Trajectory handling for LINC-NIRVANA's Field Derotators

Frank Kittmann[1a], Thomas Bertram[1], Matthew Horrobin[2], Albert Conrad[1], Jan Trowitzsch[1], Florian Briegel[1], Jürgen Berwein[1], and Lars Mohr[1]

[1] Max-Planck-Institute, Königstuhl 17, 69117 Heidelberg, Germany
[2] I. Physikalisches Institut, Universität zu Köln, Zülpicher Straße 77, 50937 Köln, Germany

Abstract. The near infrared interferometer LINC-NIRVANA combines the beams coming from the two primary mirrors of the Large Binocular Telescope to increase the resolution of the science camera image. LINC-NIRVANA is using layer-oriented multi-conjugate adaptive optics (MCAO) to reduce the influence of the atmospheric turbulence. The deformable mirrors of the MCAO systems are conjugated to the ground layer and a second layer in the upper atmosphere, respectively. Ground layer wavefront sensors and high layer wavefront sensors measure the wavefront in these two layers. Due to geometrical constraints unique to this type of instrument, it is not possible to provide a single derotator at the entrance of each incoming beam. Due to that fact, field derotation has to be applied for each wavefront sensor and for the science detector separately. The fields of the high layer wavefront sensors are derotated by the use of K-Mirrors, whereas the ground layer wavefront sensors and the science detector rotate themselves to compensate field rotation. All derotators are driven by common software that provides functionality to generate, execute, and maniulate a trajectory. The trajectory handling for these field derotators will be discussed in this paper. We introduce an algorithm that translates a requested trajectory into motor controller commands in due consideration of the required accuracy. Finally, we present the achieved positioning accuracy of the translated trajectory and compare that result with the traced position path of the derotator.

1 Introduction

LINC-NIRVANA[1] is the Near-Infrared interferometric imaging camera for the Large Binocular Telescope (LBT). Once operational, this joint German-Italian project will be able to provide an unprecedented combination of high angular resolution, photometric sensitivity and Field of View (FoV). By coherently combining the two 8.2 m beams of the LBT, the full 23 m extent of the binocular aperture can be utilized to achieve angular resolutions of, in the best case, 10 milli-arcseconds.

The instrument consists of a number of systems that control the incoming wavefronts and ensure a time stable diffraction limited interferometric image in the focal plane of the science camera. A layer-oriented MCAO system is employed for each arm of the interferometer. Deformable mirrors for real time wavefront correction are conjugated to the ground layer of the atmosphere and to an additional layer in the upper atmosphere, at ~7100m above the telescope. The groundlayer wavefront sensor (GWS) can acquire up to 12 natural guide stars in an annular FoV with a diameter of 6 arcminutes. The high layer wavefront sensor (HWS) can acquire up to 8 additional natural guide stars within the central 2 arcminute FoV. Groundlayer correction will be applied via the adaptive secondary mirrors of the LBT; an additional Xinetics-349 deformable mirror (DM) on each side is used to apply highlayer corrections. The central ~1 arcminute will also be exploited by the Fringe and Flexure Tracker System (FFTS). It is located inside the cryostat of the instrument and deals with the cophasing of the two incoming wavefronts by analyzing the PSF of a natural reference star and correcting phase delays with a dedicated piston mirror. The NIR science camera covers the central 10 arcsecond FoV.

Each of the aforementioned systems operates in its own focal plane. A total of 6 focal planes (2 × GWS, 2× HWS, FFTS, science camera) are realized within the instrument. LINC-NIRVANA will be installed at two of the Gregorian focal stations of the LBT. The optics of the binocular telescope and the instrument are mounted on a single gimbal mount. Because of this altitude-azimuth mounting, the fields imaged in the focal planes are subject to diurnal rotation as the telescope tracks a target on sky. Field derotators have to be used to counteract the field rotation in each focal plane and to maintain the orientation of the images over long periods. A combined derotation of science and AO

[a] kittmann@mpia.de

fields, introduced at the entrances of the instrument, is not possible due to geometrical constraints – otherwise the interferometric FoV could not be realized[2]. The mechanical realization of the field derotators varies from system to system. A K-Mirror in the optical path is used to derotate the HWS fields. In the case of the GWS, the sensor rotates itself. The same for the science detector. The FFTS decomposes the circular trajectory of its off-axis reference star into its linear components and uses an xy positioning stage to follow the trajectory.

Each field derotator is managed by the system it is associated with. System software packages consider the specific high-level derotation use cases: for the wavefront sensors the orientation of the fields with respect to the sensors should be maintained as long as possible, whereas for the science detector, a more frequent change of the orientation may be required. On a lower level, however, all field derotators require the same functionality: trajectories have to be accurately executed and manipulated. Common software building blocks provide this functionality for all system software packages.

In this presentation we discuss the handling of motor trajectories in the framework of LINC-NIRVANA. An algorithm is presented which is used to translate requested trajectories into precise motor controller commands.

2 Requirements

Each field derotator type employed within LINC-NIRVANA has its own requirements concerning temporal and spatial positioning accuracy. The science detector requires a different degree of field stability than the GWS. On top of the specific requirements a set of common requirements can be identified, which drives the design of the common derotation software building blocks:

- **Maximum parallactic angle velocity:** Each field derotator shall be able to provide the required field stability while tracking objects down to zenith distances of 1.5°. At this distance the parallactic angle velocity is to 480 arcseconds per second.
- **Field rotation direction:** The field rotation depends on the hour angle and the declination of the target that is being tracked. The field derotators shall provide the required field stability independent of the declination of the source and the hour angle (outside the zenith distance limit). Changes of direction shall be considered.
- **Common motor control hardware:** The same motor controller hardware shall be employed for all field derotators (cf. Section 3)
- **Parallactic angle trajectory:** The parallactic angle trajectory shall be provided by the pointing kernel of the telescope. It will be distributed to the various field derotators within LINC-NIRVANA.
- **Field derotation trajectory:** A field derotation trajectory shall be executed by each derotator. The trajectory will be based on the parallactic angle trajectory and will consider the specifics of the field derotator mechanics.
- **Response time to changes:** Within 1 second the field derotator trajectory shall be adaptable to changes (in the parallactic angle trajectory or other).
- **Smooth transitions:** The field derotation trajectory shall be adaptable to new situations without having to stop the field derotation. Corrections to the field derotation trajectory shall be applied by altering the executed trajectory. Discontinuities in the trajectory shall be prevented.
- **Maximized field derotation range:** With the exception of the science detector, all derotators shall maintain the field orientation as long as possible. A change of the field orientation requires a reacquisition of the guide stars.

3 Motor Controller Trajectories

The same motor controller type is used to control the motion of all servo- and stepper motors within LINC-NIRVANA, including all field derotator drives. The “MoCon” is an in-house development, which can synchronously control up to 8 axes. It can execute trajectories which are provided as external profiles. An external profile represents the commanded trajectory as a piecewise polynomial of fourth degree. Each segment of this piecewise polynomial is parameterized by its duration and the start values for position, velocity, acceleration and jerk. A ring buffer stores the parameters of up to 2^{32} consecutive segments. Based on this information the motor controller introduces the required number of steps at each cycle of the internal clock of the MoCon (figure 1).

An external profile can be uploaded into the ring buffer and the starting point of its execution can be specified. While the external profile is being executed, it can be manipulated by altering the parameters of any upcoming segment in the buffer. This allows for changes of the trajectory or recurrent uploads of shorter sections of the trajectory without having to interrupt the motion.

```
FOR every determined coefficient DO
    CommandedP = NextPositionCoefficient
    CommandedV = NextVelocityCoefficient
    CommandedA = NextAccelerationCoefficient
    CommandedJ = NextJerkCoefficient
    SegmentD   = NextSegmentDuration
    FOR cycle = 0 to SegmentD DO
        CommandedP = CommandedP + CommandedV + CommandedA / 2 + CommandedJ / 6
        CommandedV =              CommandedV + CommandedA     + CommandedJ / 2
        CommandedA =                         + CommandedA     + CommandedJ
    ENDFOR
ENDFOR
```

Fig. 1. This pseudo code describes the MoCon internal method to reproduce the user-defined trajectory. The code sequence reads for every segment the coefficients and calculates the commanded values for each cycle of the internal clock of the MoCon. The meaning of the variables are: CommandedP - commanded position; CommandedV - commanded velocity; CommandedA - commanded acceleration; CommandedJ - commanded jerk; SegmentD - segment duration.

4 External Profile Generation

The common software of LINC-NIRVANA, "TwiceAsNice" [3], contains a package, "basda-mocca", which provides external clients with the ability to configure and control the MoCon hardware. It also provides the functionality to convert a user-defined trajectory into an external profile for the MoCon.

Figure 2 outlines the algorithm: The user-defined trajectory is provided as a sequence of positions which are equally spaced in time. A polynomial is fitted to each set of position samples that represent a segment in the external profile. A quadratic correction (figure 4) is added to the determined coefficients: position, velocity, and acceleration. The fit is compared with the input sequence. If it deviates by more than the specified fault tolerance, the segment length is reduced by half and polynomial fitted to each of the halves. The determined polynomial coefficients and the segment length are the parameters for the corresponding external profile segment. Finally, the terminating coefficients are calculated to stop the trajectory.

Figure 3 presents the result of the fitting algorithm without the quadratic correction and illustrates the fitting problem. The determined polynomial coefficients (green X) represent the user-defined trajectory (red dots). The green curve is based on these coefficients and is reproduced according to the MoCon internal method (figure 1). Due to rounding errors the reproduced trajectory is not continuous. The deviations at the end of each segment are within the specified fault tolerance, but thus lead to position jumps. Motors are not able to follow these jumps because of the inertia. That leads to a step loss. The additional quadratic correction of the determined trajectory (figure 4) ensures the continuity and convergence towards the user-defined trajectory (figure 5).

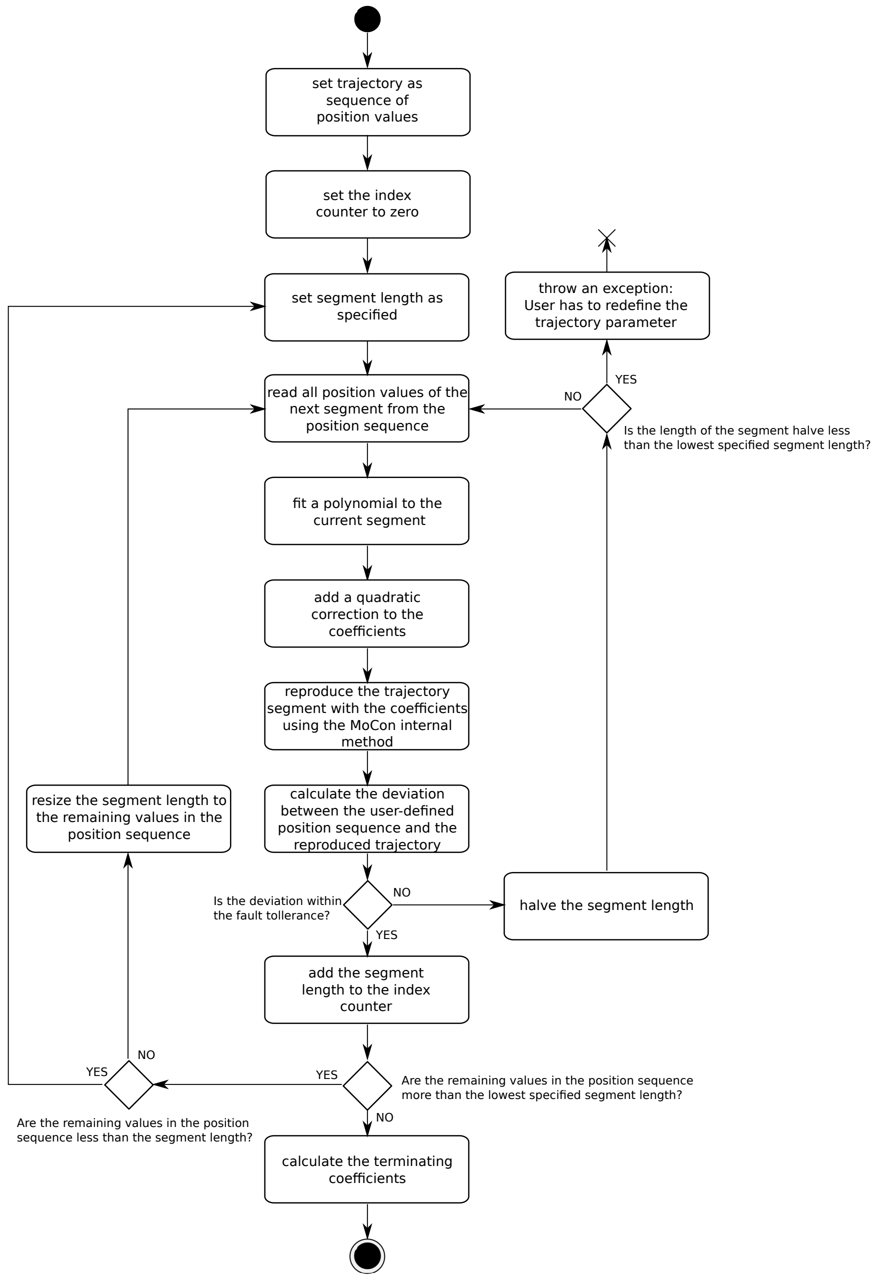

Fig. 2. Activity diagram presenting the algorithm to find the correct coefficients that represent the segments in the user-defined position sequence. Each segment of the user-defined position sequence is polynomial fitted. A quadratic correction is added to the determined coefficients: position, velocity, and acceleration. The coefficients are checked against the user-defined position sequence using the MoCon internal method to reproduce the trajectory. When deviation of the reproduced trajectory from the user-defined position sequence exceed the specified fault tolerance, the segment length is reduced by half and fitted again. If the segment length is shorter than the lowest defined segment length, the algorithm rejects the fitting. The residual amount of position values, that are less than a segment length, are fitted as one segment. Finally the terminating coefficients are calculated to stop the trajectory.

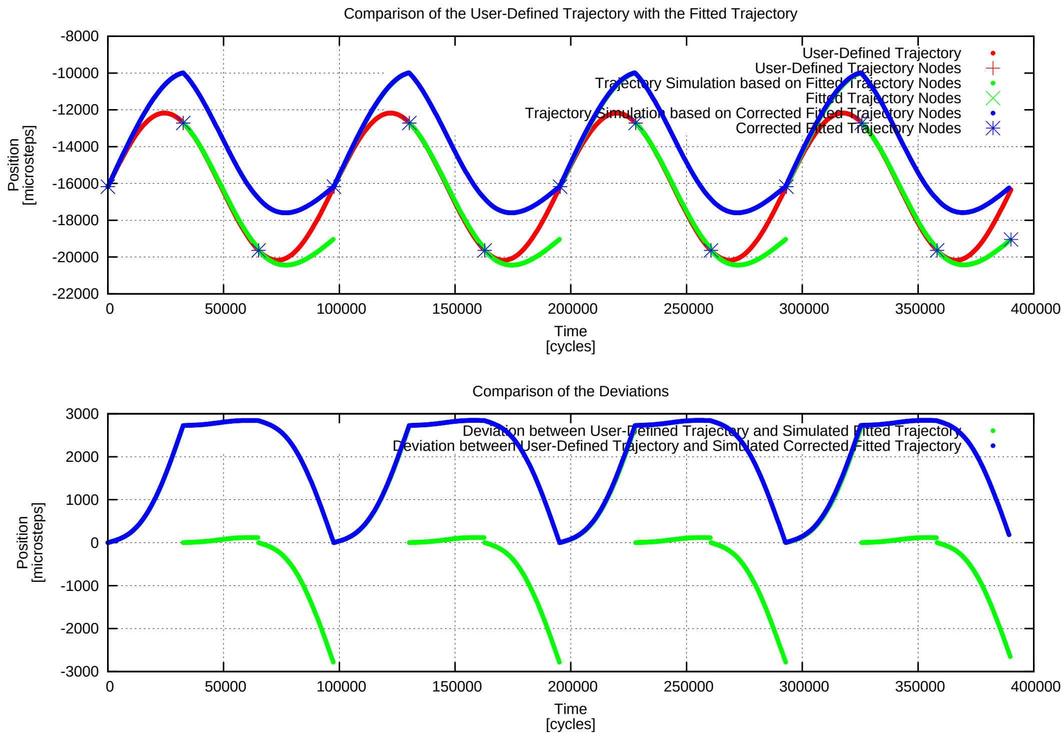

Fig. 3. This plot shows the discontinuity of the fitted trajectory and the shift of the corrected trajectory. The top plot shows the user-defined trajectory (red), the fitted trajectory (green 'x'), and the trajectory reproduced according to the MoCon internal method (green dots). The blue curve is the trajectory, which is corrected only in space. The shift must be compensated by correcting velocity and acceleration coefficients. The difference between the user-defined trajectory and the polynomial fitted trajectory (green dots), as well as the position corrected trajectory (blue dots) is shown in the bottom plot.

```
Position     = PositionCoefficient     + Deviation
Velocity     = VelocityCoefficient - 2 * Deviation / SegmentDuration
Acceleration = AccelerationCoefficient + Deviation / SegmentDuration * SegmentDuration
Jerk         = Jerk
```

Fig. 4. The fitting algorithm uses the quadratic correction in order to produce a continuous trajectory. A simple position correction to the last deviated position of the previous segment would lead to a drift in space and must be corrected.

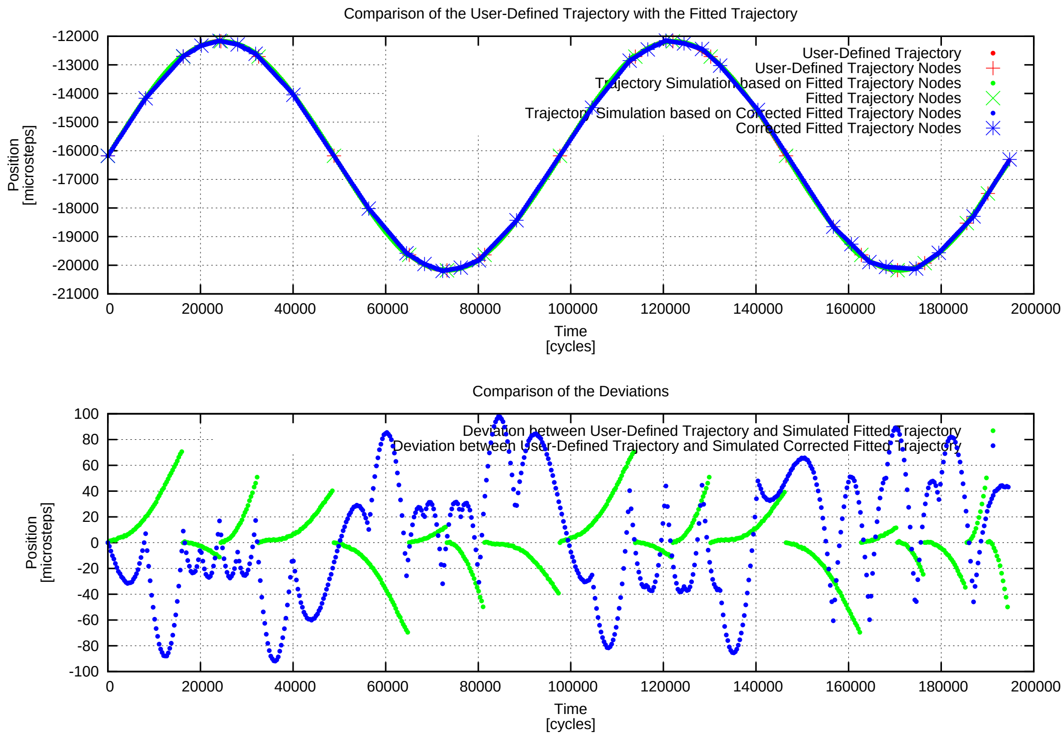

Fig. 5. This plot presents the fitted trajectory with the quadratic correction applied. The trajectory is continuous and each segment follows the user-defined position sequence. The size of the segment lengths of the uncorrected fitted trajectory is constant, while the segment length size of the corrected trajectory is shortened where the moving direction changes. In the bottom plot the deviation of the reproduced trajectory with and without the quadratic correction is compared. The deviation of the corrected trajectory (blue) is almost a smooth line, while the deviation of the fitted trajectory (green) always jumps back to zero when a new segment starts.

5 Test Results

For a validation of the position accuracy of the fitted trajectory, the position of the motor is logged over time and compared with the desired position. The recorded trajectory is accomplished via the trace functionality of the motor controller. The curve plotted in figure 6 is a complete trace of the position of the moving motor. The result of the trace shows that the calculated trajectory is almost identical with the real encoder position. The difference between the commanded MoCon positions and the predicted positions is, as expected, one step; which is caused by rounding errors. The large deviation of the independently measured actual encoder positions oscillates between -3 and 6 microsteps and arises from the inertia of the motor. 8 microsteps are 1 full step. The maximum deviation is 0.75 full steps. The final position of the trajectory is within the user-defined fault tolerance and therefore lies within the expected target range.

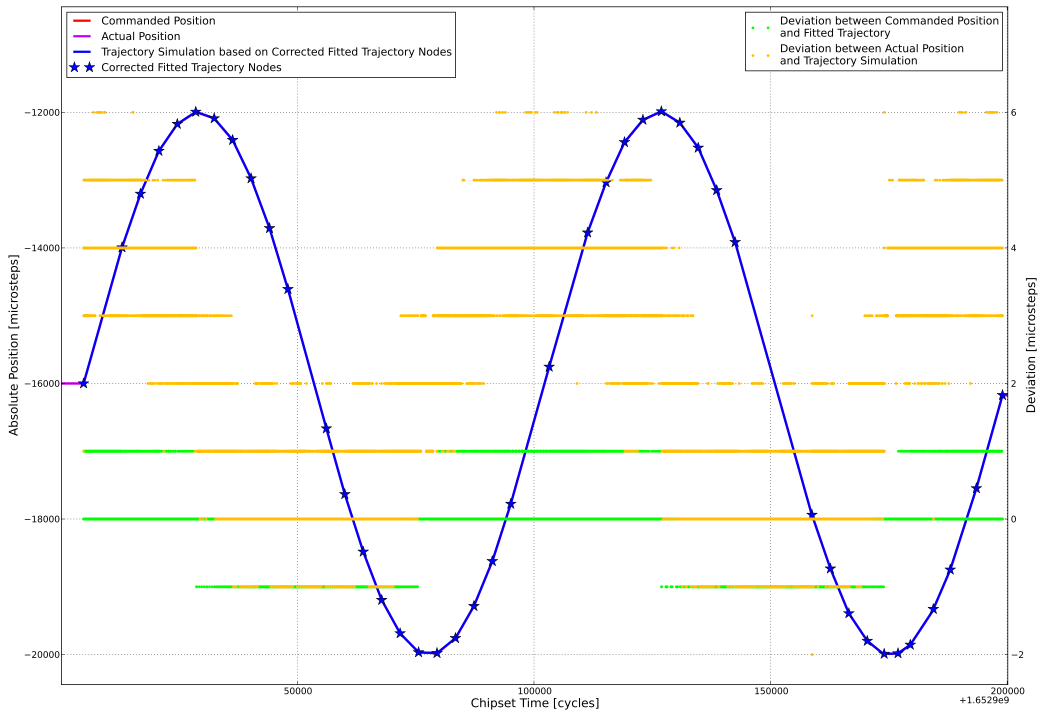

Fig. 6. The commanded position (red line) and the actual position (purple line) of the motor are traced and overlaid with the fitted trajectory (blue line). Based on the corrected polynomial coefficients (blue triangles) representing the start of each segment, the trajectory is predicted according to the MoCon internal method. The commanded MoCon positions deviate from the predicted trajectory by only one microstep (green dots). By comparison, the actual MoCon positions deviates up to 6 microsteps (orange dots).

References

1. Herbst, T. *Optical and Infrared Interferometry II* (Danchi, William C.; Delplancke, Françoise; Rajagopal, Jayadev K 2010) 773407-773407-7
2. Bertram, T. *Adaptive Optics Systems II* (Ellerbroek, Brent L.; Hart, Michael; Hubin, Norbert; Wizinowich, Peter L 2010) 77361S-77361S-10
3. Berwein, J., *411, Astronomical Data Analysis Software and Systems XVIII* (Bohlender, David A.; Durand, Daniel; Dowler, Patrick , Québec City, QC, Canada 2009) 289
4. Kittmann, F., *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* (Radziwill, Nicole M.; Bridger, Alan 2010) 77402P-77402P-7