PROCEEDINGS OF SPIE

The hardware control system for WEAVE at the William Herschel Telescope

Jose Miguel Delgado, Saúl Menéndez Mendoza, Jose Alonso Burgal, Jose Miguel Herreros, Sergio Picó, et al.

SPIE.

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 7/21/2018 Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

The hardware control system for WEAVE
at the William Herschel telescope

Jose Miguel Delgado[a], Saúl Menéndez Mendoza[b], Jose Alonso Burgal[a], Jose Miguel Herreros[a], Sergio Picó[c], Don Carlos Abrams[c], Carlos Martin[c], Diego Cano[c], F.J. Gribbin[c], Jure Skvarč[c], Kevin Dee[c], Emilie Lhome[c], Gavin Dalton[de], Kevin Middleton[d], Scott Trager[f], J. Alfonso L. Aguerri[a], Piercarlo Bonifacio[g], Antonella Vallenari[h], Esperanza Carrasco[i]

[a]Instituto de Astrofísica de Canarias, 38200 La Laguna, Tenerife, Spain; [b]ULL, University of La Laguna, 38200 San Cristóbal de La Laguna, Santa Cruz de Tenerife, Spain; [c]Isaac Newton Group, 38700 Santa Cruz de La Palma, Spain; [d]RALSpace, STFC Rutherford Appleton Laboratory, Harwell Oxford, OX11 0QX, UK; [e]Dept. of Physics, University of Oxford, Keble Road, Oxford, OX1 3RH, UK; [f]Kapteyn Institut, Rijksuniversiteit Groningen, Postbus 800, NL-9700 AV Groningen, Netherlands; [g]GEPI, Observatoire de Paris, Place Jules Janssen, 92195 Meudon, France; [h]Osservatorio Astronomico di Padova, INAF, Vicolo Osservatorio 5, 35122, Padova, Italy; [i]INAOE, Luis Enrique Erro 1, Tonantzintla, Puebla, Mexico;

ABSTRACT

When an alt-azimuth telescope is tracking at a specific field, it is necessary to use a de-rotator system to compensate the Earth's rotation of the field of view. In order, to keep the telescope tracking the field of view selected, the instrument will need to a rotation system for compensating it [1]. The new WEAVE [2] two degrees field of view requires a new field de-rotator on the top-end of the telescope. The rotator system has been designed with a direct drive motor which eliminates the need for mechanical transmission elements such as gearboxes, speed reducers, and worm gear drives. This design is a huge advantage for the system performance and lifetime because it eliminates undesirable characteristics such as long-time drift, elasticity, and backlash. The hardware control system has been developed with a Rockwell servo-drive and controller. The rotator has to be controlled by the high-level software which is also responsible for the telescope control. This paper summarizes the model developed for simulating and the software which will be used to accept the rotator system. A performance study is also carried out to test the CIP (Common Industrial Protocol) for communications between the high-level software and the rotator hardware.

Keywords: WEAVE, control system, rotator, system modeling, CIP, Ethernet/IP.
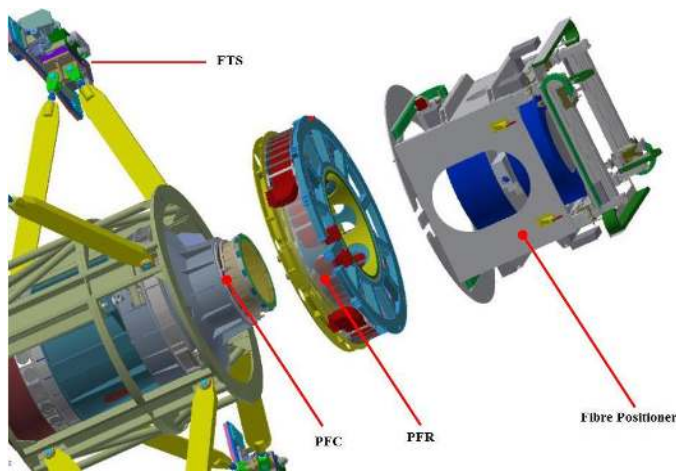
1. INTRODUCTION

Figure 1:WEAVE prime focus assembly: Focus translation (FTS) , Focus Corrector (PFC), Rotator (PFR), and Positioner on WHT

WEAVE is the next-generation wide-field optical spectroscopy facility for the 4.2m William Herschel Telescope (WHT) in La Palma, Canary Islands, Spain. WEAVE is a multi-object and multi-IFU facility, collecting light via the new 2-degree prime focus of WHT. The prime focus is being re-fitted with a new corrector and tumbler unit with a robotic pick-and-place fibre positioner. The light from either 1000 fibres, 20 mini-IFUs or a single large IFU is transported to the GHRIL enclosure on the Nasmyth platform, which hosts a dual arm, multi-mode spectrograph that will provide full coverage of the majority of the visible spectrum in a single exposure at a spectral resolution of ~5000 or modest wavelength coverage in both arms at a resolution ~20000. In order to compensate the field rotation, WEAVE Fibre Positioner must be assembled on a Prime Focus Rotator (PFR) on the WHT. [3]

The PFR must provide a rotation degree of freedom for the Fibre Positioner with a high bending stiffness. The PFR positions the Fibre Positioner to an accuracy of 5 arcsec when tracking and guides all the fibres and other power and control lines through a cable wrap, for which the available space is limited. IDOM, a Spanish company is manufacturing the PFR system. They have completed the final design and have started the manufacturing phase. [4]

This paper summarizes the system model of a de-rotator and the performance using the Common Industrial Protocol (CIP) for communications between the high-level software and the rotator hardware. The main motivation of this work is the need to dispose of a software simulator of the rotator system. This simulator will allow to test, check, and verify the communication interface and its performance as well as to test the functionality and verify the rotator hardware requirements. For this, it is necessary to have all the components of the assembly, from the communication in high level as with the hardware of the rotator.

This is the reason why a mathematical model of the PFR and its control has been generated. In this way, without having the hardware, which is in production, you can develop the control interfaces and check the performance in communication. This mathematical model can be implemented as a reference model for high-level adaptive control. This control is carried out in a generic way with a PI control, since having a model of the system could be implemented with adaptive control, such as a Self-Tuning Regulator (STR) or a Model Reference Adaptive Controller (MRAC).

The model itself could be used for the future selection of components for projects of the same nature, allowing to see the behaviour of different components before their acquisition. In this way the selection of components for the project is improved.

The possibilities of communication and studies are wide. Since you can check each of the interfaces in three systems: one of them being the definitive PFR, another one being a hardware prototype with a small motor and the other one a software simulator as close as possible to the behaviour of the PFR. In this way, the risk of damaging the final hardware is reduced, since it has previously passed two checks before being implemented on it.

The system will have a first communication and performance test with the mathematical model. The second test will be with the engine prototype. And finally, it will be checked with the real system, an ETEL's motor.

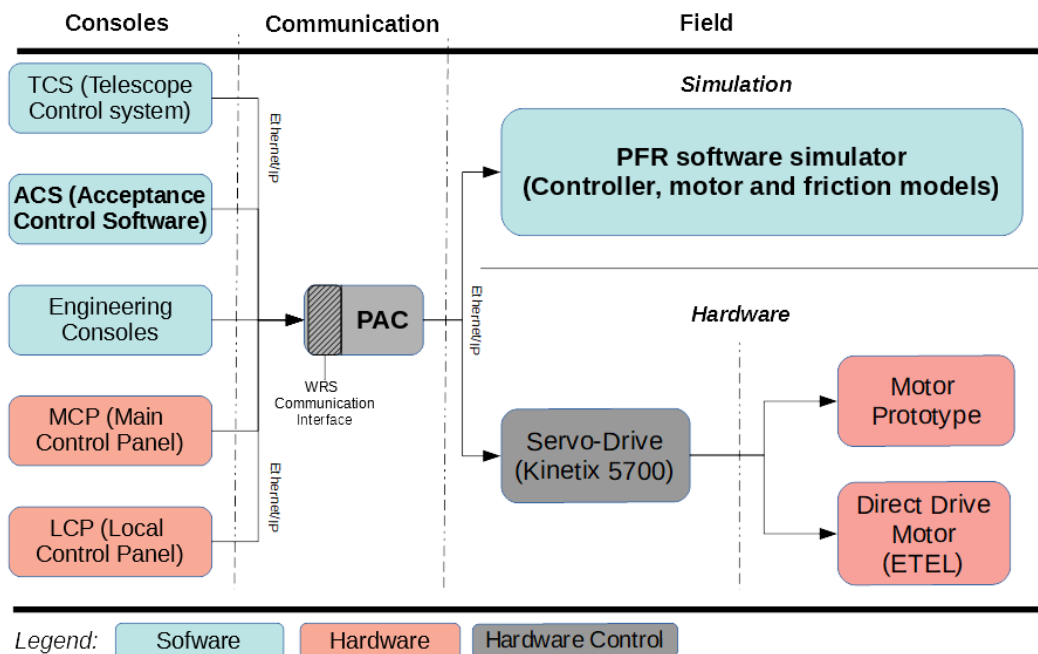Figure 2: Communication diagram between the system components

Telescope Control System (TCS): It is the High Level Software. The TCS will command the demanded velocity to the PFR via CIP [5]. It will responsible of tracking a target sky field and therefore keeping the target into the position limits required by WEAVE. It will be able of closing a position control loop with the information received from the auto-guider

system. This control loop is also called the 'external loop' or the 'high-level loop'. The High-level software is a dedicated component running inside a computer which will be a point of access to the Low-Level sofware, centralizing demands and publishing mechanism information.

The Acceptance Control Software (ACS): It will be used for the development, verification and acceptance phases. It will emulate the TCS behaviors. Its interface is identical to the one with the TCS. It is added a simulator of the control panels to reproduce their functionality.

Engineering Consoles: to provide an independent and complete access to the instrument mechanisms using a different path than the one used by the standard command interface of the observing system; it will provide an interface with the same look and feel of other instruments developed by the Isaac Newton Group (ING), and not require any previous knowledge about the command syntaxes. There are two engineering consoles: the final WEAVE engineering console and the rotator console.

Local Control Panel (LCP), used for manual movement of the prime focus rotator; it will show the rotator position, display information about interlocks, alarms and limits, the motor current and the operational mode; it also will have an acknowledge button for silencing the alarms.

Main Control Panel (MCP), situated in the control room; it will consist of a touch-sensitive display and a number of physical buttons, including the emergency stop buttons and a key for switching between the manual and computer control. Its interface is identical to the one with the LCP.

Programmable Automation Controller (PAC): The PFR controller is an Allen Bradley 1756-L73 LOGIX5573 controller with 8Mb RAM. The embedded control system takes the form of a PAC microprocessor system using similar technologies to those that have been successfully employed at the ING in the recent past. This control system will be responsible for performing the actions requested under the constraints of reliability and time. It will also be responsible for updating the internal data structures with an accurate snapshot of the instrument state. In practice, it will not be aware of which external component is requesting a demand. The PAC will provide a general way to accept demands using a very simple and effective approach; allowing a subset of global variables and tag names, which will send instructions to the PAC, to be written by an external component on demand. The control interface is defined in this controller through many structures tags which correspond with the data required for controlling and retrieving the feedback information.

Figure 3: Allen-Bradley ControlLogix controller

Figure 4: Kinetix 5700 Allen-Bradley servo-drive

Servo-Drive: An Allen-Bradley's Kinetix 5700 Ethernet/IP servo-drive is used to control the motor. The Kinetix servo drives implement an acceleration/torque loop, which is nested within a velocity PI control loop, which is nested within an outer position PI control loop. This type of control structure includes these advantages: Precise control of position, velocity, and torque; Ability to switch between position, velocity, and torque modes without changing tuning gains; and manual tuning.

Motor prototype: It is a hardware simulator which is done with a small motor for testing purposes. It is developing by Idom and it will allow verifying the PFR functionality as well as the communication interface.

Direct-drive motor: It is the final motor used by the PFR. It is an ETEL TMB0990-050-3VDN-S82-NN manufactured by ETEL S.A. Switzerland. It is equipped with three integrated temperature sensors for safety purposes; a pneumatic brake system; limit and end-stop switches; for accurate rotator position readings, a full-circle tape encoder for internal mounting with reference points and three reading heads are used. The selected products are Heidenhain MSB ERA 7400C tape, and Heidenhain AK ERA 7480 reading heads. Three Hall effect sensors are foreseen for determining the initial absolute rotator position and motor initialization.

PFR software simulator: It is a software created with Labview which integrates the controller, motor and friction models of the PFR running in a real-time (computer clock). It can simulate the velocity, current, torque, position outputs, limits, break, temperature and typical failures.

Ethernet/IP [2]: The connection between the TCS/ACS and the PFR uses a CIP model for communications over TCP/UDP/IP, called Ethernet/IP (EIP). The CIP is a communication protocol for transferring automation data between two devices. In the CIP Protocol, every network device represents itself as a series of objects. Each object is simply a grouping of the related data values in a device. This allows programmers do not worry about the communications details and guarantee the deterministic time in the data interchange.

Figure 5: ETEL TMB0990-050 torque motor

2. ROTATOR SOFTWARE SIMULATOR

The model was based on a previous work [6]. In the beginning, a Simulink model was created and tested. The next step was the model implementation, in Labview for running the simulation in real time. Labview control & simulation tools have been used to generate each control loop. The communication interface was incorporated in the controller using the NI Ethernet / IP module. An acceptance software client has also developed which will simulate the control panels, telescope control, and user configured demands.

2.1 Motor model

The motor selected to drive the rotator has been an ETEL TMB0990-050 model, providing sufficient torque at all rotation speeds and telescope elevation angles and to maintain tracking accuracy. The baseline design uses passive air cooling. This motor construction enables an exceptional peak force density in the magnetic gap, as well as unparalleled thermal efficiency [7].

The important parameters of the motor are derived from the specifications and the model is a standard DC motor representation, including the motor electrical time constant. The mathematical model of the ETEL motor has electrical characteristics, such as the voltage supply to the winding (This voltage is provided by the speed control), the current flowing through the rotor (armature current), and the resistance of the rotor winding and the inductance of the stator winding.

PERFORMANCE		UNIT	3VDN	3VHS
	Winding codes		FREE AIR CONVECTION	
Tp	Peak torque	Nm	6650	6650
Tc	Continuous torque	Nm	1710	1710
Ts	Stall torque	Nm	1300	1300
Kt	Torque constant	Nm/Arms	82.5	41.2
Ku	Back EMF constant (*)	Vrms/(rad/s)	47.7	23.9
Km	Motor constant	Nm/√W	51.7	51.7
R20	Electrical resistance at 20°C (*)	Ohm	1.70	0.425
L1	Electrical inductance (*)	mH	19.7	4.92
Ip	Peak current	Arms	154	307
Ic	Continuous current	Arms	21.8	43.5
Is	Stall current	Arms	16.5	33.0
Pc	Max. continuous power dissipation	W	1730	1730

SPECIFICATIONS		UNIT		
Udc	Nominal input voltage	VDC	600	600
τth	Thermal time constant	s	4050	4050
Rth	Thermal resistance	K/W	0.0637	0.0637
2p	Number of poles	-	176	176
J	Rotor inertia	kg.m²	5.01	5.01
Mr	Rotor mass	kg	25.1	25.1
Ms	Stator mass	kg	111	111
Td	Max. detent torque (average to peak)	Nm	56	56
ns	Stall speed	rpm	0.0017	0.0017
Δθw	Water temperature difference for Pc	K	N/A	N/A
qw	Minimum water flow for Δθw	l/min	N/A	N/A
Δpw	Max. pressure drop at qw	bar	N/A	N/A

Figure 6: ETEL TMB0990-050 motor performance and specifications

This pair is influenced by that produced by dynamic friction. The mechanical time constant is realized with the motor inertia and the external friction subsystem [8].

The prototype created in MATLAB is shown in Figure 7, which as indicated before having two inputs, the first with the voltage supplied to the motor and the second with the torque resulting from the dynamic friction model. The outputs of the system will be the speed of rotation of the motor and the current supplied.
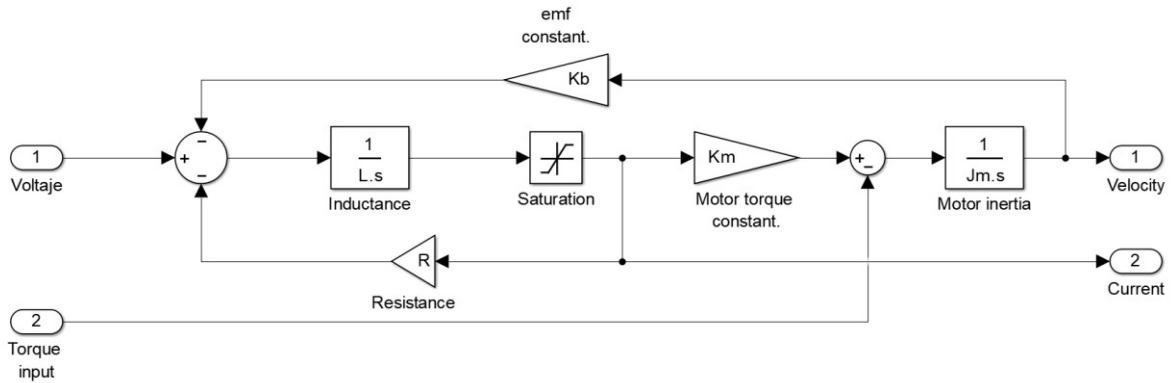
Figure 7: ETEL motor simplified model in Simulink

Once the correct functioning of the model has been verified, the design has been carried out in LabView, allowing a real-computer time running. The result is shown in Figure 8.
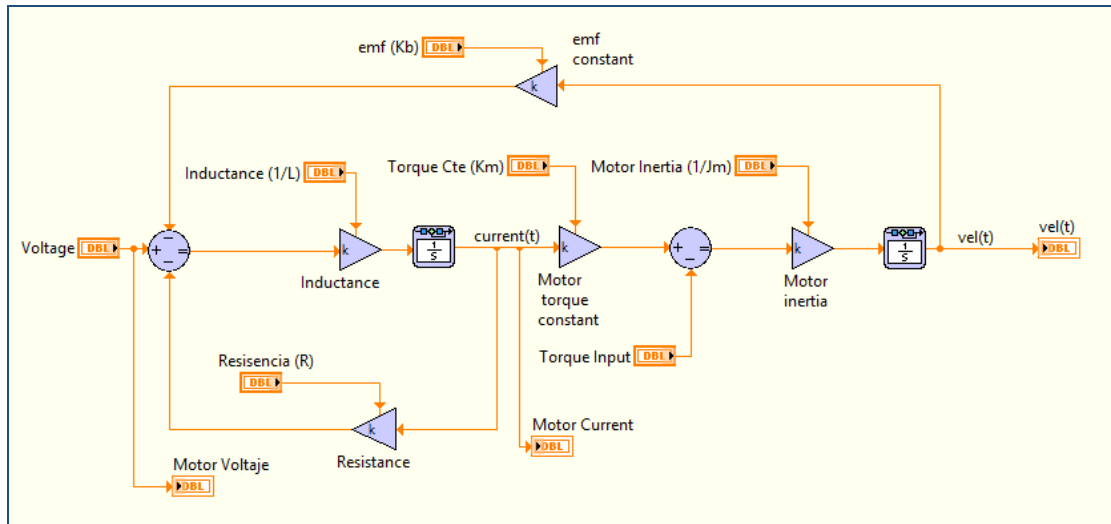

Figure 8: ETEL motor model in Labview

2.2 Controller model

The large variation in setpoint speeds makes necessary to generate an adaptive controller for correct behavior. The Kinetix 5700 has been modeled with an implementation of a gain scheduling controller. This type of control allows the adjustment of the parameters of the controller in the function of the working ranges. The working range corresponds to the setpoints speeds of the PFR.
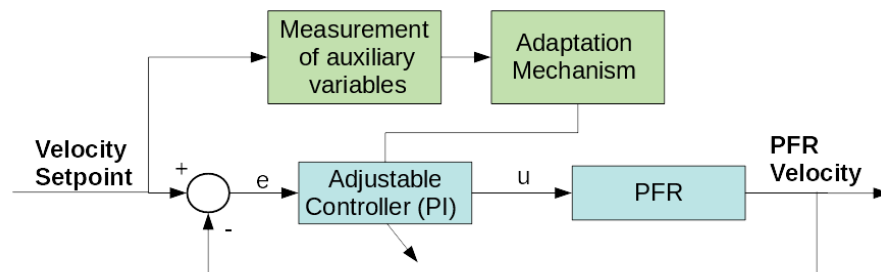

Figure 9: Gain scheduling diagram blocks

The values of the gains only are suitable in the simulation with the mechanical and electric parameters of the PFR, if a change of plant is needed then the gains must be modified until the PI controller can work optimally. This kind of control requires certain time to obtain a good performance of the system.

2.3 Dynamic friction model

Friction is a phenomenon that reduces the efficiency of the mechanical elements and, in turn, hampers proper control of these through automated systems. Therefore, a thorough knowledge of this property can allow the design of more precise control systems.

The objective of this section is the modeling of the friction component of the system. The LuGre model has been used to model the dynamic friction of the PFR. First of all, is necessary to define and explain the different friction action involves in the model.

Coulomb friction: Also known as dry friction, it is defined as the phenomenon of friction that appears when two surfaces interact without the presence of fluid between them.

It is characterized by being directly proportional to the natural force (F_N) between the surfaces in contact and independently of the relative speed between them.

$$F_f = \mu \cdot F_N$$

- F_f = Friction force (N)
- F_N = Normal force (N)
- μ = Friction coefficient (Adimensional)

Viscous friction: Most sliding contacts are lubricated. This explains that dry friction does not faithfully represent the contact phenomenon between surfaces. In this type of friction is characterized by being independent of the normal force between the surfaces and proportional to the relative speed between these, although recent studies show that this friction is dependent on normal force, since the experiments performed on this model with variation of the lubricant pressure have shown that there is a certain relationship.

$$F_f = \gamma \cdot v$$

- F_f = Friction force (N).

- v = The relative velocity between surfaces (m/s).

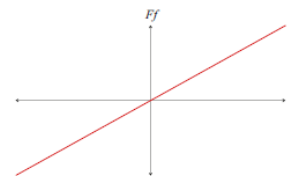- γ = Viscous friction coefficient ($\frac{N}{m/s}$).

Figure 10: Viscous friction [9]

Static friction: When the relative speed between the surfaces is zero, the friction force is bigger than the dynamic friction. This is due to the inertia of the system and to the adhesion produced between surfaces. The coefficient of static friction is assigned the Greek letter "mu" (μ), with a subscript "s". The maximum force of static friction is μs times the normal force on an object. (Figure 11)

Stribeck effect: When analyzing the friction in lubricated contacts it can be observed that, in the beginning, the force decreases with increasing speed until it reaches a minimum value. After this point, the friction begins to increase behaving as indicated by the law of viscous friction.

This phenomenon is called the Stribeck effect, named after Richard Stribeck, the first person to develop a model capable of explaining it. (Figure 12)
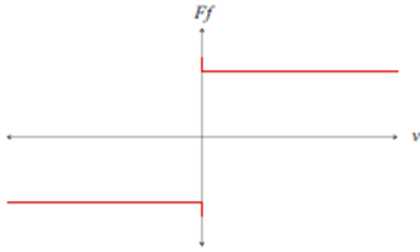
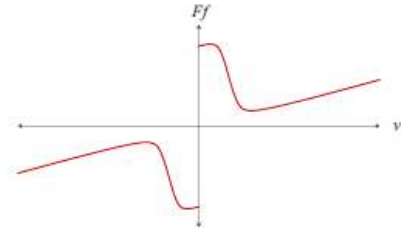Figure 11: Static friction and Coulomb friction [9]

Figure 12: Stribeck effect [9]

Figure 13 represents the coefficient of friction (μ) as a function of the specific film thickness parameter (λ) or the dimensionless velocity. Where η is the dynamic viscosity of the fluid, N is the entrainment speed of the fluid and P is the normal load.
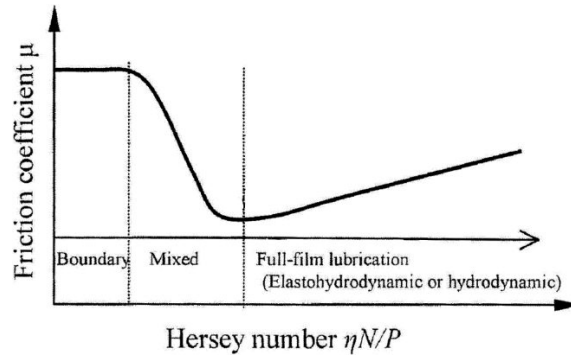

Figure 13: Stribeck curve [9]

Dahl model: It is a static model, which does not depend on the relative speed, it only considers the relative displacement between the surfaces. The friction force is model as a function of displacement. The model only explains the behavior prior to sliding according to the deformation of the surface roughness.

LuGre model: The characterization of friction is a complex task since a high knowledge of the system is required. At present we have several friction models, which make a faithful representation of the phenomenon. This model is named after the universities of Lund and Grenoble, with whose collaboration it could be developed. Add viscous friction and the Stribeck effect to the Dahl model. It is, therefore, a dynamic model that also considers, not only speed but also acceleration [9].

This model with constant speeds, the friction curve ends up behaving as the Coulomb model predicts. The following equations explain The LuGre model:

$$F = \sigma_0 z + \sigma_1 \dot{z} + f(v)$$

$$\frac{dz}{dt} = v - \sigma_0 \frac{|v|}{g(v)} z = v - h(v)z$$

$$g(v) = F_c + (F_s - F_c)e^{-\left(\frac{v}{v_r}\right)^{\alpha}}$$

σ_0 = Hardness for an elastic behaviour at small displacements.

z = Internal friction state, position.

$g(v)$ = Function dependent on speed.

σ_1 = Additional cushioning associated with micro-displacements (viscous friction).

F_s = Static friction

F_c = Coulomb friction.

v_r = Relative speed in which the system behaves like dry friction.

$f(v)$ = Velocity dependent term. For viscous friction $f(v) = \sigma_2 v$.

The friction in permanent regime is the following:

$$F_f(t)|_{rp} = \sigma_0 \cdot g\big(\omega\,(t)\big) \cdot sgn\big(\omega\,(t)\big) + \sigma_2\,\omega\,(t)$$

For the description at low speeds:

$$g\big(\omega\,(t)\big) = F_c + (F_s - F_c)e^{-(\frac{v(t)}{v_r})^\alpha}$$

Dynamic friction model implementation: The dynamic friction subsystem is based on the LuGre friction model. This does not incorporate hysteretic pre-sliding friction but is relatively straightforward to model and does incorporate the Stribeck friction feature that is responsible for stick-slip phenomenon. The static, Coulomb and viscous friction parameters of the model are easily measured or calculated but the parameters defining the internal state of the model, like the Stribeck velocity, the micro-scale stiffness σ0, and its associated damping σ1 are not. They can be identified by precision tests carried out on the actual component but that is not practical for this simulation. Instead, the parameters are estimated based on tests carried out on similar systems. For example, bearings will have a micro-scale stiffness of between 1×10^6 and 1×10^8 depending on the specific materials and pre-load. The associated damping is derived from this assuming a typical damping factor.

The Stribeck velocity is perhaps the hardest to estimate since it also depends on the lubrication. A typical value for a 40mm mean diameter bearing is 5 x 10-3 rad/s. This is a tangential velocity at the balls of about 0.1mm/s which, when divided by 30 (typically the actual slip velocity at these interfaces is about 1/30th of the component velocity) gives about 3 μm/s. Typical slip velocities for lubricated rolling are between 1 μm/s and 10 μm/s. This relationship is used to estimate the Stribeck velocity at other bearing radio [8].

The model created in MATLAB has only one input, the PFR velocity. The output of the subsystem is a torque. This torque is introduced in the engine model subsystem producing the real PFR rotatory velocity. Once the correct functioning of the model has been verified, the design has been carried out in LabView, allowing a real-time study.

Figure 14: Dynamic friction subsystem in LabView

3. COMMUNICATION INTERFACE
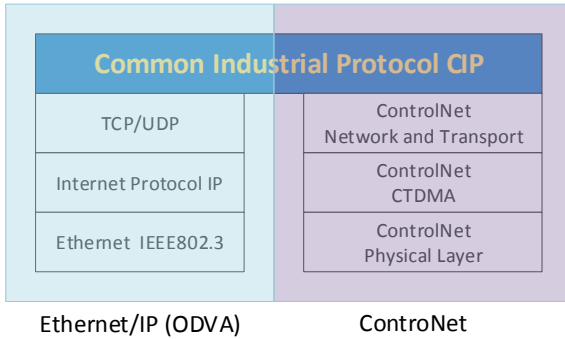
3.1 The Common Industrial Protocol (CIP)

Figure 15: CIP over TCP/UDP/IP and over ControlNet

CIP is a media independent, connection-based, object-oriented protocol designed for automation applications. The CIP is led by ODVA. [10].

CIP provides a set of services for automation applications: control, safety, synchronization, motion, configuration, and monitoring. Every network device contains many information elements which can be read or written directly through its CIP address. Ethernet/IP is the implementation of the CIP over standard Ethernet and the TCP/IP protocol. CIP applies a common application layer over an Ethernet network by encapsulating messages in TCP/UDP/IP. This common application layer provides interoperability and interchangeability of industrial automation and control modules on an Ethernet network.
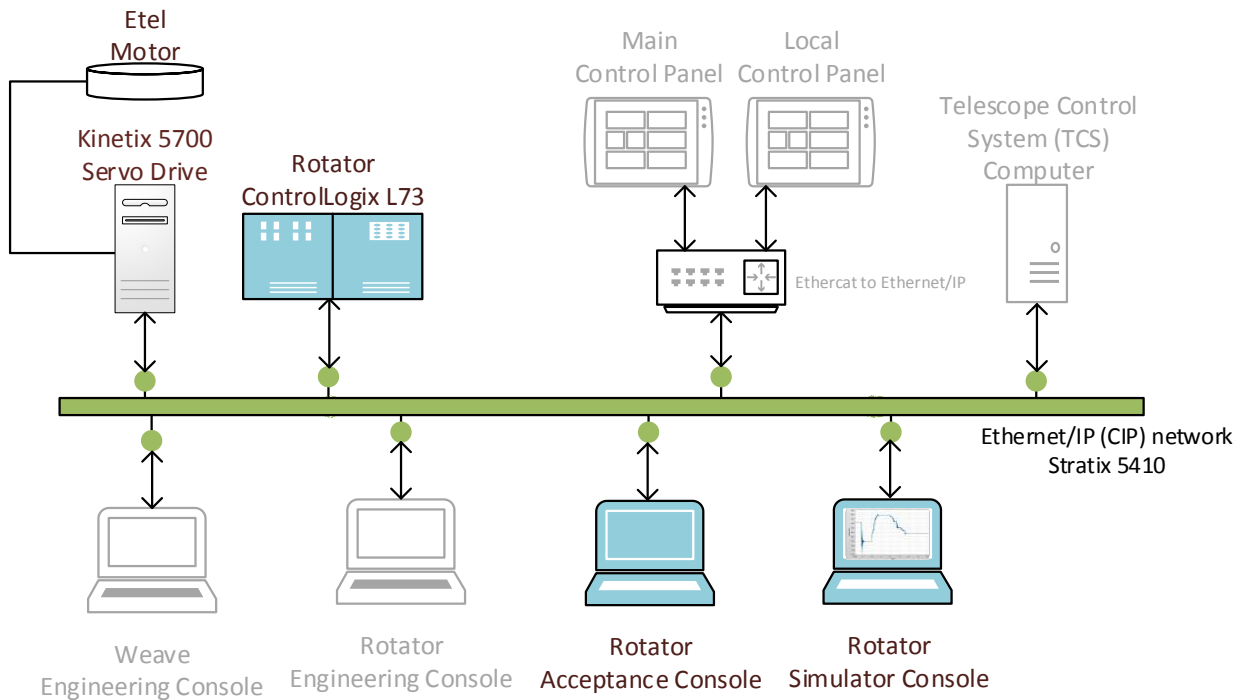
Figure 16: WRS communication bus (Ethernet/IP)

The Ethernet/IP (CIP) bus communication technology is chosen to be used in WEAVE with the Focus, Atmospheric Corrector, Rotator, Spectrograph and Calibration Unit control hardware. The development process will be carried out using the Allen Bradley 1756 family ControlLogix controller. Although a common Ethernet switch can be used to hold the CIP network, an Ethernet/IP 20 ports Stratix 5410 switch has been selected as the main element of switching to keep the communication performance.

The National Instrument Labview Ethernet/IP module has been used to give the CIP capabilities to the Rotator Engineering Consoles, Rotator Acceptance Console and Rotator Simulator Console which are the components developed under the study of this paper.

The EtherNet/IP (EIP) [5] network supports both time-critical (implicit) and non-time critical (explicit) message transfer services of CIP. Exchange of time-critical messages is based on the producer/consumer.

Implicit messages are time critical in nature. This includes I/O and produced/consumed tags model where a transmitting device produces data on the network and many receiving devices can consume this data simultaneously. Implicit refers to information (source address, data type, and destination address) that is implied in the message but not contained in the message. The implicit messages are used commonly with Real-time I/O data, Functional safety data or Motion control data. Implicit messages use the User Datagram Protocol (UDP) and can be unicast or multicast. Each message is transferred on a time basis (cyclic trigger) or Requested Packet Interval (RPI). The RPI is the update rate specified for a particular piece of data on the network.

Explicit connections are non-time critical and are request/reply in nature. Explicit refers to basic information (such as source address, data type, or destination address) that is included in every message. Explicit messages use Transmission Control Protocol (TCP). Explicit messages are used for point-to-point, client-server transactions. To send explicit messages to a Logix system, connected messaging is recommended instead of unconnected messaging. Unconnected messaging resources are limited and can become overwhelmed with activity. When this happens, response times become very long and the user application can suffer as a result.

3.2 Communication interface

The communication interface between the rotator system and the external client (TCS or ACS) is defined as several different structure tags inside the ControlLogix rotator controller. This interface will be used by all the PFR clients. The simulator, rotator engineering console, and acceptance control software are using the Labview user-defined controls which correspond with the same tag structures defined into the Controllogix. (Figure 17)
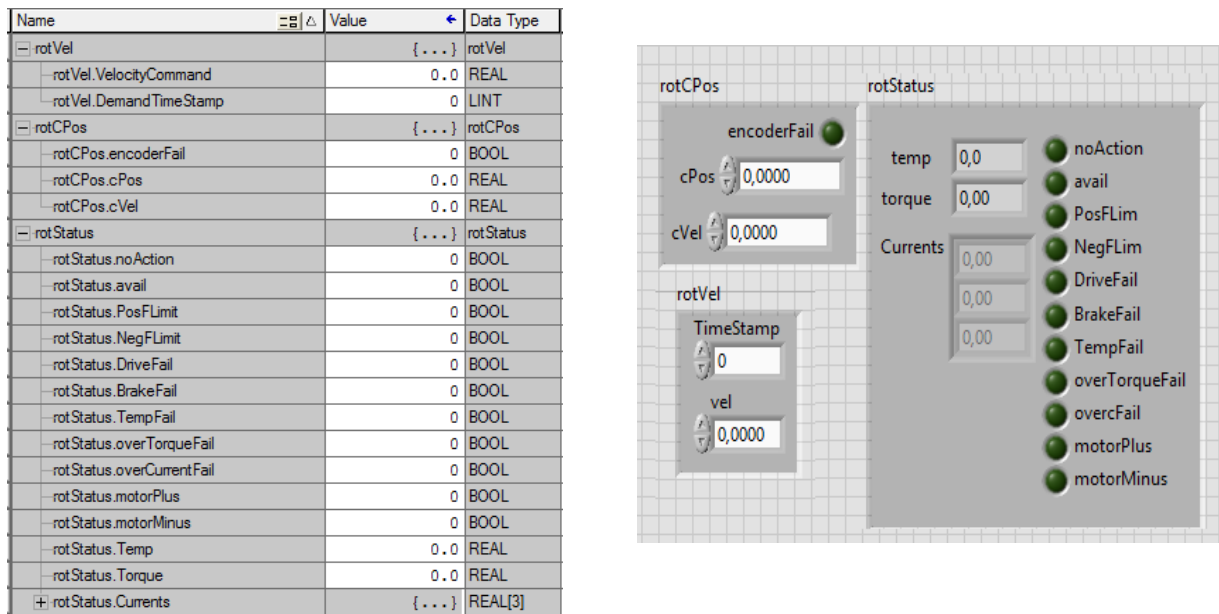

Figure 17: Communication Interface: Tag structures in Studio 5000 and Labview clusters

 A set of Labview functions have been implemented to easily communicates with the rotator controller. The simulator software is also using this communication interface for replacing the rotator hardware. Only the controller IP address and the program name where the tags are created are needed to write and read the structures defined by the communication interface.

4. ACCEPTANCE CONTROL SOFTWARE

The ACS will act the role of the TCS, the High-Level software [11], requesting actions to the PAC or Low-Level software. This software is developed in Labview and is using the communication interface described above.

The purpose of this layer is: to coordinate and synchronize PFR PAC actions: Simulation, running tests, command a position, command a tracking velocity, homing, stopping and so forth; to transfer status information from the PAC to the ACS monitor.

It can send to the rotator or to the simulator many type of demands: steps, trapezoidal, sin signal. It also can simulate the telescope position and tracking sending the rotator velocity demand. It is able of closing a simple control loop in position and commanded in velocity. A small interference signal can be applied to this case.
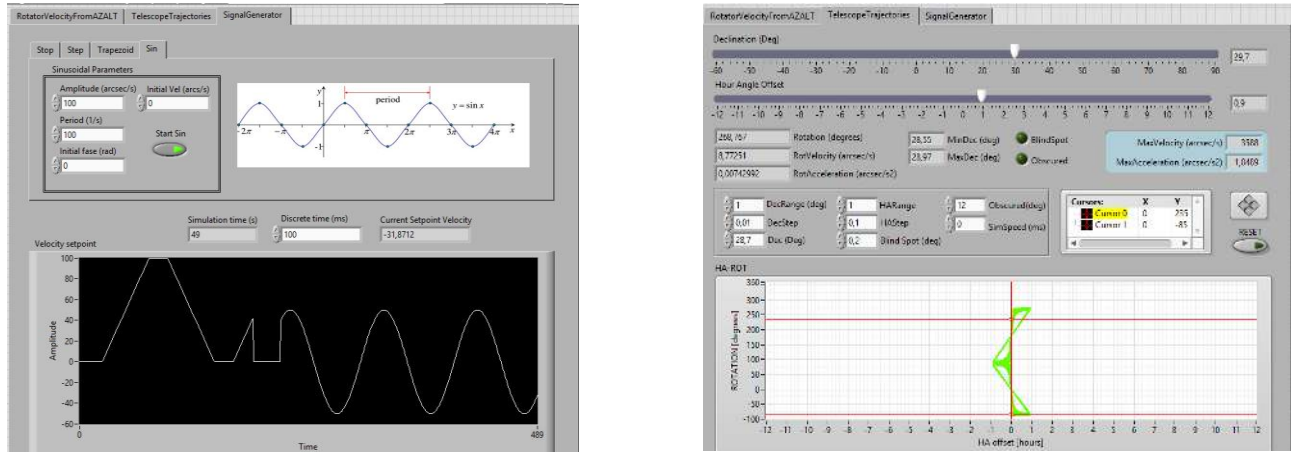

Figure 18: Acceptance software, telescope trajectories and velocity demand generator

5. RESULTS

5.1 Software Simulator

The simulator integrates the three models described before; the logic which implements the limits, brakes, and temperature model; and the Labview panel to be managed by the user (Figure 19). The model parameters can be configured on the tabs. The velocity input demand is available from the PAC communication interface. The simulator is running continuously, calculating the rotation position, current, torque, temperature, and output velocity. If a limit is reached then the specific limit switch simulation is activated. Limit switches ensure that no movement beyond specified limits is allowed. If actuated, the movement of the rotator shall be stopped and the rotator should be removed from the area beyond limits through the process described in the design. The user is able of simulating many different failures which can be occurred in the real-hardware system.
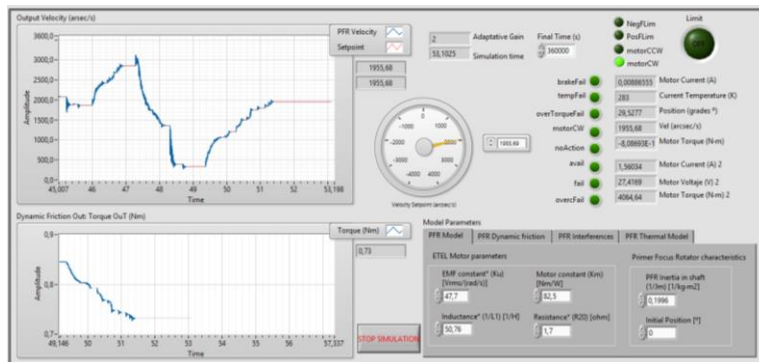

Figure 19: PFR software simulator

The simulator has been tested with different configurations, simulating different motors, and loads. The results have been as they expected; however, the simulator should be configured and tested with parameters closer to the final hardware. It is planned to be done when the first hardware tests are performed.

5.2 Communication tests

Several communication tests were carried out to check both types of messages. An office common IP network was used for the testing, therefore better performance results are expected using a Stratix 5410 switch which is designed to deal with EIP communication.

For the implicit communication test, a consumed and a produced tags were configured at the Controllogix as well as at the Labview test. A 50 ms loop was created in the Labview side and an RPI of 1 ms was configured. The delay time and jitter were between reasonable values. (Figure 20 and Figure 21)

Explicit messages were also tested similarly. The Labview side, in this case, was requesting periodically (each 50 ms) some tags to the controller. (Figure 22)
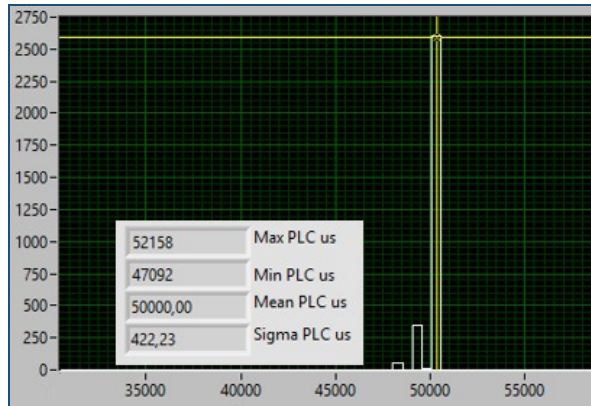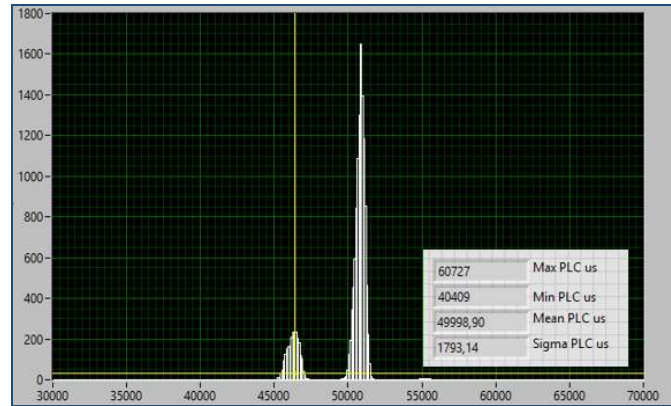

Figure 20: Implicit messages test result

Figure 21: Explicit messages test resut

Communications delay between a client information requests to the controller has also been measured. The time since the request is sent until the information reach back to the client is around 5 ms ± 3 ms.
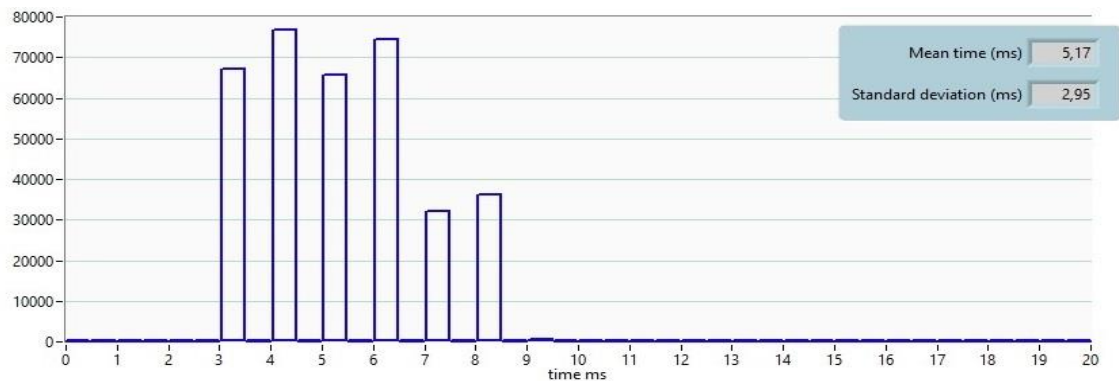

Figure 22: Explicit request time

In our case, the explicit connection was chosen due to the low-frequency communication required by the TCS or external position loop that is 20Hz. However, synchronization between the position feedbacks requested by the client with the external control loop is required.

6. CONCLUSION

The work presented in this paper will be beneficial for testing the final TCS software and the final prime focus rotator on WEAVE. It also will allow substituting the hardware for testing at the integration phase as well as at the operational phase.

The EIP communication is has been verified and it seems to be work with slow control loops around 20Hz to 50Hz.

This work has been can be used for future adaptive control system developing The IAC is involved in other similar projects where this work can be applied just by changing the configuration parameters. The model presented can be adapted to any de-rotator system and can be integrated easily with other types of interface communication.

REFERENCES

[1] C.Y. Tan, "An Analysis Of Field De-rotation For Alt-Az Mounted Telescopes" (August 2015).

[2] The Isaac Newton Group of Telescopes (ING), "WEAVE Project website", <https://www.ing.iac.es/weave> (2018).

[3] Gavin B. Dalton, "Construction progress of WEAVE: the next generation wide-field spectroscopy facility for the William Herschel telescope", SPIE paper 10702-47 (June 2018).

[4] IDOM, "The new Prime Focus Rotator for the William Herschel telescope (WHT)", SPIE Paper 10700-235 (June 2018).

[5] Allen-Bradley Rockwell Automation, "Ethernet/IP Design Considerations – Reference Manual", Publication ENET-RM002C-EN-P (May 2013).

[6] Martin Fisher-Fisher Astronomical Systems Engineering, "New WHT Prime Focus Rotator for Weave" (January 2013)

[7] ETEL, "TMB_data_sheets_TMB0990-050", <https://www.etel.ch/> (July 2015)

[8] Martin Fisher-Fisher Astronomical Systems Engineering, "Modeling the WHT PF and WEAVE Rotator" (January 2013).

[9] Astor García Amor, "Fricción, una Propiedad Limitante en el Movimiento Controlado", Universidad de Oviedo. Spain < http://digibuo.uniovi.es> (16 August 2016).

[10] John Rinaldi, "An Overview of Ethernet/IP", Real Time Automation < www.rtaautomation.com> (2003).

[11] Sergio Picó, "The WEAVE observatory control system" SPIE Paper 10704-83 (June 2018).