RTCS software as they will serve as tools for testing and tuning.

ACKNOWLEDGEMENTS

This activity is funded by the Canary Islands Local Government, within the program "Canarias objetivo de progreso" promoted by the European Regional Development Fund of the European Union, operative program 2014-2020. It is pre-financed through a loan from the Spanish Ministry of Economy (State Secretary for Research).

REFERENCES

[1] J. A. López *et al.*, “FRIDA, the diffraction limited NIR imager and IFS for the GTC,” presented at the Highlights of Spanish Astrophysics VIII, 2015, pp. 29–40.

[2] “L10: Adaptive Optics.” [Online]. Available: http://slittlefair.staff.shef.ac.uk/teaching/phy217/lectures/telescopes/L10/. [Accessed: 22-May-2017].

[3] “Centre for Advanced Instrumentation : DARC - Durham University.” [Online]. Available: https://www.dur.ac.uk/cfai/projects/darc/. [Accessed: 22-May-2017].

[4] A. G. Basden and R. M. Myers, “The Durham adaptive optics real-time controller: capability and Extremely Large Telescope suitability,” *Mon. Not. R. Astron. Soc.*, vol. 424, no. 2, pp. 1483–1494, 2012.

[5] “Centre for Advanced Instrumentation : Durham AO Simulation Platform - Durham University.” [Online]. Available: https://www.dur.ac.uk/cfai/adaptiveoptics/dasp/. [Accessed: 23-May-2017].

[6] “First Light,” *First Light Imaging - EMCCD, e-APD, InGaAs Camera*. .

[7] J.-L. Gach, P. Balard, E. Stadler, C. Guillaume, and P. Feautrier, “OCAM2: world’s fastest and most sensitive camera system for advanced Adaptive Optics wavefront sensing,” in *Second International Conference on Adaptive Optics for Extremely Large Telescopes. Online at http://ao4elt2. lesia. obspm. fr, id. 44*, 2011.

[8] “Cilas. Deformable mirrors & adaptive optics,” *CILAS*, 24-May-2016. [Online]. Available: https://www.cilas.com/en/adaptive-mirrors. [Accessed: 24-May-2017].

[9] “Cilas Stack Array Deformable Mirrors (SAM) for Astronomy.” [Online]. Available: https://www.cilas.com/sites/default/files/public/media/panes/sam_deformable_mirror.pdf. [Accessed: 24-May-2017].

[10] “Cilas Deformable Mirrors Drive Electronics (DMDE).” [Online]. Available: https://www.cilas.com/sites/default/files/public/media/panes/digital_electronic_driver_dmde.pdf. [Accessed: 24-May-2017].

[11] “Point spread function,” *Wikipedia*. 29-Dec-2016.

[12] “Fried parameter,” *Wikipedia*. 05-Aug-2015.

[13] “Shack–Hartmann wavefront sensor,” *Wikipedia*. 21-Jan-2017.

[14] “Coherence time,” *Wikipedia*. 04-Nov-2016.

[15] B. R. Hunt, “Matrix formulation of the reconstruction of phase values from phase differences,” *JOSA*, vol. 69, no. 3, pp. 393–399, Mar. 1979.

[16] “Active optics,” *Wikipedia*. 09-Feb-2017.

[17] “Adaptive optics,” *Wikipedia*. 18-Apr-2017.

[18] P. J. Stomski Jr and J. C. Shelton, “Compensating for pupil rotation in the WM Keck Observatory adaptive optics system,” in *Astronomical Telescopes and Instrumentation*, 2000, pp. 608–619.

[19] “Download darc manual from SourceForge.net.” [Online]. Available: https://sourceforge.net/projects/darc2/files/man-1.2.0.pdf/download. [Accessed: 22-May-2017].

[20] M. Nuñez Cagigal *et al.*, “Feedback control baseline strategies for GTC adaptive optics with NGS,” in *Adaptive Optics for Extremely Large Telescopes 5–Conference Proceedings*, 2017, vol. 1.

[21] L. F. Rodríguez Ramos *et al.*, “Non-common Path Aberrations measurement using the NWIWM method,” in *Adaptive Optics for Extremely Large Telescopes 5–Conference Proceedings*, 2017, vol. 1.