MOONS Fibre Positioner Control and Path Planning Software

Steven Beard, Bart Willemse, Stephen Watson, David Atkinson, Pablo Gutierrez Cheetham, et al.

Published version information:

Citation: SM Beard et al. MOONS fibre positioner control and path planning software. Proc SPIE 12189 (2022): 121890V. Is in proceedings of: Software and Cyberinfrastructure for Astronomy VII, Montréal, Québec, Canada, 17-23 Jul 2022

DOI: 10.1117/12.2629239

PROCEEDINGS OF SPIE

SPIEDigitalLibrary.org/conference-proceedings-of-spie

MOONS fibre positioner control and path planning software

Steven Beard, Bart Willemse, Stephen Watson, David Atkinson, Pablo Gutierrez Cheetham, et al.

Steven Beard, Bart Willemse, Stephen Watson, David Atkinson, Pablo Gutierrez Cheetham, Paolo Franzetti, Johannes Nix, "MOONS fibre positioner control and path planning software," Proc. SPIE 12189, Software and Cyberinfrastructure for Astronomy VII, 121890V (29 August 2022); doi: 10.1117/12.2629239

SPIE.

Event: SPIE Astronomical Telescopes + Instrumentation, 2022, Montréal, Québec, Canada

MOONS Fibre Positioner Control and Path Planning Software

Steven Beard[a], Bart Willemse[a], Stephen Watson[a], David Atkinson[a], Pablo Gutierrez Cheetham[b], Paolo Franzetti[c], Johannes Nix[d]
[a] STFC, UK Astronomy Technology Centre, Royal Observatory Edinburgh, Edinburgh, EH9 3HJ, UK; [b] ESO, Karl-Schwarzschild-Str. 2, 85748, Garching bei München, Germany; [c] INAF, Viale del Parco Mellini, Rome, Italy; [d] Deutsches Zentrum für Luft-und Raumfahrt, Institut für Robotik und Mechatronik, Münchener Strasse 20, 82234 Weßling, Germany

ABSTRACT

The MOONS multi-object spectrograph relies on an array of 1000 fibre positioners to acquire targets in the focal plane. The fibre positioners have a larger overlap than similar instruments because MOONS can observe in the infrared. The large overlap gives MOONS the ability to acquire close pairs of object and sky targets, but it makes moving positioners to their targets without a collision even more technically challenging. We describe how the MOONS fibre positioner control system overcomes those challenges with custom electronics to manage the synchronisation between the positioners, a collision protection system, and a grid driver software system which manages the control of the fibre positioners. We also describe our experiments with different path planning algorithms and present the latest results from MOONS testing.

Keywords: Multi-object Spectrograph, Fibre positioners, Path planning, Software

1. INTRODUCTION

MOONS [1] is a multi-object, fibre-fed spectrograph for the European Southern Observatory (ESO) Very Large Telescope (VLT). It uses a hexagonal grid of 1000 2-axis positioners to place fibres on targets at the telescope focal plane. The fibres channel light from those targets to a pair of 3-channel spectrographs, which receive 500 fibres each.

1.1 MOONS Operating Modes

MOONS observes targets at infrared wavelengths, which means that accurate sky subtraction is important. There are two main operating modes to support sky subtraction, as shown in Figure 1.

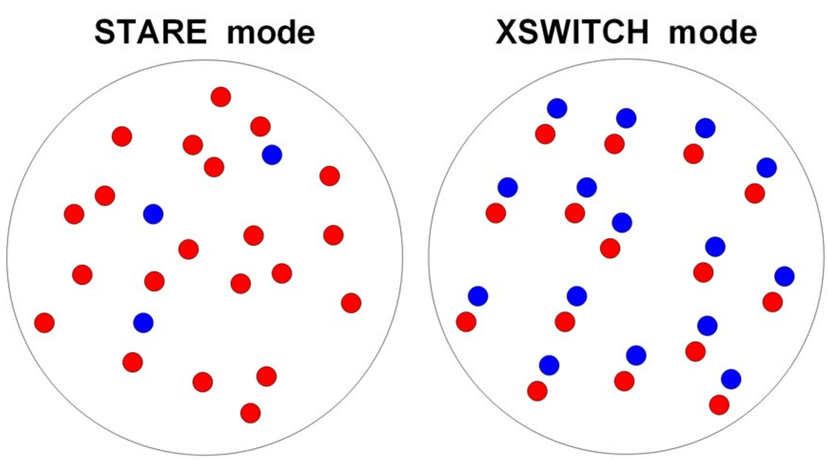


Figure 1. MOONS has two main operating modes: STARE mode, where most fibres are placed on a target (red) and a few fibres (blue) sample the sky background; and XSWITCH mode, where each target fibre (red) is paired with a sky fibre (blue). Fibre pairs are typically placed between 10 and 30 arcseconds apart.

Software and Cyberinfrastructure for Astronomy VII, edited by Jorge Ibsen,
Gianluca Chiozzi, Proc. of SPIE Vol. 12189, 121890V · © 2022 SPIE
0277-786X · doi: 10.1117/12.2629239

Fibres dedicated to astronomical targets are shown in red, and fibres dedicated to sky are shown in blue. In STARE mode, sky subtraction can be enhanced by nodding the telescope to increase the number of sky samples (STARE-NOD mode). XSWITCH mode provides the most rigorous sky subtraction, where the telescope can be nodded to place the targets alternately on the red and blue fibres and the sky background sampled alternately on each side of the target. The XSWITCH mode requires any location on the sky to be reachable by at least two fibre positioners. The requirement means the MOONS fibre positioners have a higher degree of overlap than instruments with similar fibre positioner systems (such as DESI [8]) which operate at visible wavelengths.

1.2 MOONS Observation Planning

MOONS Observations must be planned in advance and described in executable "Observation Blocks", just like observations made with other ESO/VLT instruments. It is important to know in advance which of the selected targets is going to be observed. Observations can also be scheduled in any order (as the exact order depends on scheduling constraints and weather conditions). For this reason, we always start a MOONS observation by moving the fibre positioners from a fixed starting location, as shown in Figure 2. The fibre positioner control software has the task of moving the fibre positioner safely from this starting location to the targets and then reversing back to the starting location.

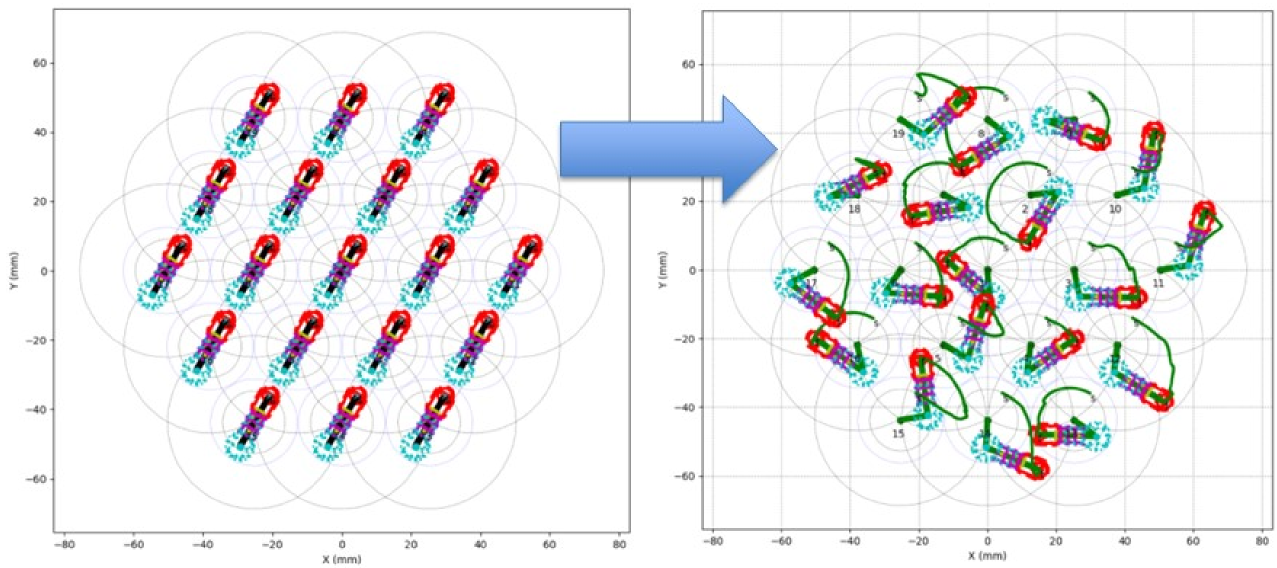


Figure 2. The problem to be solved. How to move 1000 overlapping fibre positioner safely from fixed starting locations to targets and back again.

1.3 Fibre Positioner Control Requirements

Besides needing to move from their starting point to targets without a collision, fibre positioners must also move to their targets within a time limit of about 1 minute, so it makes sense to move them all at the same time. To achieve these requirements, the positioners need a path planning algorithm capable of steering them all simultaneously around a crowd of moving obstacles, each obtacle a neighbouring fibre positioner. The algorithm cannot work without knowing the location these neighbours at each iteration, which means fibre positioner movements must be accurately synchronised. The path planning algorithm maintains a safety tolerance between fibre positioners of 0.5mm on the MOONS focal plane while they are moving. A positioner moving at maximum speed can move through this 0.5mm tolerance in about 5 milliseconds, so it is important for fibre positioner movements to be synchronized to a few milliseconds.

Although the fibre positioner software must avoid collisions, the fibre positioners themselves must also be resilient to faults such as collisions and power failures. The software must also be able to cope with broken positioners by treating them as stationary obstacles.

1.4 Fibre positioner location feedback

MOONS has a camera-based metrology system capable of providing feedback on the location of the fibre positioners, which is described in [3]. The metrology system must be capable of detecting all the fibre positioners and determining their locations and orientations unambiguously to within the same positioning tolerance as the positioners themselves.

2. FIBRE POSITIONER SYSTEM DESIGN

2.1 Mechanical Design

MOONS fibre positioners have a two-armed design, as described in [2]. Each positioner can swivel around a central (alpha) axis. A beta arm 17mm in length is attached to the end of the 8mm long alpha arm. Their alpha and beta motors work together to allow a target to be reached within a doughnut-shaped patrol zone 9-25mm from their centre, as shown in Figure 3. Each fibre positioner has a local patrol area 9-12.5mm from its centre. Positioners are spaced by 25mm, so each one can reach the centre of its neighbours. A fibre positioner is at risk of a collision if it moves outside its local patrol area (more than 12.5mm from its centre), or if any of its 6 neighbours have moved outside their local patrol area. Fibre positioners are safe from collisions when they are all within their local patrol area (see Figure 9). Since every fibre positioner has an identical design, fibre positioners are also safe from collisions if they all make identical movements. This feature can be used for calibration.

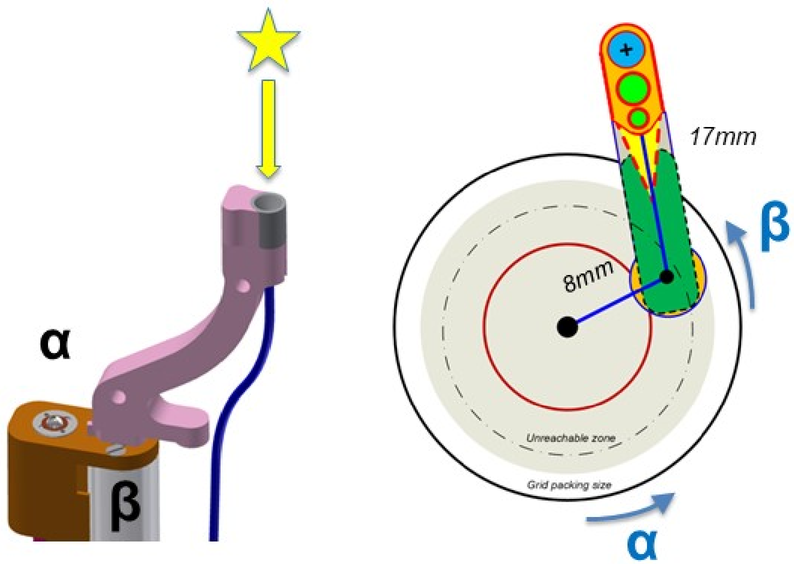


Figure 3. 3-D and 2-D models of a MOONS fibre positioner (left and right), showing the alpha and beta axes. Together, the arms allow the positioner to reach a target within a doughnut-shaped patrol field with inner diameter 17-8 = 9mm and outer diameter 17+8 = 25mm.

Fibre positioners can collide when similar zones on two neighbouring positioners overlap in 3-D space. The avoidance zones tracked by the software are shown in Figure 4. The software detects a collision when any of these similar zones overlap. For simplicity, the curved beta arm is modelled in the 2-D simulation as a set of cuboids.

Every new set of fibre positioner targets is first analysed by conflict checking software to ensure that none of these avoidance zones will overlap when the positioners are on their targets. The software also determines which beta arm parity setting (left-armed or right-armed) places the avoidance zones furthest from nearby obstacles. The MOONS observation preparation software (Moonlight) uses this conflict checking software to determine which combination of targets will be the most easily achieved in one observation.

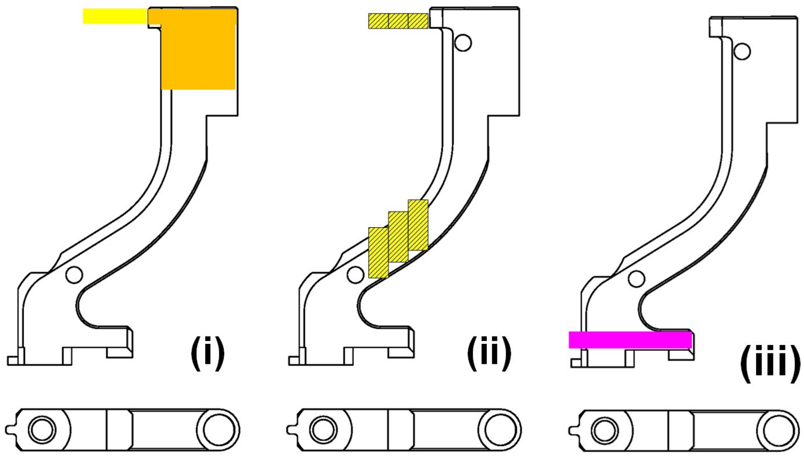


Figure 4. Avoidance zones on each positioner as modelled by the software. Each zone can collide with a similar zone on a neighbouring positioner when they overlap. (i) The fibre holder and metrology target area (orange) plus an optional "fibre snag" zone (yellow). (ii) The beta arm, represented in the 2-D model by 3 cuboids at different levels (hatched orange). (iii) The beta datum actuator (magenta).

2.2 Electronics Design

The electronics design is determined by three main requirements: the requirement to communicate with many fibre positioner units; the requirement to synchronize the movements of all those fibre positioner units; and the requirement for the fibre positioners to be controllable by ESO-standard software.

Figure 5 shows the communication architecture we chose to meet those requirements. CANBus is used because it is a simple and robust industry standard used to communicate with large number of embedded devices. However, communicating with 1000 fibre positioners requires them to be connected using multiple CAN buses, due to electrical drive limitations. It would be difficult to achieve the synchronisation requirement if those CANBuses were connected separately to the Linux workstation running the ESO software. We solved the requirement by using custom made "EtherCAN" cards [4].

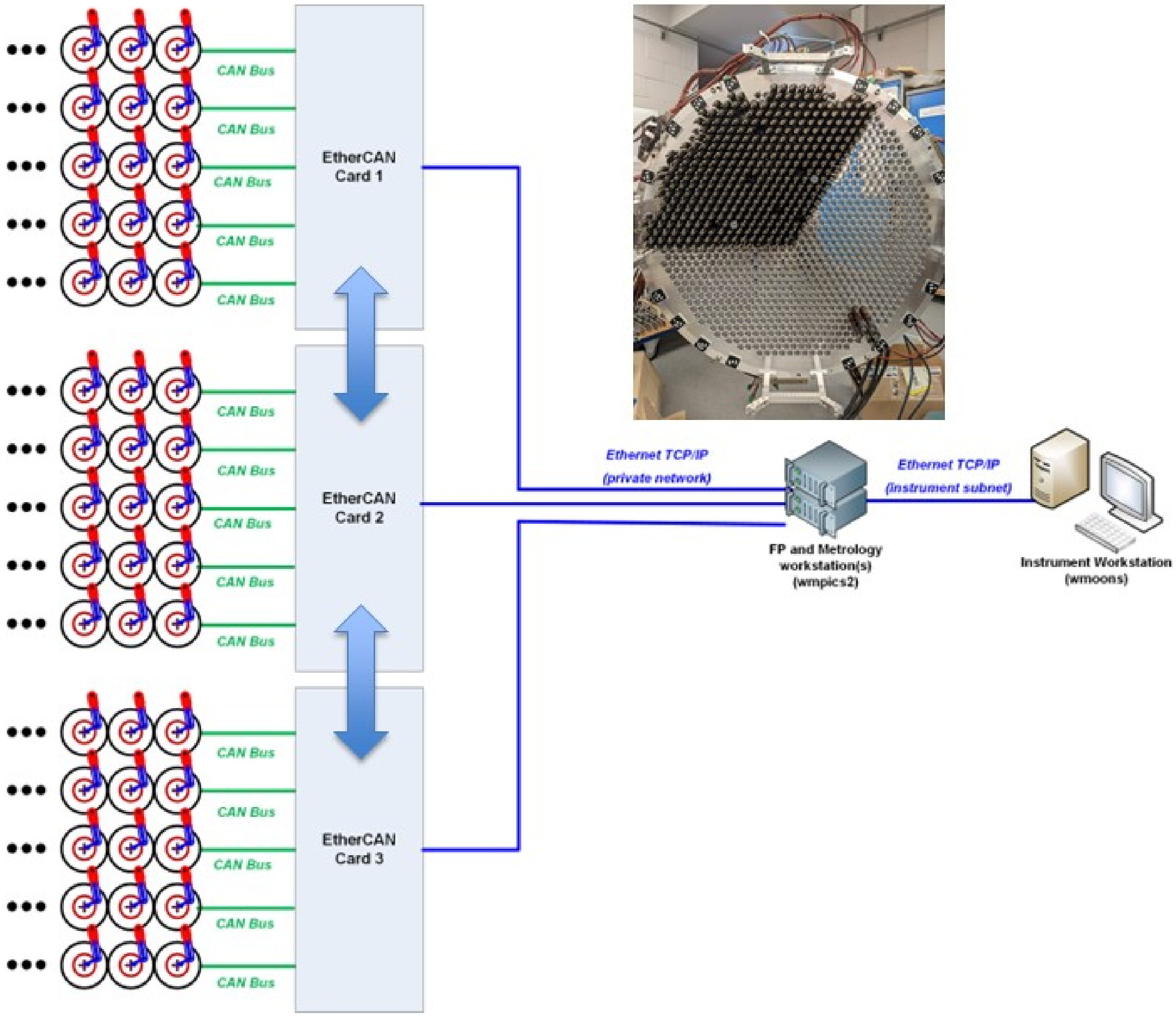


Figure 5. MOONS Fibre positioner communication architecture. We achieve synchronisation by using custom made "EtherCAN" control cards. Each card controls fibre positioners in 1/3 of the focal plane (as shown in the inset).

Each EtherCAN card can receive fibre positioner commands over a TCP/IP connection and forward those commands to the appropriate CANBus and CAN address. Commands designed to be broadcast to all fibre positioners can be sent synchronously across all the CANBuses by means of dedicated high-speed digital I/O lines connected between the EtherCAN cards. The MOONS fibre positioners are controlled using 3 EtherCAN gateway cards.

- The firmware installed on each EtherCAN gateway can send the movement trigger CAN commands to its 5 CAN buses with a timing accuracy of 1-2 milliseconds
- Synchronised triggering between the 3 EtherCAN gateways is achieved by the "master" gateway sending an electrical trigger pulse to the two "slave" gateways.

The fibre positioner arms are controlled using stepper motors and are located by counting steps from a reference point, which is defined for both arms by detecting the edge of a datum switch. In addition, the alpha arm has a limit switch which prevents the arm from over-twisting the fibre.

The electronics includes a collision detection system, where each positioner is connected to one of 7 voltages, as shown in Figure 6. The layout is designed so that every positioner is surrounded by neighbours at different voltages. If the conductive parts of any two positioners touch a current flows, a collision event is communicated to the software, and the motor power is cut. This protects each positioner from collision damage. The collision detection system also detects when a beta arm goes beyond its limits.

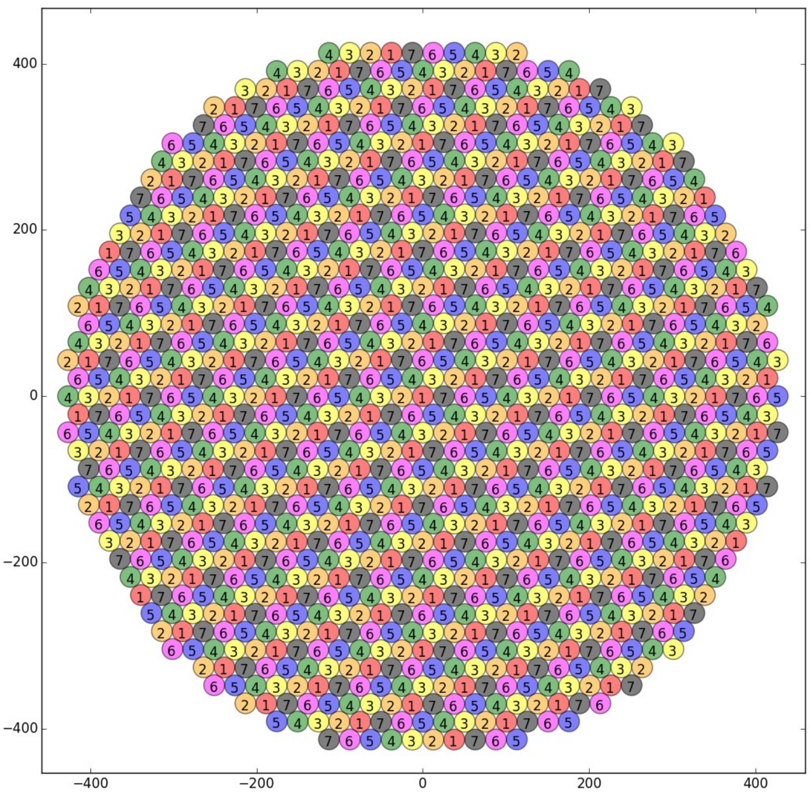


Figure 6. Collision protection voltages.

2.3 Firmware and Low-level Software Design

Each fibre positioner unit (FPU) is managed by a microcontroller which controls its motors through embedded firmware, using commands shown in Table 1. The firmware maintains the internal state of each FPU and monitors the CANBus for commands. Each motor can be programmed to make complex movements by downloading a "waveform" which defines a step count to be achieved in each 125ms time step. A waveform can be divided into a maximum of 256 of these time steps (limited by the memory available in the firmware). Waveforms can contain a simple ramp-up, cruise, ramp-down instruction to make a simple movement, or can describe the movements necessary to move the positioner through a complex path.

Table 1. Main commands recognised by the MOONS fibre positioners.

Command	Description	Synchronised?
findDatum	Define the reference point by locating the datum switch.	Yes
configMotion	Download a waveform which pre-programs a movement	
executeMotion	Start the motion described by the downloaded waveform	Yes
reverseMotion	Instructs positioners to prepare to execute the last waveform in reverse	
repeatMotion	Instructs positioner to prepare to repeat the last movement	
abortMotion	Instructions all positioner to stop moving immediately	Yes

At the Linux side of the communication link, low-level control of the fibre positioners is achieved through “grid driver” software, which can direct commands to specific fibre positioners or broadcast a synchronisation request to all fibre positioners. In addition to the commands described in Table 1, there are also commands which help recover the positioners after a limit breach, collision, or abort.

Each fibre positioner goes through the state changes shown in Figure 7. During normal operation the positioners will be moving between the “RESTING”, “LOADING”, “READY” and “MOVING” states. The states “OBSTACLE_ERROR” and “ABORTED” are used to manage fault conditions. Broken positioners may be locked.

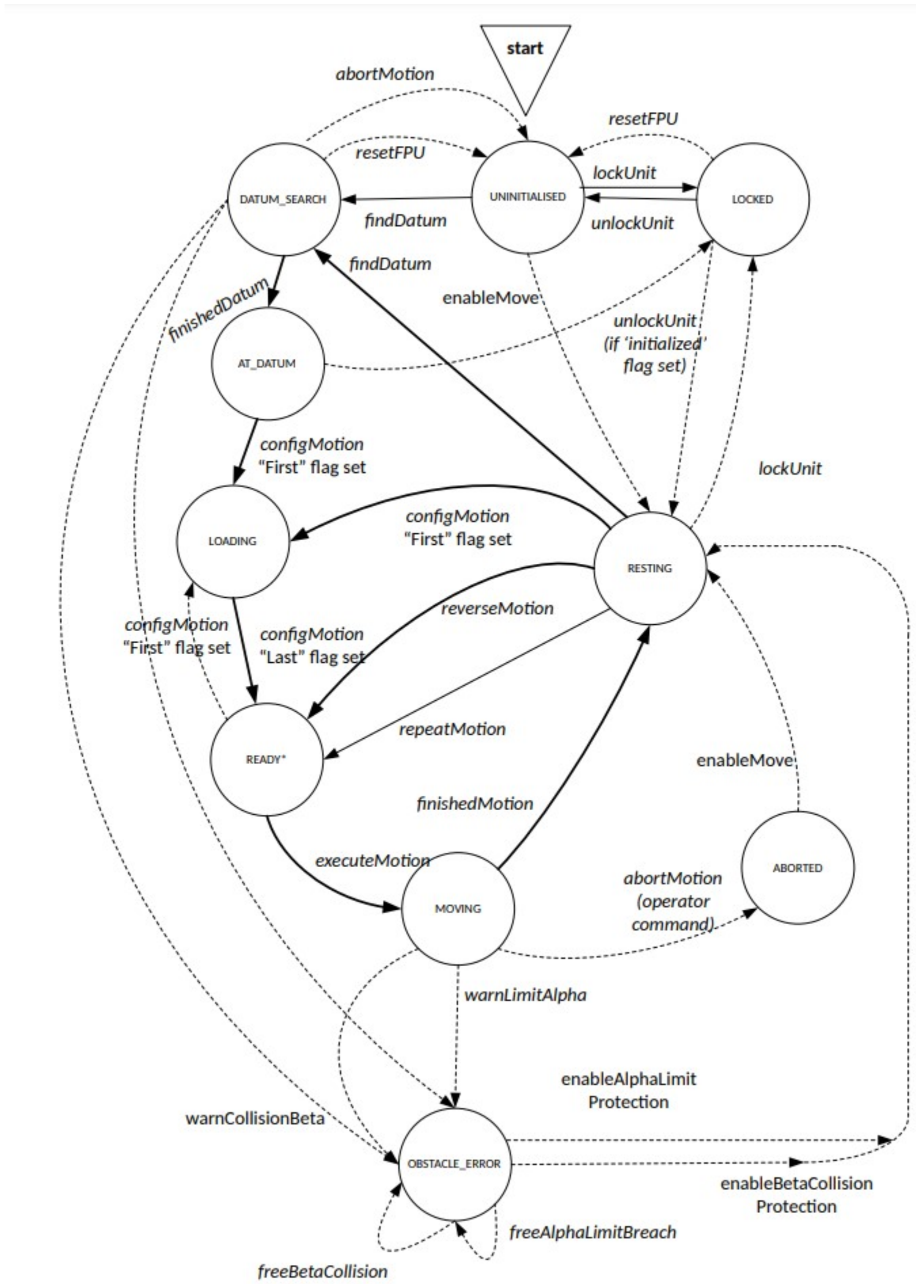


Figure 7. Fibre positioner firmware state diagram.

1.1 **Motion Restrictions**

The MOONS fibre positioners are required to acquire targets with a positional accuracy of around 20 microns. The following limitations need to be applied to achieve that accuracy and sure that the fibre positioner stepper motors will not lose steps:

- A minimum speed of 500 steps/second, to avoid low frequency resonances, and a maximum starting and stopping speed of 550 steps/second, to avoid step losses from sudden stops or starts. The element at the very beginning or at the very end of a waveform is an exception: smaller step increments can be used here to complete the movement. The firmware treats these end elements as a special case and moves the motor at minimum speed for a fraction of the 125 millisecond element time step.
- A maximum speed of 2000 steps/second, to maintain enough motor torque.
- A maximum change in speed of ±400 steps/second between each waveform element, to avoid step losses. The one exception is the larger step change needed to jump over the low frequency resonances.

2.4 High-level Software Design

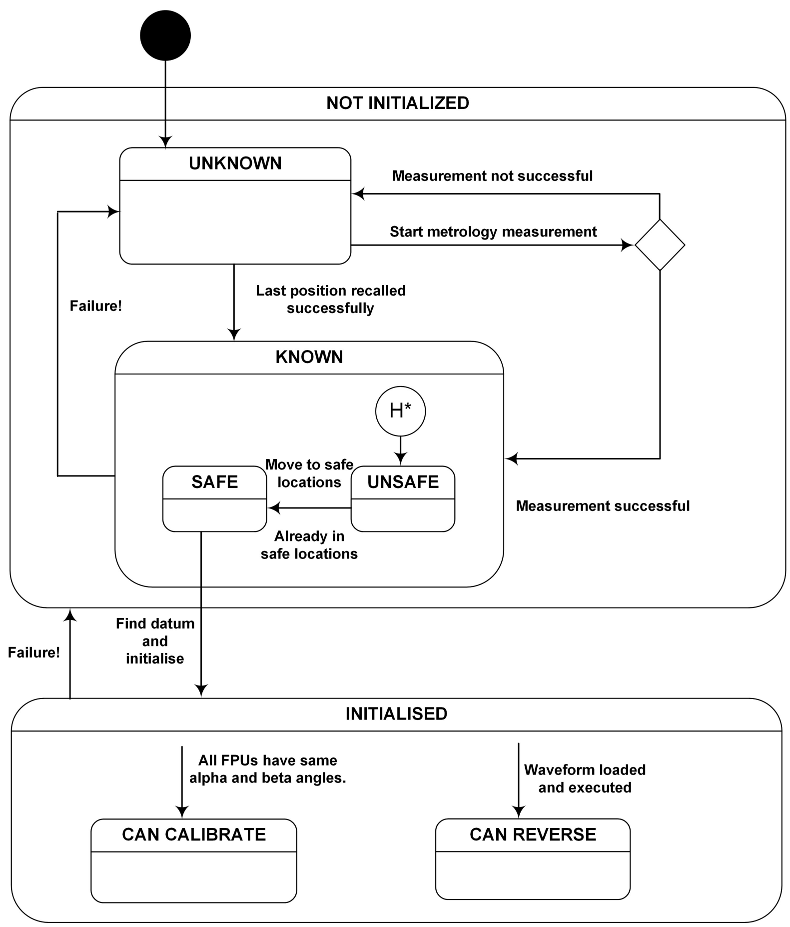


Figure 8. Fibre positioner top-level state diagram.

The high-level software provides the interface between the grid driver software (which treats the fibre positioner system as a collection of individual devices) and the ESO instrument control software (which treats the fibre positioner system as a single, very complex, device). A challenge for the top-level software design is managing a mechanism which cannot simply be reinitialised by searching for its datum reference. There is a kind of “catch 22” situation where the mechanism must be initialised before it can be moved, but the mechanism must be moved to a safe place before it can be initialised.

We have solved this conundrum with a state machine (Figure 8) which manages what the fibre positioner system can and cannot do at any moment. The states are described in detail in Table 2.

The fibre positioner software uses a Lightning Memory-Mappped Database (LMDB) to maintain an up-to-date description of its status. The last known configuration of the fibre positioners is saved to that database after each command. It is also written whenever the software is shut down and recalled when the software is restarted. The software can operate reliably when the information from that database is consistent with the information provided by the firmware. When the two sets of information differ (such as might happen after a power failure) the metrology system can correct that discrepancy and provide accurate location information.

The datum reference switches are located at motor angles (alpha=-180.0°, beta=6.5°) where the fibre positioners cannot overlap and collide. The positioners will normally be safely parked near to these switches so they may be referenced easily after starting the software. But if the positioners end up in different locations (due to a fault or engineering work) they must be moved back to safe locations before the datum switches can be referenced. Fibre positioners are SAFE to datum as long as they lie within their local patrol zone, as shown in Figure 9, which in practice positioners means their beta angle must be closer to zero than ±20°.

Table 2. Fibre positioner states.

States	Description
INITIALISED/NOT INITIALISED	The fibre positioner system can only be operated in the INITIALISED state. The system needs to move to the INITIALISED state at the beginning of each night, usually by finding the datum references.
KNOWN/UNKNOWN	The fibre positioner system is in the KNOWN state when the fibre positioner configuration is reliably known by the software. A power failure which erases information about the fibre positioners will send the system to the UNKNOWN state. A metrology camera system can be used to measure the fibre positioner configuration and recover back to the KNOWN state.
SAFE/UNSAFE	The fibre positioners are in a SAFE state when all of them are within their local patrol area (see Figure 9). When they are in the SAFE state, fibre positioners can move to their datum switches without risk of colliding.
CAN CALIBRATE	The fibre positioner system only enters this state when every positioner has the same motor angles, to within a small safety tolerance. It is possible for all the fibre positioners to make identical movements from this state without a risk of collision.
CAN REVERSE	The fibre positioner system enters this state when a waveform has been loaded and executed. It is then possible to reverse the movement by sending the reverseMotion command.

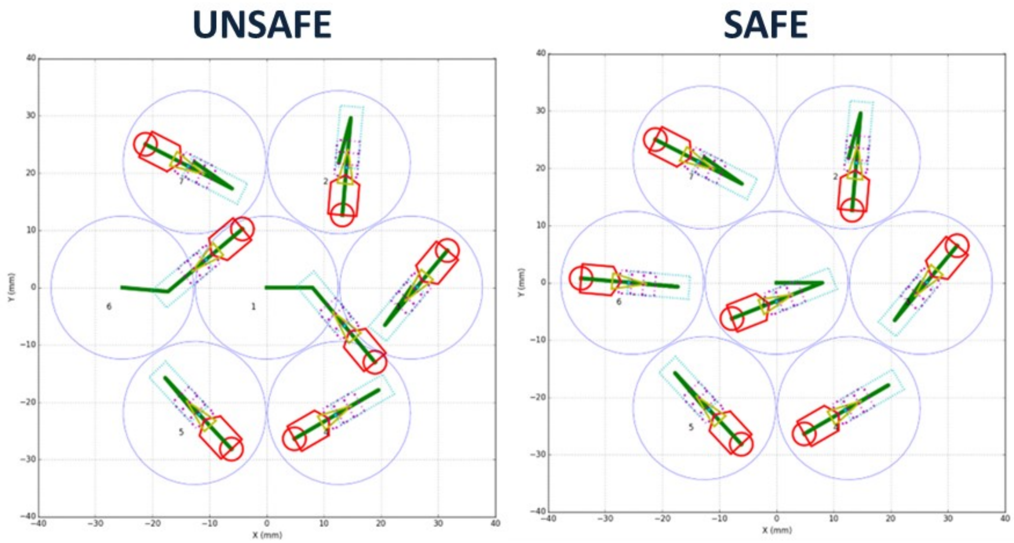


Figure 9. The difference between UNSAFE and SAFE states. The blue boundaries show the 12.5mm radius local patrol areas.

3. PATH PLANNING ALGORITHMS

Moving from one configuration to another without colliding is an extremely important part of the control of the MOONS fibre positioners. The MOONS control system currently uses two kinds of path planning algorithm.

3.1 The Discretized Navigation Function (DNF) Algorithm

The basic version of the DNF algorithm was developed at EPFL [5], [6]. The original algorithm was aimed at a simpler fibre positioner design with a smaller degree of overlap. Fibre positioners are treated as small robots with two rotating arms. The location of the fibre holder of fibre positioner i, (x_i,y_i), is determined by the following equation:

$$\begin{pmatrix} x_i \\ y_i \end{pmatrix} = \begin{pmatrix} x_{i_{cen}} \\ y_{i_{cen}} \end{pmatrix} + \begin{pmatrix} \cos \alpha_i & \cos(\alpha_i + \beta_i) \\ \sin \alpha_i & \sin(\alpha_i + \beta_i) \end{pmatrix} \begin{pmatrix} l_\alpha \\ l_\beta \end{pmatrix} \quad (1)$$

where (x_{icen},y_{icen}) is the location of the fibre positioner centre, α_i and β_i are the alpha and beta arm angles, and l_α and l_β are the lengths of the alpha and beta arms (which are 8mm and 17mm for the MOONS positioners). If q_i represents the coordinate of a point (x_i,y_i) on positioner i and $\|q_i - q_j\|^2$ is the Cartesian distance between those two points on positioners i and j; the original EPFL DNF algorithm defines a navigation function of the form:

$$\psi_i = \lambda_1 \, \|q_i \; - q_{i\_target}\|^2 + \lambda_2 \sum_{j \neq i} min\left(0, \frac{\|q_i - q_j\|^2 - D^2}{\|q_i - q_j\|^2 - d^2}\right) \quad (2)$$

where the first term (before the +) defines an attractive force between fibre holder qi and its target, and the second term (after the +) defines a repulsive force between the fibre holders belonging to this positioner, i, and its 6 neighbouring positioners, j. The λ terms are tuneable weighting factors. The D term defines a switch-on distance below which the repulsive force is activated, and the d term defines a safety distance at which the repulsive term becomes infinite. Theoretically, the positioners are not able to get closer than this distance.

The navigation function determines the net "force" felt by the positioner's fibre holder. The positioner will then attempt to move in the direction of that force. A set of chain derivatives are used to convert the (δx, δy) of the force in Cartesian space into (δα, δβ) motor demands.

The MOONS positioners are more complicated than the ones used in the original EPFL DNF algorithm, and a collision is possible not only between the fibre holders, but along the whole of the zone carrying the metrology targets. In addition, the MOONS beta arms also have datum actuators that can conflict with each other. The collision zones defined in the MOONS DNS algorithm are shown in Figure 10.

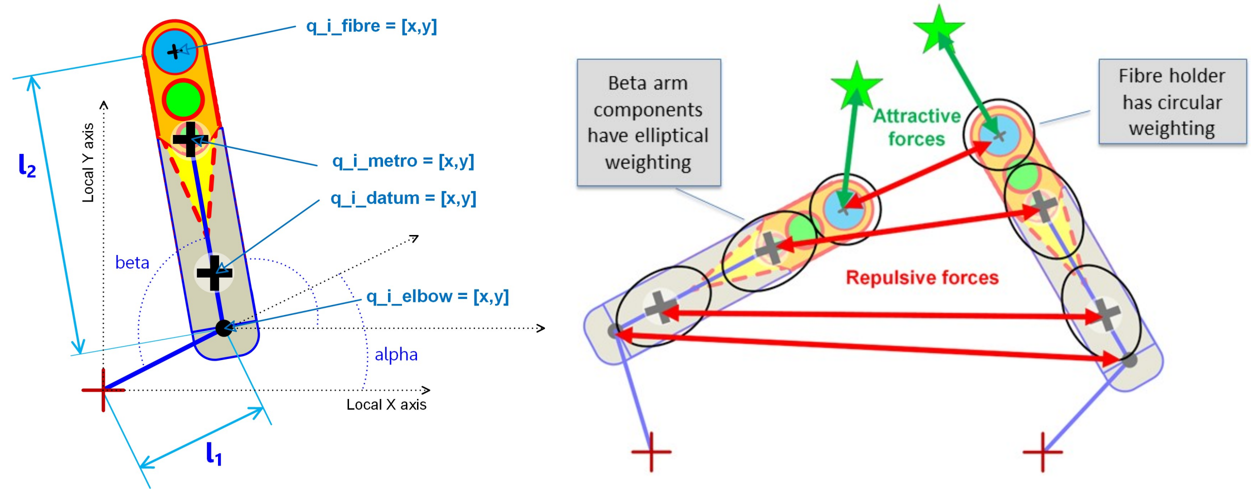


Figure 10. The MOONS DNF algorithm defines 4 reference points on each positioner (left), corresponding to the fibre holder, the metrology targets, the beta datum actuator, and the "elbow" joint. Positioners are navigated by means of imaginary forces: attractive forces between the fibre holders and their target and repulsive forces between zones on the beta arm which can collide.

MOONS uses an extended version of equation (2) which accumulates the attractive force to the target and the repulsive force between all the conflicting regions. The combined equation would look like this:

$$\begin{aligned}\Psi i = \lambda_1 \times A(q\_i\_fibre - q\_i\_target) + {} & \lambda_2 \times R(q\_i\_fibre - q\_j\_fibre) + \\ & \lambda_3 \times R(q\_i\_fibre - q\_j\_metro) + \\ & \lambda_4 \times R(q\_i\_metro - q\_j\_fibre) + \\ & \lambda_5 \times R(q\_i\_metro - q\_j\_metro) + \\ & \lambda_6 \times R(q\_i\_elbow - q\_j\_metro) + \\ & \lambda_7 \times R(q\_i\_metro - q\_j\_elbow) + \\ & \lambda_8 \times R(q\_i\_datum - q\_j\_datum) + \\ & \lambda_9 \times R(q\_i\_datum - q\_j\_elbow) + \\ & \lambda_{10} \times R(q\_i\_elbow - q\_j\_datum) + \\ & \lambda_{11} \times R(q\_i\_elbow - q\_j\_elbow)\end{aligned} \tag{3}$$

where "A" is a short-hand for the attractive term in equation (2) and "R" is a short-hand for the repulsive terms. A considerable time was spent adjusting the λ weighting factors and the D and d terms to achieve the highest efficiency with the minimum number of collisions. Table 3 contains a very simplified version of the MOONS DNF algorithm in pseudocode.

Table 3. Simplified pseudocode for Discretized Navigation Function (DNF) path planning algorithm.

```
For each path element
       For each fibre positioner
              Determine locations of the 4 key points
              Determine chain derivatives from arm angles
              For each neighbouring positioner
                     Determine locations of the 4 key points
                     Use distances between points to derive
                     repulsive part of navigation function
       Determine distance to target
              Use target distance to derive attractive
              part of navigation function
              Convert navigation function to motor demands
              Adjust motor demands to meet constraints
       Next fibre positioner
Next path element
```

Although the original DNF algorithm was declared "collision free" because the repulsive term in equation (2) becomes infinite at the safety distance, d, this condition does not hold for the real MOONS positioners for the following reasons:

- The MOONS positioners have a finite deceleration and limits on the motor speed. The "`Adjust motor demands to meet constraints`" part of the algorithm often means a positioner can't stop before it breaks the "d" boundary.
- The additional repulsive terms in the navigation function in equation (3) can sometimes conflict with each other and cancel each other out. For example, a positioner may not be able to avoid a collision between fibre holders and between datum actuators at the same time.
- Positioners can sometimes be trapped by a pincer movement, where two positioners move in on a third. When the middle positioner attempts to move away from a 2nd positioner it can collide with the 3rd positioner. Pincer movement collisions are prevented in the MOONS software by freezing the motion of any positioner which is surrounded by two or more neighbours

MOONS uses large D terms and large λ weightings to reduce the number of collisions to a minimum, but these terms can often prevent positioners from coming together to pick off the close targets needed in XSWITCH mode. The MOONS software compensates for this problem by switching the following algorithm during the last phase of movement.

3.2 The Tiptoe Algorithm

The tiptoe algorithm is used to improve the efficiency of the DNF algorithm. During the last phase of movement all the repulsive terms in the navigation function equation (3) are turned off, and all the positioners start moving directly to their targets. But at the same time, the software switches on a more intensive collision detection. Each positioner is surrounded by a collision detection zone which is larger than the stopping distance. After each iteration, the software stops any positioners whose collision zones have overlapped. This extra phase allows most of the positioners that can reach their targets without a collision to reach their targets. Table 4 shows the tiptoe algorithm in pseudocode:

Table 4. Simplified pseudocode for "tiptoe" path planning algorithm.

```
For each path element
       For each fibre positioner
              Move slowly towards the target
              For each neighbouring positioner
                     If collision zones closer than tolerance
                            Request to stop the positioner
              Adjust motor demands to meet constraints
       Next fibre positioner
Next path element
```

3.3 The Scrooge Algorithm

MOONS also uses a path planning algorithm based on the one used by the Sloan SDSS-V fibre positioners [7]. SDSS-V has fibre positioners of a similar design to MOONS and their movements are governed by the same equation (1). The SDSS-V software (also known as "Kaiju") attempts to minimise two quantities:

- **Cost**, C_i: The distance between each fibre positioner and its target, in terms of motor angles, which for MOONS will be

$$C_i = \left\| \begin{pmatrix} \alpha \\ \beta \end{pmatrix} - \begin{pmatrix} \alpha_{target} \\ \beta_{target} \end{pmatrix} \right\| \quad (4)$$

- **Energy**, E_i: A measure of local crowding, determined from the sum of the inverse squares of the closest approach distances between pairs of positioners, D_{ij}. For each positioner i, j is iterated over its neighbours.

$$E_i = \sum_{j \neq i} \left(\frac{1}{D_{ij}} \right)^2 \quad (5)$$

The algorithm builds a path for each fibre positioner in a similar way to the DNF algorithm, except at each stage a positioner can choose to move each motor forwards, backwards or leave it where it is. Each of the 9 possible moves results in a change to the cost and/or energy (see Figure 11). Any move which would result in a collision between two positioners is not allowed. The algorithm comes in two variants:

- Greedy Choice: In this variant, each positioner attempts to minimise its cost, as long as the move will not result in a collision.
- Markov Chain: In this variant, each positioner is governed by two weighting factors: a greed factor determines its desire to minimise the cost; and phobia factor determines its desire to minimise energy. Positioners decide probabilistically based on these weights – sometimes choosing to move towards their target to minimise cost and sometimes choosing to move away from other positioners to minimise energy. The random nature of this algorithm can reduce the amount of deadlock

The SDSS-V algorithm is found to be most efficient when run backwards: i.e., positioners begin at their targets and attempt to move back to the starting position

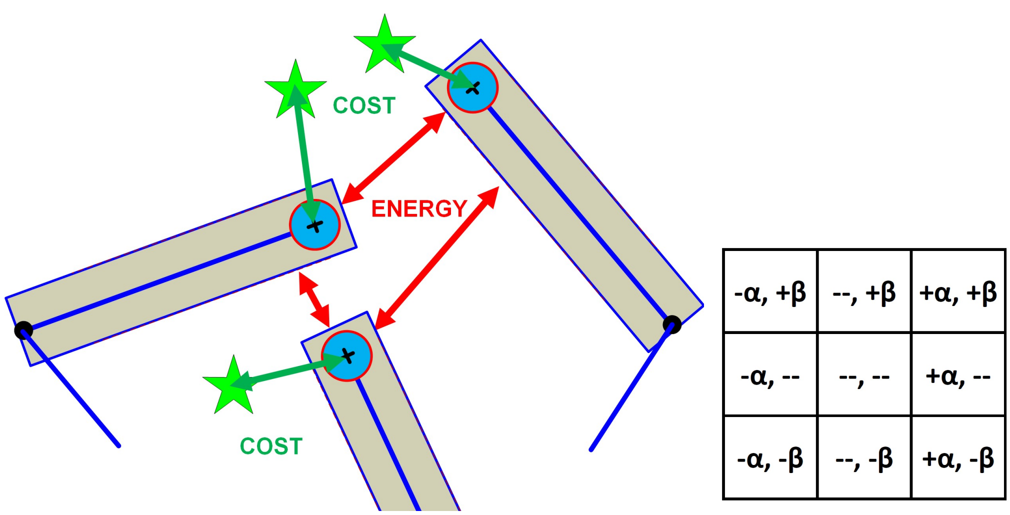


Figure 11. The Greedy Choice/Markov Chain algorithms (left) attempt to minimise the total cost (defined from the distance between the positioners and their targets) and the energy (defined from the closest approach distances between neighbouring positioners). Each positioner has a choice of 9 things it can do at each iteration of the algorithm (right): move each arm forwards, backwards or stay still.

Table 5 shows a very simplified version of the SDSS-V path planning algorithm in pseudocode.

Table 5. Simplified pseudocode for SDSS-V path planning algorithm.

```
For each path element
      For each fibre positioner
            Compute cost from distance to target
            If the cost is zero (target reached)
                  If no neighbour nearby
                        Stay at target. Next positioner.
            Compute energy from distances to neighbours
            For each of the 9 options
                  If the option doesn't risk a collision
                        Compute the change in cost and energy
            Randomly choose to minimise cost or energy.
            Choose option which minimises cost or energy.
            Adjust motor demands to meet constraints
      Next fibre positioner
Next path element
```

One of us (Paolo Franzetti) has developed a variation of the Greedy Choice algorithm for MOONS called "Scrooge". The MOONS fibre positioners have different motor constraints from SDSS-V, so we have had to make the following adjustments to the algorithm:

- The Scrooge algorithm moves the motors at close to the minimum speed, so they are able to stop at any time (when the "stay still" choice is made).
- The MOONS fibre positioners always acquire targets with a right-armed parity, because the algorithm is more efficiency when all the arms move in the same direction from the starting beta angle.
- The algorithm uses a starting position with the alpha arm at the lower end of its range. Again, this is to allow all the arms to move in the same direction (in this case towards the upper end of the alpha range).

Although the motors move more slowly, the Scrooge algorithm can achieve a much higher target acquisition efficiency than the DNF algorithm (see Figure 17) and is likely to be used for most MOONS science cases. The greater flexibility of the DNF algorithm means it will still be used for initialisation and fault recovery (see below).

4. MOONS OPERATION

4.1 Observation Planning and Execution

The path planning algorithms can take a significant amount of time to run (up to 2 minutes for the DNF algorithm and up to 30 minutes for the Scrooge algorithm) so MOONS paths are generated in advance during observation planning. Figure 12 illustrates how this works. Target coordinates and paths are received from the observation database, the paths are executed and then (if the final adjustment is sufficiently small) the positioners are adjusted to compensate for changes due to atmospheric refraction or thermal expansion of the focal plate.

The flow chart also allows the same targets to be observed in consecutive observations. The positioners are only reversed back to their starting positions if new target coordinates are detected.

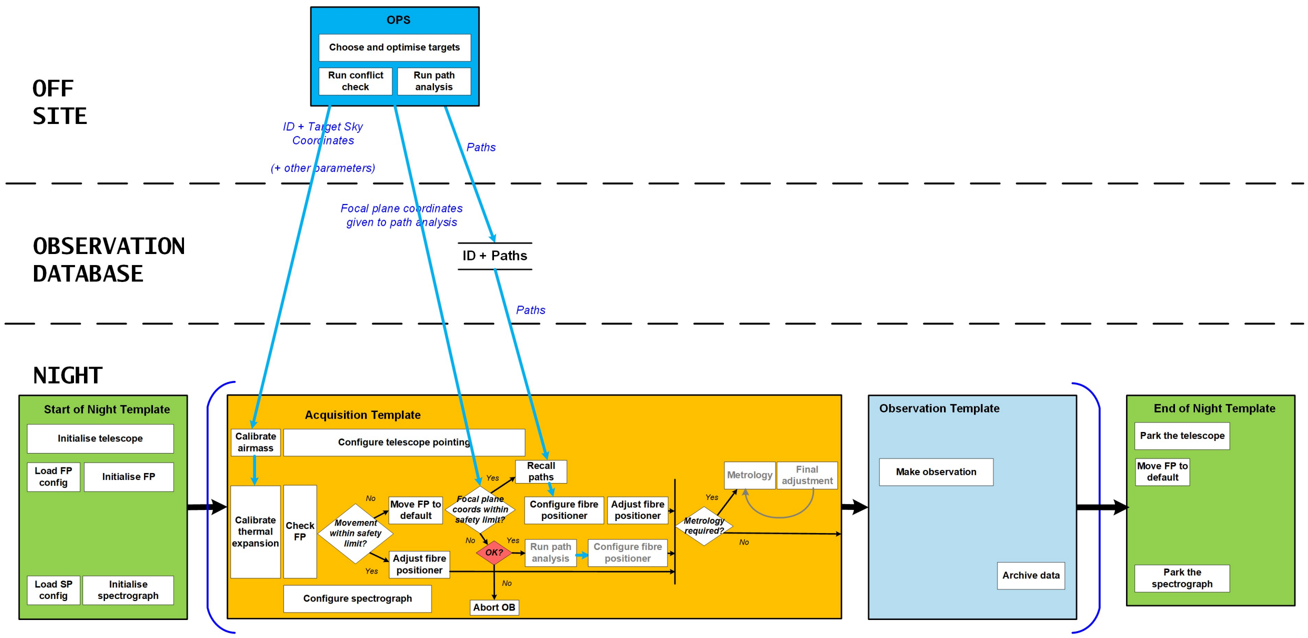


Figure 12. An illustration of how the MOONS fibre positioner software is used during operation. The blocks between the brackets can be repeated as many times as necessary to make all the observations planned for a night.

4.2 Initialisation and Fault Recovery

Several recovery strategies have been built with the DNF algorithm to allow the positioners to be restored to safety:

- **SPACE**: The attractive term in equation (3) is switched off and the algorithm is run with an increased switch-on distance, D. This will cause the positioners to move as far apart as possible. It is a useful strategy to move positioners apart after a collision.
- **FSPACE**: Each positioner is given a target which is chosen to maximise the spacing between itself and its neighbours and attempts to move to that target. Effectively, this gives each positioner the easiest possible target to move to and is a more controlled way of creating space than the SPACE mode.
- **SAFE**: Each positioner is given a target which corresponds to the nearest safe location and attempts to move to that target.
- **RSAFE**: Each positioner is given a target which corresponds to a *random* safe location and attempts to move to that target. Using a random location helps to bounce some positioners out of deadlock.
- **DEFAULT**: Positioners attempt to move back to their default starting locations.

Figure 13 contains flow charts showing how those recovery strategies may be used to bring the fibre positioners to a safe state at the beginning of the night, or after a fault such as a collision. The fibre positioners are normally parked during the day in a safe state. If nothing has changed since they were parked, the flow will pass through the upper part of the "Initialise FP" box of Figure 13; and initialisation consists of finding the datum reference point. A metrology measurement [3] can be used to determine the locations of the fibre positioners after a fault.

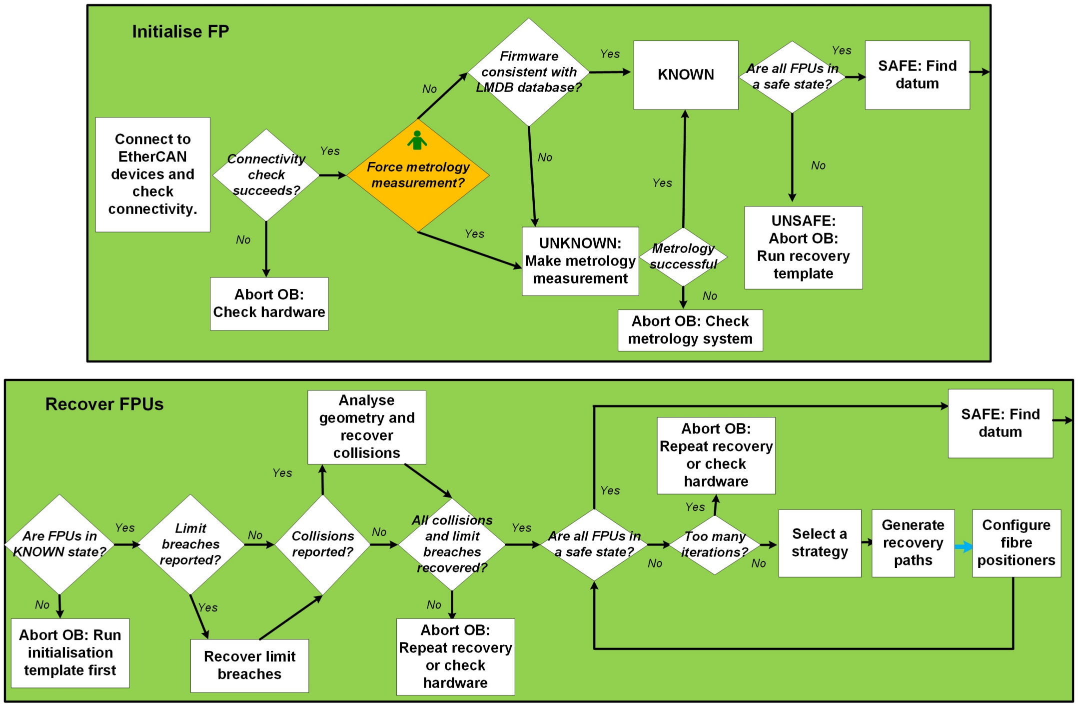


Figure 13. Fibre positioner initialisation procedure (top) and recovery procedure (bottom).

The "Recover FPUs" box in Figure 13 shows the procedure needed to bring fibre positioners back to a SAFE state after a fault. Low-level firmware commands are first used to back the positioners away from a collision or limit breach. When the collision detection system (Figure 6) indicates the collision has been resolved, the DNF path planning algorithm is then used iteratively to bring the fibre positioners back to a SAFE state.

5. RESULTS

5.1 Fibre Positioner Synchronization

As discussed earlier, it is imperative that the synchronisation between moving FPUs is always within the few milliseconds required, because the path planning and collision avoidance algorithms rely on this assumption for their dynamic movement path generation. In practice, this means that the 3 x EtherCAN gateway units must send CAN bus broadcast FPU movement trigger commands to their 15 CAN buses with a timing synchronisation of a few milliseconds.

The accuracy and reliability of this synchronisation needed to be tested in detail. The requirements for this testing were as follows:

- Timing measurement resolution of 1 millisecond or less
- Testing the synchronisation across as many FPUs, CAN buses and gateways at once as possible, and in various connectivity configurations.

The MOONS metrology camera [3] can't be used to measure synchronisation because it cannot achieve the required time resolution, even if the camera is operated in video mode. The time measurement requirements were achieved by using a 16-channel logic analyser to view and measure the relative synchronisation timings between the initial stepper motor electrical movement pulses across up to 16 FPUs at once. For these measurements, a set of 88 spare FPU printed circuit boards were used because it wasn't desirable to solder monitoring wires onto full production FPUs. These 88 FPU PCBs were mounted onto panels and their CAN bus connectivity could be configured into various arrangements across different CAN buses and gateways to maximise test coverage. The test setup is shown in Figure 14:

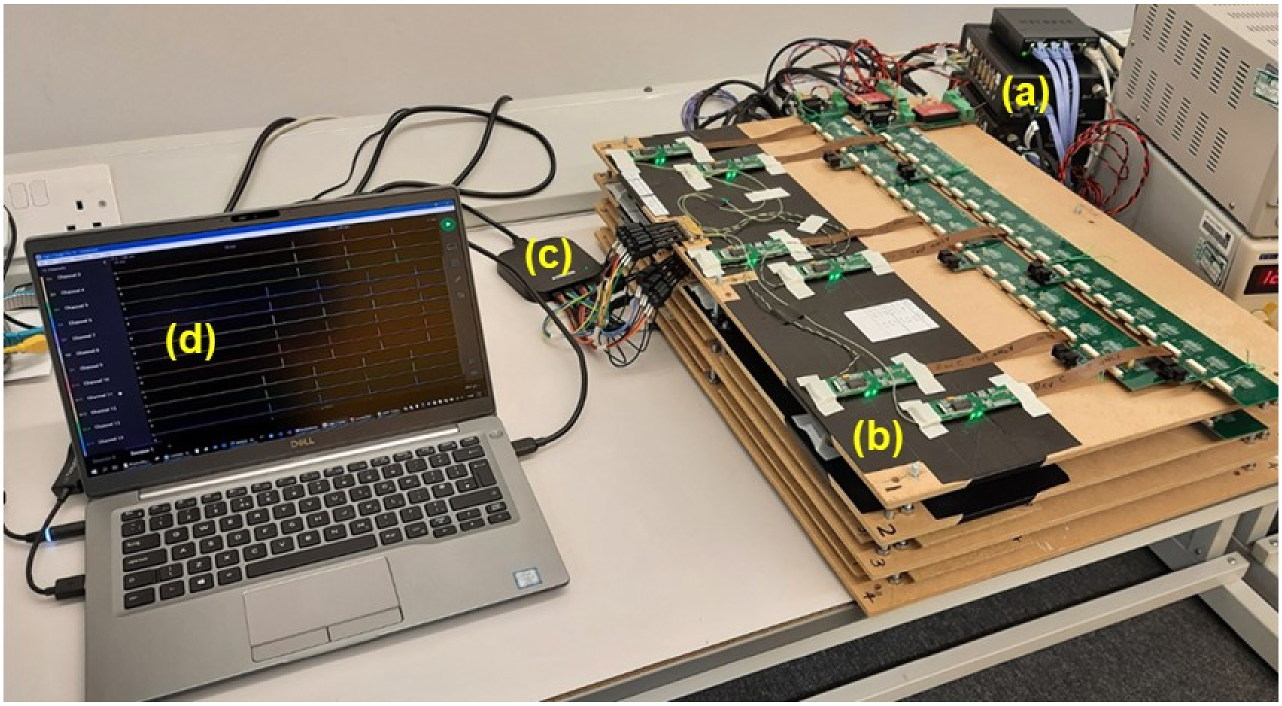


Figure 14. The test bench used to measure the fibre positioner synchronisation, showing the EtherCAN cards (a), fibre positioner CANBus units (b), logic analyser (c) and laptop (d).

This testing confirmed that the worst-case synchronisation was about 1ms for over 300 test runs in various CAN bus configurations, and that the synchronised triggering mechanism was very reliable and stable. As an example, the following logic analyser screenshot (Figure 15) shows a typical <1ms synchronisation timing spread of the first few stepper motor pulses from FPUs connected to the 3 different gateways.

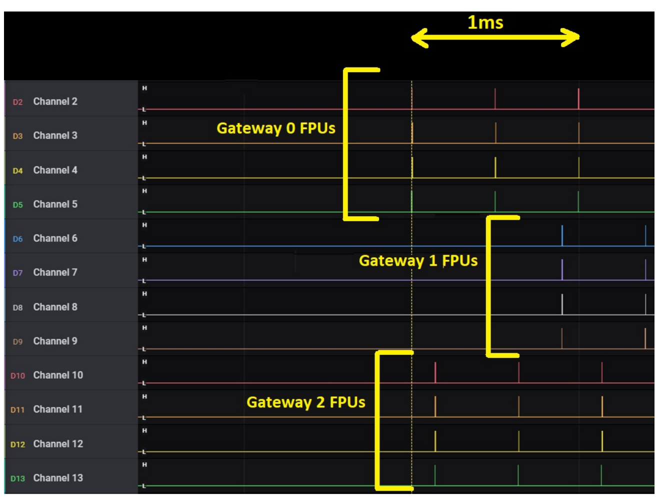


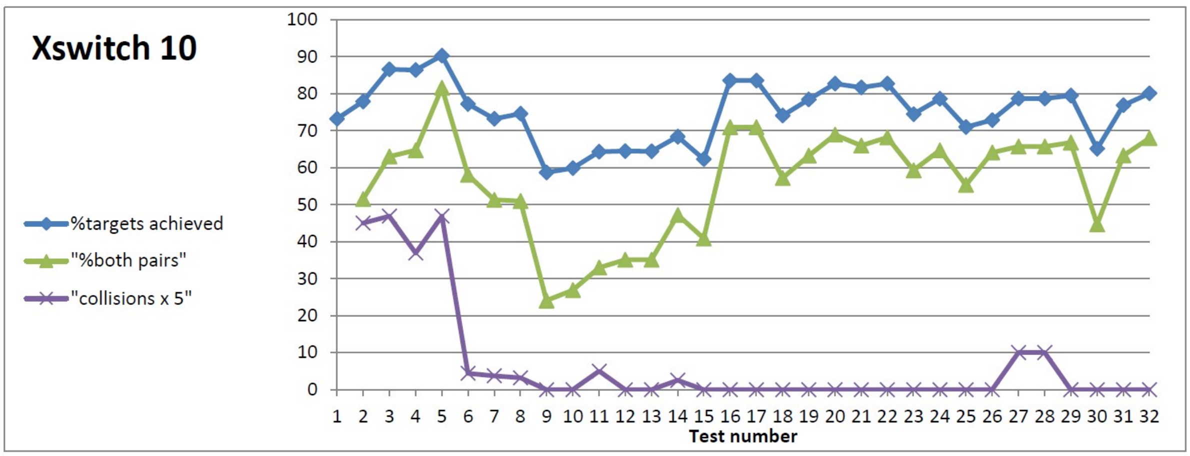
Figure 15. Screenshot showing the fibre positioners starting their motion within 1ms of the trigger pulse.

5.2 Fibre Positioner Positioning Accuracy

The performance of each fibre positioner was verified using an automated test rig, as described in [4]. This rig confirmed an average positioning repeatability of 12.0 microns and an average positioning accuracy of 35.8 microns. The test rig generated a gearbox calibration lookup table for each fibre positioner, which is taken into account by the software when making the final positional adjustments shown in Figure 12.

5.3 Path Planning Efficiency

The DNF algorithm was tuned using two sets of test cases: targets selected for XSWITCH mode with an object/sky spacing of 30ms; and targets selected for XSWITCH mode with a more difficult object/sky spacing of 10ms. Figure 16 shows the evolution of the efficiency of the MOONS DNF algorithm as it was developed. Each rise in target acquisition efficiency is due to an improvement in the balance of the avoidance zone sizes (D and d) and weighting factors (λ). The big reduction in collisions at test 6 was due to the introduction of a repulsive field around the datum zone. An adjustment in motor speed limits caused the reduction in efficiency in test 9, but the introduction of the "tiptoe" algorithm at test 16 restored that efficiency. Another restriction in motor acceleration limits caused some collisions in tests 27 and 28, but those were again removed by readjusting the size of the repulsive fields. The target assignment efficiency is now around 90% for STARE mode and easy XSWITCH observations and 80% for the most difficult observations, although the efficiency drops below 70% if you only count cases where both the object and sky fibres of a pair reached their target.



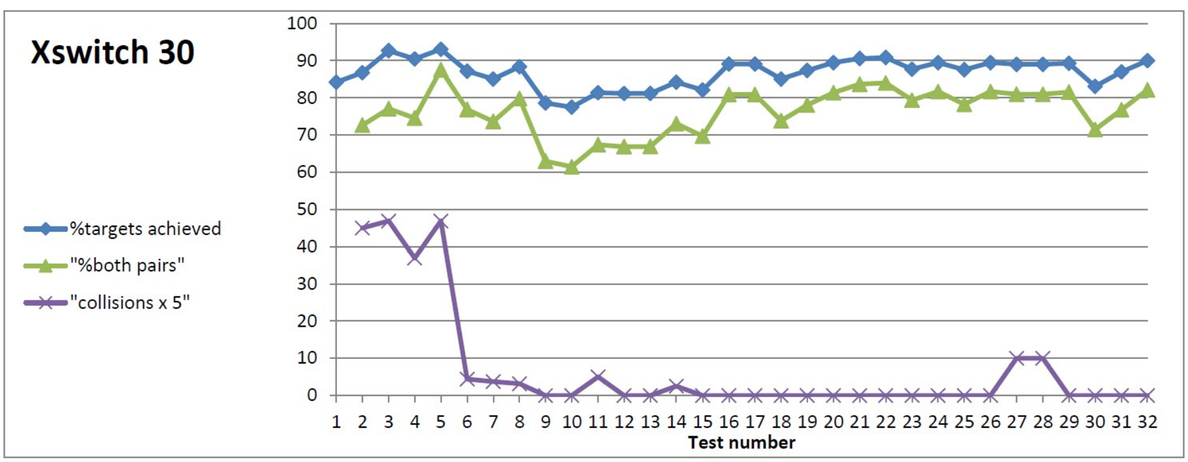


Figure 16. Evolution of the target acquisition efficiency of the MOONS DNF algorithm during development for easy XSWITCH observations (30 arcsec) and difficult XSWITCH observations (10 arcsec). The blue line at the top of each graph shows the fraction of individual fibres which successfully reached their targets. The green line in the middle shows the fraction object/sky pairs in which fibres successfully reach *both* locations. The purple line at the bottom shows the number of collisions detected (multiplied by 5 to exaggerate small numbers). Collisions now remain at zero.

The efficiency of the DNF and Scrooge algorithms was compared using simulations of real extragalactic observations made in XSWITCH mode. The results are shown in Figure 17, which compares how successfully the algorithms could move both the object and sky fibres in an XSWITCH pair to their targets. The X axis shows how the number of available targets gives the MOONS observation preparation software (Moonlight) more chances to give the DNF algorithm an easier combination of targets. The efficiency of the DNF algorithm changes from around 65% to 80% as the target density increases, but the Scrooge algorithm achieves around 98% efficiency regardless of the target density.

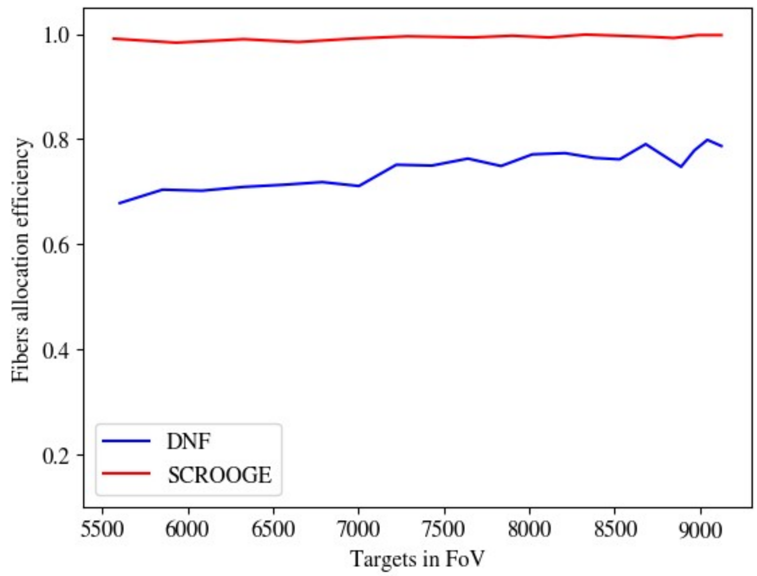


Figure 17. A comparison between the target allocation efficiency of the Scrooge algorithm compared with the DNF algorithm from simulations of extragalactic observations made in XSWITCH mode. The Y axis shows the fraction of pairs of fibres which successfully reach both their targets, and the X axis shows how this efficiency varies with number of available targets. The efficiency of the DNF algorithm improves with target density because the MOONS observation preparation software (Moonlight) has more opportunities to choose targets that are easier for the algorithm to achieve. The Scrooge algorithm appears to be very good at all target densities.

The results are a work in progress, and we will continue to test both algorithms as more information on the behaviour of the MOONS fibre positioners becomes available.

6. CONCLUSIONS

Controlling the MOONS fibre positioners poses some difficult challenges due to their high degree of overlap and the need to move them quickly and simultaneously to their targets. The control system must also be resilient to unexpected faults. We have developed the following solutions to these problems:

- Each fibre positioner has an alpha arm capable of swivelling around its centre and a beta arm capable of placing a fibre anywhere within a doughnut-shaped patrol field 9-12.5 mm from the centre.
- Each fibre positioner is controlled by a local microprocessor capable of storing and executing a motion waveform of up to 256 x 125ms elements per motor on demand.
- A collision protection system ensures that if any neighbouring positioners touch their motors are powered down and an error us reported to the software.
- Fibre positioners receive commands over CANBus. The firmware recognises several low-level commands, including to find the datum reference, to download a new movement waveform and to execute that movement waveform. There are also commands to free a collision or limit breach.

- Communication with the fibre positioners is managed by three EtherCAN gateway cards, which can synchronise all the fibre positioner movements to within a millisecond.
- MOONS has two path planning algorithms (DNF or Scrooge) which are both capable of moves the positioners from a starting location to their targets without colliding. Scrooge takes longer to execute but can achieve target assignment efficiencies as high as 98%, whereas DNF usually achieves 65-90% (and is better for STARE mode observations or for XSWITCH observation with a high density of available targets). The DNF algorithm is more flexible and can be used for on-the-fly recovery after a fault.
- High-level software ensures that all the fibre positioners are in the correct state before they are initialised or operated. The high-level software also provides DNF-based procedures to recover positioners back to a safe state after a fault.

REFERENCES

[1] Gonzalez, O., et al., "MOONS – Multi Object spectroscopy for the VLT: overview and instrument integration update “, Paper 12184-38 SPIE Astronomical Telescopes + Instrumentation (2022).
[2] Watson, S., et al., "MOONS fibre positioning module: instrument build overview", Paper 12188-79 SPIE Astronomical Telescopes + Instrumentation (2022).
[3] Gonzalez, O., et al., "MOONS – Multi Object spectroscopy for the VLT: Design and testing of the MOONS metrology system”, Paper 12188-138 SPIE Astronomical Telescopes + Instrumentation (2022)
[4] Gutierrez, P., et al., “Ethernet to multi-CAN gateway for VLT MOONS instrument control”, Proc SPIE 11452, 2020 [https://doi.org/10.1117/12.2557215]
[5] Makarem, L. et al., "Collision Avoidance in Next-Generation Fiber Positioner Robotic System for Large Survey Spectrograph", Astronomy & Astrophysics, 566, p.A84, 2014.
[6] Makarem, L. et al., “Collision-free Motion Planning for Fiber Positioner Robots: Discretization of Velocity Profiles”, Proc. SPIE 9152
[7] Sayers, C et al., “Fast, Collision-Free Trajectory Planning for Heavily Overlapping Robotic Fiber Positioners”, Astronomical Journal, 161 2 2021. [https://arxiv.org/abs/2012.04721]
[8] Levi, M. E. et al. “The Dark Energy Spectroscopic Instrument (DESI)”, [https://arxiv.org/abs/1907.10688]