universität wien

MASTERARBEIT / MASTER'S THESIS

Titel der Masterarbeit / Title of the Master‘s Thesis

„Guiding and Target Acquisition for the ARIEL Space Mission“

verfasst von / submitted by

Gerald Mösenlechner, BSc

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien, 2021 / Vienna 2021

Studienkennzahl lt. Studienblatt / degree programme code as it appears on the student record sheet:	A 066 861
Studienrichtung lt. Studienblatt / degree programme as it appears on the student record sheet:	Masterstudium Astronomie
Betreut von / Supervisor:	Univ.Prof. Mag. Dr. Franz Kerschbaum
Mitbetreut von / Co-Supervisor:	Mag. Dr. Roland Ottensamer

Contents

Acronyms

AIRS	ARIEL Infrared Spectrometer 3–6, 8
AKE	Absolute Knowledge Error 13, 14, 59, 61
AOCS	Altitude and Orbit Control System 7, 14, 27, 68
APE	Absolute Performance Error 13, 14, 55, 59, 61
ARIEL	Atmospheric Remote-sensing Exoplanet Large-survey vi, vii, 1–3, 5, 7, 8, 10, 23, 27, 28, 30, 34, 35, 39, 44, 45, 58, 64, 67, 68
CCD	Charge-Coupled Device 17
CEE	Centroid Estimation Error 14, 21, 29, 37, 38, 44–46, 48–55, 58, 60, 63
CFEE	Cold Front End Electronics 3
CoG	Center of Gravity 15–17, 29, 31, 62, 64
CPU	Central Processing Unit 4, 6
CR	Cramer-Rao 21
DCU	Detector Control Unit 4, 5
DPU	Data Processing Unit 4, 6, 44
FCU	FGS Control Unit 6, 41, 44, 45, 58, 59
FGS	Fine Guidance Sensor 3, 5, 6, 10, 27, 28, 30, 34, 35, 42, 43, 53, 58, 67, 68
FOV	Field of View 11, 33
FPA	Focal Plane Assembly 3, 4
FPGA	Field-programmable Gate Array 4
FWHM	Full Width Half Maximum 27, 29, 50
HK	Housekeeping 4
IASW	Instrument Application Software 4, 6, 23
ICU	Instrument Control Unit 4, 6
IWC	Intensity Weighted Centroiding 16, 29, 31, 50, 54, 56, 58, 59, 64, 65, 67
IWCoG	Iterative Weighted Center of Gravity 16–18, 27–30, 34, 55, 58, 64, 66, 67
KDE	Knowledge Drift Error 13, 14
KRE	Knowledge Reproducibility Error 14
MKE	Mean Knowledge Error 13, 14
MPE	Mean Performance Error 13, 14
NIRSpec	Near Infrared Spectrometer 5, 8, 68
PDE	Performance Drift Error 14
PEC	Pointing Error Contributor 12

PES	Pointing Error Sources 11, 12
PRE	Performance Reproducibility Error 14
PSF	Point Spread Function 15, 16, 21, 28–31, 35, 37, 39, 41–44, 46, 47, 50–53, 58–60
PSU	Power Supply Unit 6
PUS	Packet Utilisation Standard 6
RKE	Relative Knowledge Error 13
ROI	Region of Interest 16, 17, 29, 33, 34, 39, 44, 50, 53, 56, 59, 60, 62, 64–67
RPE	Relative Performance Error 13, 47, 50
SCA	Sensor Chip Assembly 3
VISPhot	Visible Photometer 5, 8
WCoG	Weighted Center of Gravity 16, 64

Abstract English

The Atmospheric Remote-sensing Exoplanet Large-survey (ARIEL) is the most recent addition to the Cosmic Visions program of ESA as its M4 slot. ARIEL is designed to perform high resolution, multi-wavelength spectroscopy of known transiting exoplanets. These measurements are carried out over long time periods and demand a high level of photometric stability. These observations will allow the study of the chemical composition of exoplanetary atmospheres and their thermal structure. This unique use-case of the mission demands high precision of the telescopes' pointing in order to reduce noise.

This warrants the implementation of an instrument that is dedicated to both scientific measurements and the guiding task, the Fine Guidance Sensor. The ARIEL FGS combines three photometric channels ranging from 0.6 µm to 1.2 µm and a spectrometer. Two of the photometric channels form the Fine Guidance System while the other photometric channel and the spectrometer are used to provide additional scientific capabilities. The two guiding channels operate in the range of 0.6-0.8 µm (Channel 1) and 0.8-1.1 µm (Channel 2) and form a redundant system.
The channels are used to take images, which are then transferred to the Detector Control Unit (DCU). These are then sent to the Data Processing Unit (DPU) of the FGS, the FGS Control Unit (FCU). The FCU runs the Instrument Application Software (IASW) that provides the instrument with a wide range of functionality. The IASWs two main operation modes are Tracking and Acquisition. Both of these modes measure the position of the target star using a centroiding algorithm.

In the mode Tracking, the IASW samples the position at a rate of 8Hz or 10Hz on a small Filed of View (FoV). This positional data is then given to the Altitude and Orbit Control System (AOCS) and provides a closed loop that corrects the pointing of the telescope during operation.
The mode Acquisition is used to obtain the initial position of the target star after the rough pointing of the telescope is achieved by the star trackers. In addition to the positional measurement, it is also analysed, whether or not the acquired star is the given target.
This thesis covers the development and performance testing of the algorithms used by the IASW. A wide range of algorithms has been studied for their viability in regard to processing time, precision and stability. A set of test cases has been created using the StarSim simulator that has already been used in the context of the Cheops space mission. Lastly, this thesis describes the methods implemented as a quality index for the measurements and various correction processes.

Abstract German

Das Atmospheric Remote-sensing infrared Exoplanet Large-Survey, ARIEL, ist die neuste Ergänzung des Cosmic Visions Programms der ESA in Form des M4 Slots. ARIEL ist designed um hochaufgelöste, multi-wellenlängen Spektroskopie von bekannten Transit-Exoplaneten durchzuführen. Diese Messungen werden über lange Zeitspannen aufgenommen und verlangen einen hohen Grad an photometrischer Stabilität. Diese Beobachtungen dienen dem Studium der chemischen Zusamensetzung von Exoplaneten-Atmosphären und deren thermische Struktur. Dieser einzigartige Anwendungsfall benötigt eine hoch-präzise Teleskopnachfürung, um Rauschen zu vermindern.

Dies führt zur Implementation eines Instruments, das sowohl für wissenschaftliche Zwecke als auch der Nachführung gewidmet ist, dem Fine Guidance Sensor (FGS). Der ARIEL FGS kombiniert drei photometrische Kanäle mit einer Bandbreite von 0.6 µm bis 1.2 µm und einem Spektrometer. Zwei der photometrischen Kanäle bilden das Fine Guidance System während der andere photometrische Kanal und der Spektrograph als zusätzliche wissenschaftliche Ressourcen dienen. Die zwei Nachführungs-Kanäle decken 0.6-0.8 µm (Kanal 1) und 0.8-1.1 µm (Kanal 2) ab und bilden ein redundantes System.
Diese Kanäle nehmen Bilder auf, welche and die Detector Control Unit (DCU) übertragen werden. Diese werden dann and die Data Processing Unit (DPU) des FGS, die FGS Control Unit (FCU), geschickt. Die FCU dient als Hardware für die Instrument Application Software (IASW) welche dem Instrument eine Bandbreite von Funktionen zur Verfügung stellt. Die zwei Hauptmodi der IASWs sind Tracking und Acquisition. Beide dieser Modi messen die Position des Ziel Sterns mithilfe eines Zentrierungsalgorithmus

Im Modus Tracking misst die IASW die Position mit einer Rate von 8Hz oder 10Hz mit einem kleinen Bildfeld. Die Positionsdaten werden dann an das Altitude and Orbit Control System (AOCS) gegben und bilden einen Regelkreis für die Ausrichtungskorrektur des Teleskops während Beobachtungen.
Der Modus Acquisition wird verwendet um die initiale Position des Zielsterns zu finden, nachdem das Telesekop grob mithilfe der Star Tracker ausgreichtet wurde. Zusätzlich zu den Positionsmessungen wird auch analysiert, ob der gefundene Stern das gewünschte Ziel ist.
Diese Masterarbeit behandelt die Entwicklung und das Testen der Algorithmen, welche die IASW verwendet. Eine Vielzahl von Algorithmen wurden auf deren Verwendbarkeit in Hinsicht auf Rechenzeit, Genauigkeit und Stabilität untersucht. Für die Messung der Genauigkeit der Algorithmen wurden Testfälle mithilfe des StarSim Simulators erstellt,

welcher bereits bei der Cheops Weltraummission Anwendung fand. Letztens behandelt diese Arbeit die Methoden, die als Qualitätsindex der Messungen verwendet werden, sowie diverse Korrekturprozesse.

Acknowledgements

First and foremost I want to thank my family for supporting me throughout my life. I probably wouldn't be able to follow through with my ambitions without their continuous support.
Secondly, I want to thank my supervisors Franz Kerschbaum and Roland Ottensamer for providing me with the opportunity to contribute to such an extraordinary project and for the input they have given me over the last years.
Lastly I want to thank both my friends and colleagues David, Markus, Johannes, Dominik and Armin for the great discussions we had and their feedback to my solutions for various problems.
This work was funded by FFG ASAP project 25046019.

1. Introduction

During the course of the last decades, a staggering amount of diverse extrasolar planets have been discovered and brought a variety of new questions with them. These planets range from rocky Earth-like planets to massive gas giants, where either have been observed with different masses, radii and orbits. One of the major points of interest in regard to these exoplanets is their formation and evolution.

By looking at the bulk chemical composition of exoplanetary atmospheres, one can gain valuable insight about these processes. However, in order to gain any significant information, an extensive survey of different planets orbiting various stellar types is needed to properly study the relation.

The Atmospheric Remote-Sensing Infrared Exoplanet Large Survey (ARIEL) aims to perform these needed observations. ARIEL is the latest addition to ESA's Cosmic Visions program and is equipped with two instruments operating in the near infrared.

The instrumentation of ARIEL is specifically designed to allow spectral and photometric observations of exoplantes over extended periods of time. Given the long-term nature of these observations, noise can be detrimental to the resulting information.

One of the main sources of noise in the spectroscopic time-series measurements is the movement of the target star inside of the aperture of the spectrometer (Ottensamer et al., 2014). In order to mitigate this, ARIEL needs to maintain high precision pointing over extended periods of time.

This will be achieved with the use of the Fine Guidance Sensor. The FGS is an instrument dedicated to the measurement and correction of the telescopes' pointing. These measurements are taken with the help of a centroiding algorithm, which is the main focus of this thesis.

1.1 The ARIEL Space Mission

ARIEL was chosen as the fourth medium size mission of ESAs' Cosmic Visions program in 2018. ARIEL is the spiritual successor of the ECHO space mission and is a collaboration of more than 50 institutes from 12 countries (Figure 1.1). The mission is planned to start in 2029 and is designed to answer questions about the still widely unknown nature of the formation of exoplanets.

ARIEL will perform transit- and eclipse-spectroscopy of a large sample of known transiting exoplanets over its lifetime of four years. In this time, the mission will observe a large range of different types of exoplanets, covering everything from earth-sized planets

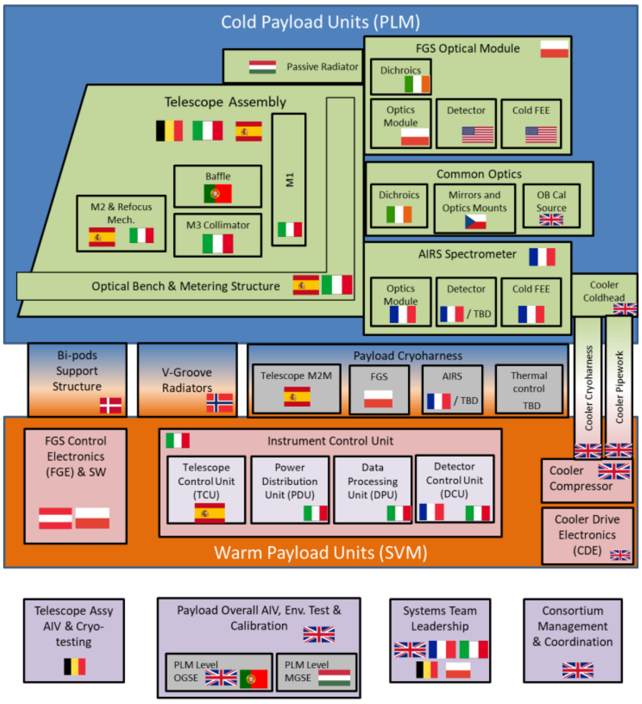


Figure 1.1: Structure of payload contributors of the ARIEL Space Mission (Eccleston, 2017).

up to gas giants around various types of host stars. The target sample itself contains predominantly hot and warm exoplanets. The reason behind this is their well mixed atmosphere with minimal condensation, which helps in measuring their bulk and elemental composition. This enables the study of the earlier stages of planetary formation during the nebular phase (Tinetti et al., 2017).

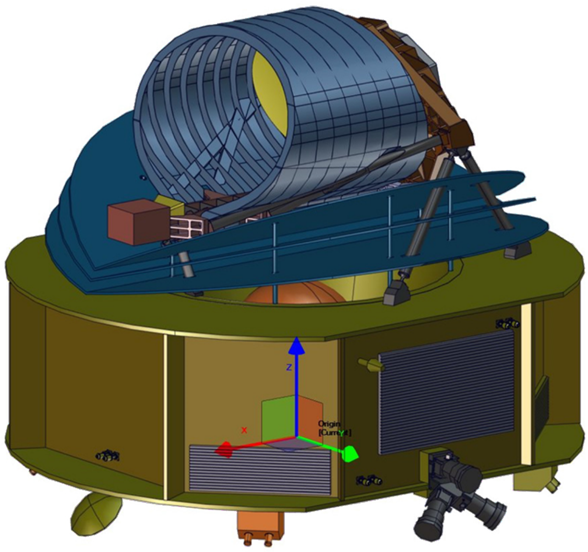


Figure 1.2: CAD Model of the ARIEL Space Mission (Eccleston, 2017).

1.1.1 The Instrumentation of ARIEL

In order for ARIEL to carry out its scientific purpose, it is equipped with a 1.1 m × 0.7 m elliptical mirror that feeds two specialized instruments: the ARIEL Infrared Spectrometer (AIRS) and the Fine Guidance Sensor (FGS). Of these two instruments, AIRS will be the main scientific payload, while the FGS serves both a scientific and an utilitarian purpose.

1.1.1.1 The ARIEL Infrared Spectrometer (AIRS)

AIRS is the main scientific instrument of ARIEL. As the name implies, AIRS is a low/medium resolution (R=30-200) spectrometer operating in the infrared. AIRS covers the spectral range from 1.95 µm to 7.8 µm, which is split into the two channels CH0 (1.95-3.90 µm) and CH1 (3.90-7.80 µm).
The optical layout of each channel consists of a slit that forms the input aperture, a collimator, a folding mirror, the dispersive element, a camera and the detector (Figure 1.3). The folding mirror is introduced, in order to position the entrance- and exit plane of both channels at the same distance. This allows for better utilisation of the instruments' volume.

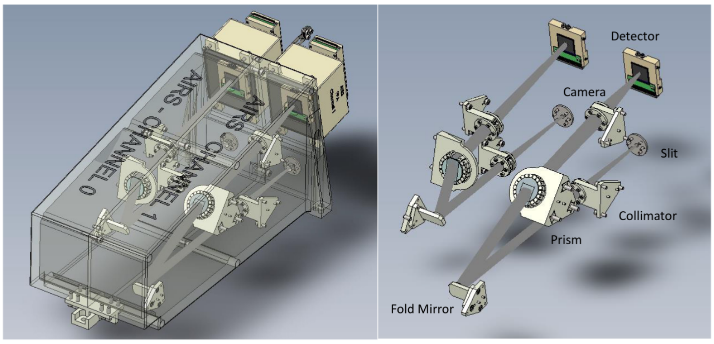


Figure 1.3: CAD Model of the AIRS instrument showing the overall construction and the optical path for both channels (Eccleston, 2017).

The dispersive element is implemented as a prism made from CaF2, which was already applied in previous missions operating in the infrared. This material is also used in the collimator and the camera, where it is used as one component of achromatic doublet systems. These are introduced in order to mitigate the effects of chromatic aberration (ARIEL Science Study Team, 2017).
The baseline detector proposed for each channel is a Teledyne H1RG infrared detector. Both detectors posses their own Focal Plane Assembly (FPA) consisting of the harness used to mount the detector (the Sensor Chip Assembly (SCA)) and the Cold Front End Electronics (CFEE), which is also provided by Teledyne.

The electrical layout of AIRS is depicted in Figure 1.4. The readout of the detectors is handled by their Detector Control Unit (DCU), a part of the Instrument Control Unit (ICU). The DCU uses a Microsemi ProASIC3-type Field-programmable Gate Array (FPGA) which offers the possibility of pre-processing of the collected science data. In addition, the DCU provides Housekeeping (HK) data of the FPA (voltages, temperatures, etc.).

Further processing of the gathered science data is handled by the Data Processing Unit (DPU). The AIRS ICU is equipped with two DPUs operating in cold redundancy. Each DPU board possesses a LEON3FT Central Processing Unit (CPU) operating at 66 MHz or 100 MHz and a co-processing FPGA. The memory of the DPU consists of memory used for booting (PROM), hosting the Instrument Application Software (IASW) (E2PROM and/or NVM), data buffering (SDRAM) and for data processing (SRAM) (Eccleston, 2017).

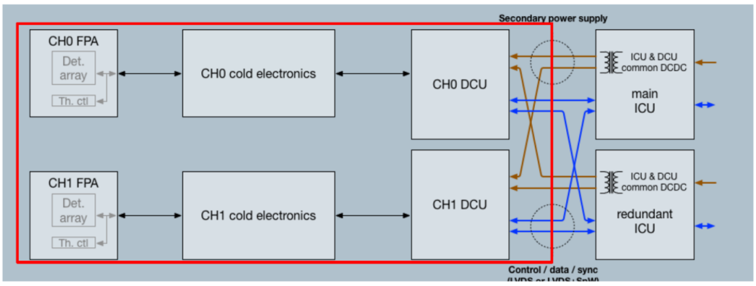


Figure 1.4: Block diagram of the AIRS ICU (Tinetti et al., 2021).

1.1.1.2 The Fine Guidance Sensor (FGS)

The FGS of ARIEL is an instrument that carries out both scientific tasks and utility tasks such as centering, focusing and guiding of the target star. The FGS possesses four channels for different purposes, that cover a total spectral bandwidth ranging from 0.5 µm up to 1.95 µm.

The channels consist of three photometric channels and one spectrometer. The Near Infrared Spectrometer (NIRSpec) is one of the two scientific channels of the FGS and provides low resolution spectroscopy (R=10) in the spectral range of 1.25-1.95 µm (Tinetti et al., 2021). NIRSpec is used to observe the scattered light of exoplanetary atmospheres during transits.

The other scientific channel of the FGS is the Visible Photometer (VISPhot), a photometer operating in the range of 500-550 nm. This photometer provides additional information of the transit depth of the exoplanet and helps to further constrain the chemical composition of the atmosphere.

In order to distribute the light entering the FGS among the four channels, three dichroic beam-splitters are used. In addition, a band pass filter is situated in front of VISPhot, in order to further restrain the wavelength.

The other two photometric channels form the redundant fine guiding system. These two channels cover the ranges from 0.8 µm to 1 µm (FGS1) and 1 µm to 1.2 µm (FGS2).

The detectors proposed for the use in the FGS are Teledyne H2RG, which are also applied in future NASA mission such as the James Webb Space Telescope (JWST). The technical specification of this type of detector is given in Table 1.1.

The layout of the FGSs' electronics is similar to that of AIRS. Once again, detector readout is performed by the DCU. In the case of the FGS, the pre-processing performed by the DCU consists of creating the image cutouts that form the guiding- and science windows.

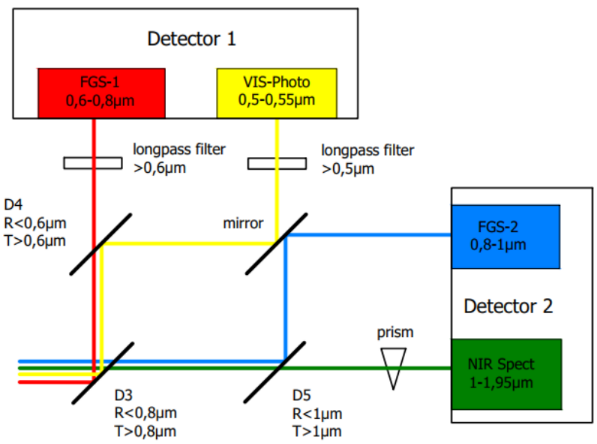


Figure 1.5: Detector layout of the ARIEL FGS. Note that the shown wavelengths are out of date (Eccleston, 2017).

Table 1.1: Technical specifications for the Teledyne H2RG detector taken from (Eccleston, 2017).

Parameter	Value
Detector Size	2000×2000 px
Pixel Size	15×15 m
Readout Mode	Non-destructive
Readout Noise	≤20 e$^-$ rms
Dark current	≤1 e$^-$/s/pixel
Quantum efficiency @ 0.5 µm	0.58
Maximum Power consumption	≤10 mW
Integration Time	0<t≤1000 s
Pixel response non-uniformity	<5%(sigma/mean)
Non-linearity	<3% variation from a linear response

Further processing is then performed by the FGS Control Unit (FCU), which consists of two DPUs and two Power Supply Unit (PSU)s that work in cold redundancy based on the block diagram shown in Figure 1.6. The main element of the FGSs' DPU is an Aeroflex GR712RC chip that uses a LEON3FT CPU clocked at 40 MHz. The memory layout is comparable to that of the AIRS ICU (Rataj et al., 2019).

One of the main tasks of the FCU is running the IASW. The IASW provides the FGS with multiple functions such as science data processing, closed loop pointing correction (Guiding), target acquisition, image acquisition and other general utilities like communication with the spacecraft. These functionalities are implemented in the form of ECSS service commands using the Packet Utilisation Standard (PUS). The main focus of this thesis covers the two IASW operation modes of Tracking and Acquisition.

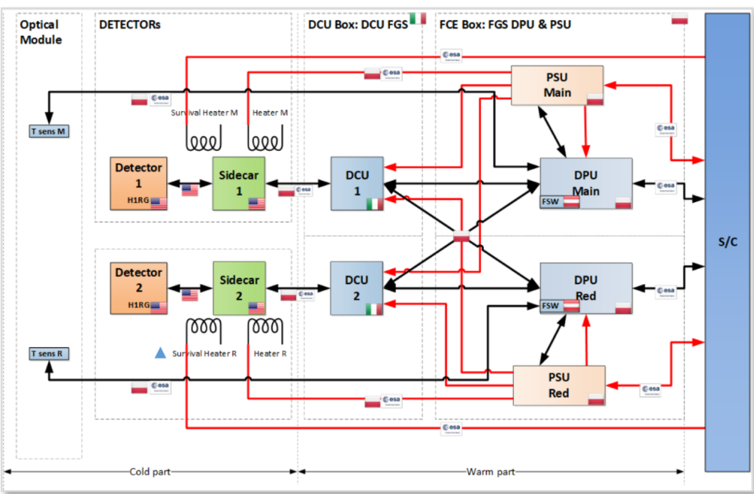


Figure 1.6: Block diagram of the FCU (Tinetti et al., 2021).

Here, the mode Tracking is used to provide the aforementioned functionality of closed loop pointing correction. The positional data of the target star is sampled at a rate of either 8 Hz or 10 Hz. The measured position is then used as an input to the Altitude and Orbit Control System (AOCS) for the correction of the telescopes' orientation. The fast sampling rate limits the integration time of the imagette to 125 ms or 100 ms.
The required precision of the measurement itself is dictated by the intensity of the target star. The faintest target ARIEL will observe is given as GJ 1214, a M5V star with a K-band magnitude of 8.8. For these stars, the mode Tracking shall be able to provide measurements with an absolute error of less than 150 mas. On the other hand the brightest target is HD 219134, a K3V star with a K-band magnitude of 3.25. Here, the error shall be below 20 mas.

The mode Acquisition is used for the initial positional measurement and target verification after the transition from the Star trackers. The highest achievable sampling rate for this mode shall be 1 Hz with an image integration time of 100 ms or lower. Here, the measurement must be performed on a much larger field of view of 24×24 arcsec. The absolute measurement error for the mode should be lower than 200 mas for both faint and bright targets. A more detailed description of the performance requirements for both operational modes is given in section 2.1.

1.1.2 Observational methods

Observing the atmosphere of exoplantes is a complex task that requires specific strategies. At the time of writing this thesis, there are three main methods that are used to study exo-atmospheres. They consist of measurements of the planet during its transit, the measurement of the system during the planets' secondary eclipse and the measurement of the phase curve during the orbital period of the planet.
In addition to these three main methods, ARIEL will also apply the two complementary methods of eclipse mapping (a subcategory of eclipse measurements) and the time-series measurement of narrow spectral bands.
In the following sections, a brief overview of each of these methods is given.

1.1.2.1 Transit Measurements

One of the most common methods to observe exoplanets is the flux reduction caused by the transiting planet covering up the star. The reduction is correspondent to the projected area ratio between star and planet and can therefore be used to determine the radius of the planet (Sing, 2018).

$$\frac{\Delta f}{f} \approx \left(\frac{R_{pl}}{R_{star}}\right)^2 \tag{1.1}$$

The reduction can be wavelength dependent since certain atomic of molecular species that are present in the planetary atmosphere will also absorb light (Tinetti et al. (2007)). This will result in a larger apparent radius of the transiting planet and therefore an increase in transit depth (Figure 1.7) (Zellem et al., 2019). By combining both photometric transit measurements using VISPhot and transit spectroscopy using AIRS and NIRSpec, ARIEL is able to obtain information about the chemical composition of the planetary atmosphere.

1.1.2.2 Measurements of the Eclipse

During the secondary eclipse of the planet, where the host star moves in front of the exoplanet, flux from the star is reflected by the planets' surface. The resulting loss of light after the complete obstruction of the planet is referred to as eclipse depth, which can be described by the following relation:

$$\frac{\Delta f}{f} \approx \frac{F_{pl}}{F_{star}} \times \left(\frac{R_{pl}}{R_{star}}\right)^2 \tag{1.2}$$

where $\Delta f/f$ is the relative light-loss, F is the respective flux of the planet and star

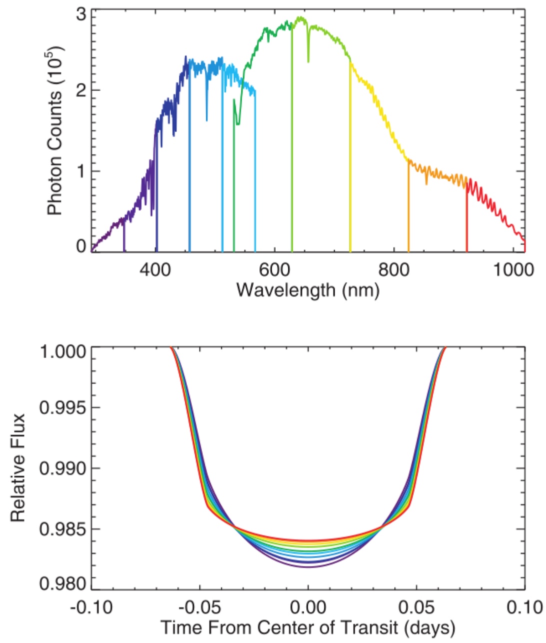


Figure 1.7: Top: two spectral measurements for G430L (293 to 567 nm) and G750L (532 to 1019 nm) Bottom: light curves for the 10 bandpasses denoted by the vertical lines in the top figure (Knutson et al., 2007).

and R is the radius of the object (Sing, 2018). This reflected quantity of light can be measured by performing time series observations of the system before, during and after the secondary eclipse. Given that the reflected light passes through parts of the planetary atmosphere, one can derive spectra of emission- and absorption features.
These features also provide clues to the temperature structure of the atmosphere. For instance, if the atmosphere has a hot stratosphere where hot gas sits on top of cooler parts of the atmosphere, both emission- and absorption features will be present in the spectra. However, if the outer layer of the atmosphere is dominated by cool gas, only the absorption lines will be present (Charbonneau et al. (2008)).
Another measurement that can be performed during the eclipse is called eclipse mapping. Here, the longitudinal temperature distribution of the planet is reconstructed from multiple observations during the ingress and egress of the planet and a spatially resolved image of the day-side hemisphere can be obtained. This can provide information about thermal structure between day and night-side as well as winds, that distribute the thermal energy across the atmosphere (Majeau et al. (2012)) (Figure 1.8).

1.1.2.3 Phase curve Mapping

While transit- and eclipse observations only focus on short periods of the planets' orbit, phase curve mapping describes the observation of the planet during its entire orbit. This means that high photometric precision is required over an extended timespan which can often reach multiple days. This makes phase curve mapping the most difficult atmo-

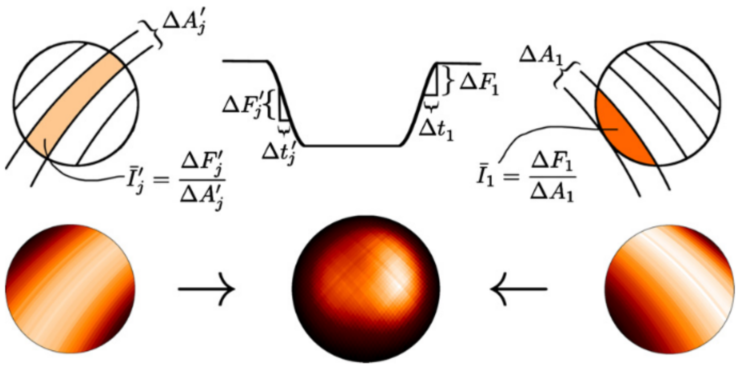


Figure 1.8: Combined map of HD189733b ingress and egress slice maps (Majeau et al., 2012).

spheric measurement to perform on a transiting exoplanet.
Similar to eclipse mapping, the phase curve measurements can be used to study the day-night circulation of the planetary atmosphere by mapping the temperature on the planets' surface. Since the measurements are done over the entirety of the orbit, phase mapping often provides more leverage on the longitudinal brightness distribution, but also suffers from less spatial resolution and higher sensitivity to other brightness variations when compared to eclipse mapping (Majeau et al. (2012)).
One advantage of phase curve observations is that one can also study the eclipse and transit of the planet from the same measurement, as they are also part of the phase curve. Given the required photometric precision, this method is only suited for space telescopes that are designed to provide the necessary pointing precision and for large ground-based telescopes. This is also one of the many reasons, as to why ARIEL is equipped with a dedicated FGS.

1.1.2.4 Narrow band time series

As the name implies, this method revolves around the study of photometric and spectroscopic measurements of narrow bands over extended periods of time. This method was already applied in a study of the cloud thickness variations of the brown dwarfs 2M2139 and SIMP0136 by Apai et al. (2013). In this case, a large quantity of spectra in the 1.1-1.7 µm wavelength range were taken using the Wide-Field Camera 3 of the Hubble Space Telescope.
In the analysis of these measurements, the absorption features of water and methane showed periodic variations that align with the light-curves of the rotation derived from ground-based observations. In addition, the measurements contained periodic components with a higher frequency and wavelength dependent phase shifts.
Based on atmospheric models, it was derived that these variations originate from scale height variations of the clouds present in the atmosphere of the two brown dwarfs. In the case of ARIEL, the method is intended to be used in the same fashion.

1.2 Pointing Errors and Guiding

The problem of finding targets and keeping them in the line of sight is a problem that's as old as the first astronomical observations with telescopes. With the introduction of photographic plates, the simplistic task of following the object while keeping a steady hand evolved into a complex technical problem. Sophisticated gearboxes powered by assistants operating handcranks corrected the movement of the object across the night sky and ensured sharp images of the desired objects.
The necessity of guiding the telescope gained further importance with the rise of electronic detectors which led to the design of the first self correcting guiding systems.
In the present day, guiding is a critical part of high precision measurements. The main focus of guiding is to combat the pointing offsets introduced by a variety of factors during the operation of the telescope. A brief discussion of common error sources, and the description of these errors and algorithms used to determine these errors is given in this section.

1.2.1 Pointing Error

In general, a pointing error is the result of a physical phenomenon that causes the center of the telescopes Field of View (FOV) to shift from its point of origin. These types of phenomena are referred to as Pointing Error Sources (PES). One can differentiate PES based on their behaviour in time and realisation. The main three categories are described as:

- time-constant: The PES does not change over time (Table 1.2).
- time-random: The PES is random in time (Table 1.3).
- ensemble-random: The PES is random in its realization.

Table 1.2: Overview of time-constant PES (T Ott, 2011).

time-constant PES
Misalignments (e.g. payload misalignments)
Calibration uncertainty (e.g. sensor bias)

Table 1.3: Overview of time-random PES (T Ott, 2011).

time-random PES
Environmental disturbances (e.g. solar pressure)
Payload intrinsic error sources (e.g. filter wheels, pumps, etc.)
Drive mechanisms (e.g. solar arrays, instruments, etc.)

time-random PES
Actuator intrinsic error source (e.g. reaction wheel imbalances)
Sensor bias and noise (e.g. star tracker)
Structure thermo-mechanical deformations (e.g. differential heating due to orbit)
System dynamics induces errors (e.g. flexible modes of mechanical components)

The transformation of the shift caused by the PES in regard to the telescope and subsystem reference frame and overall control-system forms a Pointing Error Contributor (PEC). An example for this transfer is given in Figure 1.9. A PEC describes the real contribution of a PES to the pointing performance of the system (T Ott, 2011). The total pointing error of a system is then calculated by the summation of the different PEC.

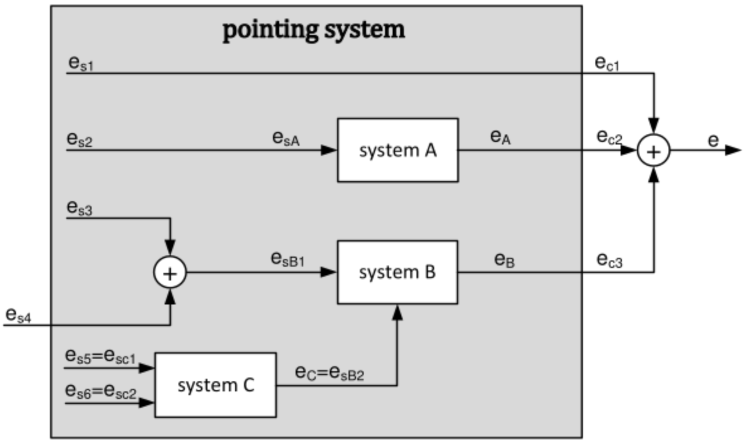


Figure 1.9: PES transfer for a pointing system containing three subsystems. The indexes s and c denote PES and PEC respectively (T Ott, 2011).

The total pointing error is further described by its behaviour over time. This is done via three different time-dependencies:

- Instantaneous Time t: describes the pointing error at any time t of the systems lifetime or an observational period.
- Window Time Δt: pointing error in a given time window Δt at the time t.
- Stability Time Δt_s: relative error between two distinct time windows Δt_1 and Δt_2. These measurements are taken at two different points in time that are Δt_s apart.

Based on this, T Ott (2011) defines ten types of pointing errors that are described in Table 1.4. The bar present in the mathematical formulation indicate the time average of a given error. This is defined by via following equation:

$$\bar{e}(t, \Delta t) = \langle e(t) \rangle_{\Delta t} = \frac{1}{\Delta t} \int_{t-\Delta t/2}^{t+\Delta t/2} e(t) dt \tag{1.3}$$

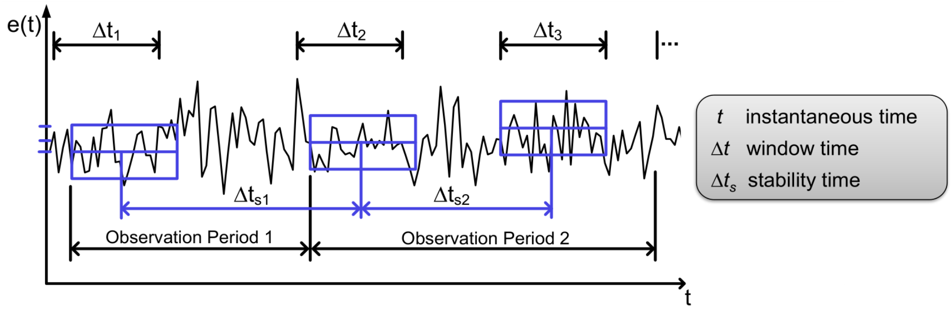


Figure 1.10: Example for the time-dependencies of the total pointing error (T Ott, 2011).

Table 1.4: Overview of different pointing errors. The index K denotes knowledge error signals while P indicates performance errors (T Ott, 2011)

Name	Definition	Mathematical Formulation
Absolute Knowledge Error (AKE)	Difference between the actual parameter and the measured parameter	$e_{AKE} = e_K(t)$
Absolute Performance Error (APE)	Difference between the target parameter and the measured parameter	$e_{APE} = e_P(t)$
Mean Knowledge Error (MKE)	Mean AKE over a time interval Δt	$e_{MKE} = \overline{e_K}(t, \Delta t)$
Mean Performance Error (MPE)	Mean APE over a time interval Δt	$e_{MPE} = \overline{e_P}(t, \Delta t)$
Relative Knowledge Error (RKE)	Difference between the APE at a given time during Δt and the MKE of the same interval	$e_{RKE} = e_K(t) - \overline{e_K}(t, \Delta t)$
Relative Performance Error (RPE)	Difference between the APE at a given time during Δt and the MPE of the same interval	$e_{RPE} = e_P(t) - \overline{e_P}(t, \Delta t)$
Knowledge Drift Error (KDE)	Difference between MKEs taken in two time intervals in the same observation that are separated by Δt_S	$e_{KDE} = \overline{e_K}(t, \Delta t) - \overline{e_K}(t + \Delta t_S, \Delta t)$

Name	Definition	Mathematical Formulation
Performance Drift Error (PDE)	Difference between MPEs taken in two time intervals in the same observation that are separated by Δt_S	$e_{PDE} = \overline{e_P}(t, \Delta t) - \overline{e_P}(t + \Delta t_S, \Delta t)$
Knowledge Reproducibility Error (KRE)	Difference between MKEs taken in two time intervals that are separated by Δt_S in different observations	$e_{KRE} = \overline{e_K}(t, \Delta t_1) - \overline{e_K}(t + \Delta t_S, \Delta t_2)$
Performance Reproducibility Error (PRE)	Difference between MPEs taken in two time intervals that are separated by Δt_S in different observations	$e_{PRE} = \overline{e_P}(t, \Delta t_1) - \overline{e_P}(t + \Delta t_S, \Delta t_2)$

All of the described errors are dependent on the instantaneous time while every error except for the APE and AKE is dependent on the window time. PDE, PRE, KDE and KRE are also dependent on the stability time.

In the context of this thesis, the main error used to describe the performance of the algorithms is the AKE. This is done based on lacking knowledge about the overall behaviour of the AOCS. In general, the errors measured in the simulations presented in this thesis will be represented in the form of the Centroid Estimation Error (CEE), which describes the absolute measurement error as the sum of the AKE in both detector dimensions.

$$e_{CEE} = \sqrt{e^2_{AKE,x} + e^2_{AKE,y}} \tag{1.4}$$

1.2.2 Centroiding Algorithms

In order to properly perform the task of guiding the spacecraft during observations, the satellite must be able to know the position of the target in relation to its line of sight. Therefore, the mission needs to perform positional measurements with repeatable precision in a short amount of time.
The typical way to achieve this is with the use of a centroiding algorithm. A centroiding algorithm enables the measurement of the position of a star with an arbitrary Point Spread Function (PSF) on a 2D image. Centroiding algorithms can be distinguished into three categories:

- Center of Gravity
- Correlation-based Centroiding
- Direct fitting

In the following sections, a brief overview of each family of algorithms will be given.

1.2.2.1 Center of Gravity

The Center of Gravity (CoG) family of centroid algorithms is one of the simplest methods to determine the position of a star inside an image. At its core, the CoG algorithm for centroiding is the same as the method used to determine the center of mass of an arbitrary 2D body, hence its name. In the case of the center of mass determination one would divide the body into sections with the same area and use following equation (Roh et al., 2008):

$$x_c = \frac{\sum_{i=0}^{N}\sum_{j=0}^{M} A\rho_{i,j}x_i}{\sum_{i=0}^{N}\sum_{j=0}^{M} A\rho_{i,j}}, \quad y_c = \frac{\sum_{i=0}^{M}\sum_{j=0}^{N} A\rho_{i,j}y_i}{\sum_{i=0}^{M}\sum_{j=0}^{N} A\rho_{i,j}} \tag{1.5}$$

with A being the size of the area and $\rho_{i,j}$ being the average surface density inside of the current area element. x_i and y_i denote the distance of the current element to a predefined point of origin and N and M are the number of elements in each dimension.

In the case of the centroid estimation of a star, the segments are predetermined by the pixels and the mass of each segment is replaced with the measured intensity $I_{i,j}$ inside of the pixel. This results in following formula:

$$x_c = \frac{\sum_{i=0}^{N}\sum_{j=0}^{M} I_{i,j}x_i}{\sum_{i=0}^{N}\sum_{j=0}^{M} I_{i,j}}, \quad y_c = \frac{\sum_{i=0}^{M}\sum_{j=0}^{N} I_{i,j}y_i}{\sum_{i=0}^{M}\sum_{j=0}^{N} I_{i,j}} \tag{1.6}$$

Formula 1.6 is the basis for all consequent versions of CoG. Due to its simplicity, this basic algorithm excels in terms of processing time and hardware requirements, which are major advantages in space application.

In the initial evaluation of centroiding algorithms for the Echo Space Mission, Ottensamer et al. (2014) state the significant impact of background noise on the centroids precision. Even though, on can improve this behaviour by using thresholds on the image, this still makes the base CoG method unsuited for highly demanding applications.
For this reason, the image is often multiplied with a weighting function, that is designed to mitigate the effects of noise on the measurement (Vyas et al. (2010); Akondi and Vohnsen (2013)). One can distinguish three different methods to achieve this:

- Intensity Weighted Centroiding (IWC)
- Weighted Center of Gravity (WCoG)
- Iterative Weighted Center of Gravity (IWCoG)

Of these three methods, IWC is the simplest way to improve the noise dependency of the base CoG. In this case, the original image is multiplied with itself and a normal CoG is performed (Formula 1.7). This will result in a better overall precision and stability with hardly any additional processing time, which makes this algorithm suited for the determination of an initial Region of Interest (ROI). (Ottensamer et al., 2014)

$$x_c = \frac{\sum_{i=0}^{N}\sum_{j=0}^{M} I_{i,j}^2 x_i}{\sum_{i=0}^{N}\sum_{j=0}^{M} I_{i,j}^2}, \quad y_c = \frac{\sum_{i=0}^{M}\sum_{j=0}^{N} I_{i,j}^2 y_i}{\sum_{i=0}^{M}\sum_{j=0}^{N} I_{i,j}^2} \tag{1.7}$$

It is to note however, that this version is very sensitive to glitches and hot pixels. This can be negated by the usage of a median filter at the cost of longer processing times.

WCoG has a similar concept but it uses a dedicated weighting function, that is multiplied with the image. This weighting function $\mathrm{W}_{i,j}$ ideally resembles the PSF of the telescope and is often approximated by a normalised Gaussian function.

$$x_c = \frac{\sum_{i=0}^{N}\sum_{j=0}^{M} I_{i,j} W_{i,j} x_i}{\sum_{i=0}^{N}\sum_{j=0}^{M} I_{i,j} W_{i,j}}, \quad y_c = \frac{\sum_{i=0}^{M}\sum_{j=0}^{N} I_{i,j} W_{i,j} y_i}{\sum_{i=0}^{M}\sum_{j=0}^{N} I_{i,j} W_{i,j}} \tag{1.8}$$

It is to note, that an initial estimation of the stars' position is necessary, because of the positioning of the generated weighting function. This can be done with a quick, but less precise centroiding algorithm or from previous acquisition or guiding measurements. One problem is, that this initial guess can have an impact on the achievable precision and the algorithm might not lead to a proper result, if the weighting function and the PSF don't overlap.

The last algorithm, IWCoG, mitigates the precision dependency on the initial guess (Baker and Moallem, 2007). This is achieved by performing multiple iterations of a WCoG algorithm , where the previous result is used as the initial guess for the next run. This makes the precision dependent on iteration count rather than the initial guess.

$$x_c^n = \frac{\sum_{i=0}^{N} \sum_{j=0}^{M} I_{i,j} W_{i,j}^n x_i}{\sum_{i=0}^{N} \sum_{j=0}^{M} I_{i,j} W_{i,j}^n}, \quad y_c^n = \frac{\sum_{i=0}^{M} \sum_{j=0}^{N} I_{i,j} W_{i,j}^n y_i}{\sum_{i=0}^{M} \sum_{j=0}^{N} I_{i,j} W_{i,j}^n} \tag{1.9}$$

Overall, this makes the IWCoG algorithm the highest performing of the CoG family, but also the most resource intensive (Ottensamer et al., 2014).

1.2.2.2 Correlation-based Centroiding

Correlation-based Centroiding describes the process of calculating the position of the star using correlation. This method already finds widespread usage in Shack-Hartmann wavefront sensors, where the algorithm is used to determine the position of the images produced by the lenslet array (Lisa A. Poyneer (2003); Willert and Gharib (1991)).

$$Corr(g,h)(x) = \int_{-\infty}^{\infty} g(y)h^*(y+x)dy \tag{1.10}$$

Formula 1.10 describes the basic correlation calculation for two arbitrary analytical functions g and h. Given that the incoming signal of the Charge-Coupled Device (CCD) is both real and discrete (sampled at a rate dictated by the pixel size), one can further simplify this equation to:

$$Corr(f,I)(x,y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m,n) I^*(x+m, y+n) \tag{1.11}$$

in regard to performance, correlation-based centroiding methods provide favourable noise dependencies and no dependency on an initial guess. However, given the nature of correlation of a discrete function, the algorithm is only able to achieve single pixel precision using the base form.

This issue can be solved in two ways. Firstly, one can take the resulting position and perform an interpolation with the neighbouring pixels to get measurements of sub-pixel precision. This is a quite simple and fast method to alleviate this problem (Lisa A. Poyneer, 2003).

Alternatively, one can increase the precision by upsampling the image before the application of the correlation. The resulting increase of the precision is a function of upsampling factor (Figure 1.11). This makes this method only suited for a smaller ROI since upsampling can be a quite time-intensive operation (Ferstl, 2016).

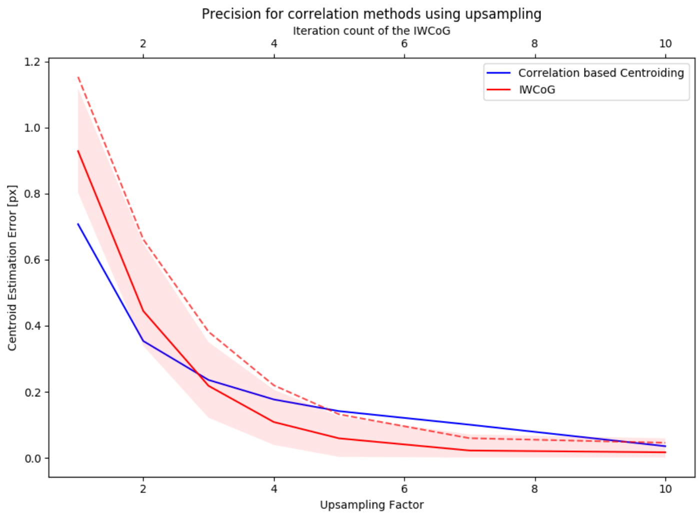


Figure 1.11: Comparison between the maximal performance of correlation based centroiding using upsampling (blue) and a statistical analysis of an IWCoG algorithm using different iteration counts (red). The shaded red area shows the maximum and minimum error of the IWCoG, while the dashed red line represents 3σ.

1.2.2.3 Direct Fitting

The last family of algorithms is that of direct fitting strategies. Here one can differentiate between two methods: Three-Point Gaussian fitting and least squares minimisation. The first method, three-point Gaussian fitting, is rather simplistic in its nature. Based on an initial guess of the stars' position, three equidistant pixels I_{i-1}, I_i and I_{i+1} are chosen for each dimension. The fit for one dimension is then performed using following formula (Tremsin et al., 2003):

$$x_0 = P\frac{lnQ_{i+1} - lnQ_{i-1}}{2[2lnQ_i - lnQ_{i-1} - lnQ_{i+1}]} + iP \tag{1.12}$$

In this formula, P is the fixed distance between the pixels and Q describes a one-dimensional, bell-shaped distribution. The analytical functions tested by Tremsin et al. (2003) are summarised in Table 1.5.

Table 1.5: Bell-shaped distributions tested by Tremsin et al. (2003)

Analytical fitting function	Formula
Gaussian	$Q(x) = \frac{A_1}{\sqrt{2\pi}A_2} exp\left[-\frac{1}{2}\left(\frac{x-A_3}{A_2}\right)^2\right]$
Hyperbolic cosine	$Q(x) = A_1 cosh^{-2}\left[\frac{x-A_3}{A_2}\right]$
Parabolic	$Q(x) = A_1 + A_2(x - A_3)^2$
Lorentzian	$Q(x) = \frac{A_1}{A_2+(x-A_3)^2}$

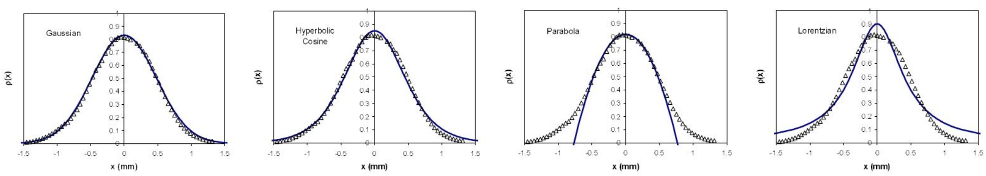


Figure 1.12: Fits of a measured signal using the functions shown in Table 1.5. The triangles represent the measured signal of the source (Tremsin et al., 2003).

It is to note that even though all of these functions will work for the task of centroid estimation, functions that resemble the original signal will provide better precision and stability. Therefore, out of the four functions tested by Tremsin et al. (2003), both a Gaussian and a hyperbolic cosine have performed best.

The other approach to measure the position of a source using direct fitting is to use two-dimensional non-linear least-squares minimization. Least-squares minimization is a common tool to determine coefficients of a function when the number of measured data points greatly surpasses the number of parameters to fit.
Here, an example for an one-dimensional system with N data points (x_i, y_i) with $i = 0, ..., N-1$ and a function $y(x|\vec{a})$ that is dependent of a set of M parameters a_j, $j = 0, ..., M-1$ is given. Using the error σ_i between the measurements and the function using a given set of parameters $\vec{a}$, one can define the merit function $\chi^2(\vec{a})$ as follows (Press et al., 2007, 801-802):

$$\chi^2(\vec{a}) = \sum_{i=0}^{N-1} \left[\frac{y_i - y(x_i|\vec{a})}{\sigma_i} \right]^2 \tag{1.13}$$

This merit function can now be used to find the optimal parameters by finding its minimum. If the initial set of values $\vec{a}_{cur}$ is already close to the minimum, $\chi^2(\vec{a})$ can be approximated using a quadratic form:

$$\chi^2(\vec{a}) \approx \gamma - \vec{d} * \vec{a} + \frac{1}{2} \vec{a} * D * \vec{a} \tag{1.14}$$

Here, D represents the Hessian matrix of the merit function with the shape M×M and $\vec{d}$ denotes a M-vector. Given that the initial set of parameters is a close approximation, one can then use the following equation to directly jump to the minimum of the merit function:

$$\vec{a}_{min} = \vec{a}_{cur} + \vec{D}^{-1} * [-\nabla \chi^2(\vec{a}_{cur})] \tag{1.15}$$

In most cases, Equation 1.15 isn't applicable, as the initial "guess" of the parameters can often be a poor approximation. In this case, instead of taking the entire "distance" between the current value of $\chi^2(\vec{a})$ and its minimum in a single leap, one can take multiple steps along the downward gradient using the steepest decent method.

$$\vec{a}_{next} = \vec{a}_{cur} - constant \times \nabla\chi^2(\vec{a}_{cur}) \tag{1.16}$$

It is to note, that the constant should be small enough to not exhaust the downhill direction.

In order to use both of these equations, the gradient of χ^2 needs to be calculated and the Hessian matrix for any given set of parameters $\vec{a}$. The first step for this is to minimise Function 1.13. For this, one has to form the first and second partial derivative:

$$\frac{\partial\chi^2}{\partial a_k} = -2\sum_{i=0}^{N-1}\frac{[y_i - y(x_i|\vec{a})]}{\sigma_i}\frac{\partial y(x_i|\vec{a})}{\partial a_k} \qquad k = 0, 1, ..., M-1 \tag{1.17}$$

$$\frac{\partial^2\chi^2}{\partial a_k\partial a_l} = 2\sum_{i=0}^{N-1}\frac{1}{\sigma_i^2}\left[\frac{\partial y(x_i|\vec{a})}{\partial a_k}\frac{\partial y(x_i|\vec{a})}{\partial a_l} - [y_i - y(x_i|\vec{a})]\frac{\partial^2 y(x_i|\vec{a})}{\partial a_k\partial a_l}\right]. \tag{1.18}$$

It is now conventional to remove the factor of 2 by defining following variables:

$$\beta_k \equiv -\frac{1}{2}\frac{\partial\chi^2}{\partial a_k} \qquad \alpha_{kl} \equiv \frac{1}{2}\frac{\partial^2\chi^2}{\partial a_k\partial a_l} \tag{1.19}$$

With this, $\vec{\alpha}$, which is usually called curvature matrix, is defined as $\frac{1}{2}\vec{D}$. This enables us to rewrite Equation 1.15 as a set of linear equations:

$$\sum_{l=0}^{M-1}\alpha_{kl}\delta a_l = \beta_k \tag{1.20}$$

Furthermore, Equation 1.16 transforms to:

$$\delta a_l = constant \times \beta_l \tag{1.21}$$

In this notation, δa_l represents an increment, that is added to the current set of parameters in order to obtain the next approximation. The components of the curvature matrix now depend on both the first and second derivative. However, the second derivative can be ignored, since it is multiplied with the term $[y_i - y(x_i|\vec{a})]$ in Equation 1.18. In order for the model to succeed, this term has to be a random measurement error for each point and will cancel out if one sums over i. Furthermore, it's favorable to discard the second derivative, since it can lead to instabilities for badly fitted models or outliers that won't be compensated. With this, Equation 1.18 turns into the following formula:

$$\alpha_{kl} = \sum_{i=0}^{N-1}\frac{1}{\sigma_i^2}\left[\frac{\partial y(x_i|\vec{a})}{\partial a_k}\frac{\partial y(x_i|\vec{a})}{\partial a_l}\right]. \tag{1.22}$$

The most commonly used algorithm in this category is the Levenberg-Marquart method. This algorithm is a combination of the inverse-Hessian- and the steepest descent method.

1.2.3 Cramer-Rao-Bounds

In order to properly quantify the performance of the algorithms developed in this thesis, the Cramer-Rao (CR) limit is used as reference. The CR Limit is the theoretical lower bound for the variance of unbiased estimators. This makes the CR limit well suited as a baseline for centroiding algorithms and other astrometric procedures, as most algorithms for positional estimation fall under the category of unbiased estimators.
Winick (1986) described the CR limit for the use on positional estimators for the discrete, two dimensional case with a Gaussian PSF. According to this, the lower bound is calculated as follows:

$$E[(\hat{\epsilon_x} - \epsilon_x)^2] \geq \left[\lambda_S \left[\sum_{ij} \frac{[g_i'(\epsilon_x) g_j(\epsilon_y)]^2}{g_i(\epsilon_x) g_j(\epsilon_y) + \lambda_N/\lambda_S} - \frac{\left[\sum_{ij} \frac{g_i'(\epsilon_x) g_j(\epsilon_y) g_i(\epsilon_x) g_j'(\epsilon_y)}{g_i(\epsilon_x) g_j(\epsilon_y) + \lambda_N/\lambda_S} \right]^2}{\sum_{ij} \frac{g_i(\epsilon_x) g_j'(\epsilon_y)}{g_i(\epsilon_x) g_j(\epsilon_y) + \lambda_N/\lambda_S}} \right] \right]^{-1} \tag{1.23}$$

with

$$g_i(\epsilon_x) = \int_{x_i - \frac{\delta x}{2}}^{x_i + \frac{\delta x}{2}} S(x, \epsilon_x) dx \tag{1.24}$$

$$g_j(\epsilon_y) = \int_{y_i - \frac{\delta y}{2}}^{y_i + \frac{\delta y}{2}} S(y, \epsilon_y) dy, \tag{1.25}$$

where S denotes a one-dimensional Gaussian function with its center at ϵ. λ_S is the number of photons detected by the CCD, λ_N represents the average electron count per pixel from background and read-out noise and δx and δy is the physical size of the pixel. In the context of this thesis, the CR limit is used as a point of reference for the following performance tests and is represented as a solid black line in the performance figures. Given that equation 1.23 is formulated for a Gaussian PSF, a Gaussian distribution with an equivalent size to the real PSF was used for this purpose. Furthermore, this equation represents the minimum square error of $\hat{\epsilon}$. Because of the rotational symmetry, the result can be expressed as a CEE by

$$CEE = \sqrt{2} E[(\hat{\epsilon_x} - \epsilon_x)] \tag{1.26}$$

2. Algorithm Design

2.1 Requirements

This section is meant to provide a generalized overview of the requirements of the ARIEL Mission Requirement Document that are applicable in this thesis. These requirements are used as a basis of the development of the algorithms used by the IASW (ARIEL Science Study Team, 2020).

R-GENPERF-1/P	Short Text	Software Requirement
	Timing "Acquisition"	The "Acquisition" procedure shall operate according to Figure 2.1.

R-GENPERF-2/P	Short Text	Software Requirement
	Timing "Tracking"	The "Tracking" procedure shall operate according to Figure 2.2.

R-GENPERF-3/P	Short Text	Software Requirement
	Coordinate Measurement	The IASW shall be able to determine the coordinates of the brightest target inside of the FOV, with respect to the FGS boresight reference frame.

R-GENPERF-4/P	Short Text	Software Requirement
	Coordinate Measurement Units	The target coordinates shall be provided in terms of equivalent angular rotations around X_{BRF} and Y_{BRF} axes, considering the right-hand screw convention.

R-GENPERF-5/P	Short Text	Software Requirement
	Faint Targets	The lowest possible target signal for the guiding task shall be 21571 e/s for FGS1 and 154984 e/s for FGS2

R-GENPERF-6/P	Short Text	Software Requirement
	Bright Targets	The highest target signal for the guiding task shall be 3087075 e/s for FGS1 and 3946414 e/s for FGS2

R-GENPERF-7/P	Short Text	Software Requirement
	Wavelength Band	It shall be possible to obtain images in the wavelength band between 0.5 and 1.95 microns

R-GENPERF-8/P	Short Text	Software Requirement
	Neighbouring Targets	The IASW shall be able to perform position measurements of targets with fainter neighbouring sources inside of the FOV.

R-GENPERF-9/P	Short Text	Software Requirement
	Acquisition FoV	The "Acquisition" procedure shall support a FoV size greater than 24 arcsec $\times$ 24 arcsec.

R-GENPERF-10/P	Short Text	Software Requirement
	Acquisition Sampling Frequency	The "Acquisition" procedure shall support sampling frequency $\leq$ 1 Hz that shall not be user configurable.

R-GENPERF-11/P	Short Text	Software Requirement
	Acquisition Accuracy	The "Acquisition" procedure shall implement an algorithm that provides an accuracy of 200 mas for both faint and bright targets

R-GENPERF-12/P	Short Text	Software Requirement
	Tracking FoV	The "Tracking" procedure shall support a FoV size greater than 8 arcsec $\times$ 8 arcsec.

R-GENPERF-13/P	Short Text	Software Requirement
	Tracking Sampling Frequency	The "Tracking" procedure shall support sampling frequency of 8 and 10 Hz that shall not be user configurable.

R-GENPERF-14/P	Short Text	Software Requirement
	Tracking Algorithm	The "Tracking" procedure shall use a centroiding algorithm for the purpose of position measurements. If there are multiple algorithms provided, the algorithm used shall be configurable by the user.

R-GENPERF-15/P	Short Text	Software Requirement
	Tracking Accuracy, General	The "Tracking" procedure shall implement an algorithm that provides an accuracy of 20 mas for bright and 150 mas for faint targets

R-GENPERF-16/P	Short Text	Software Requirement
	General statistical Bound	In the modified temporal statistical interpretation, the requirements have to be considered for any possible spacecraft configuration, except for those configurations in which one or more ensemble parameters lie outside the 99.7% bound.

R-GENPERF-17/P	Short Text	Software Requirement
	Time Stamp Acurracy	The measurement time tag shall be provided with an accurracy better than 1 ms

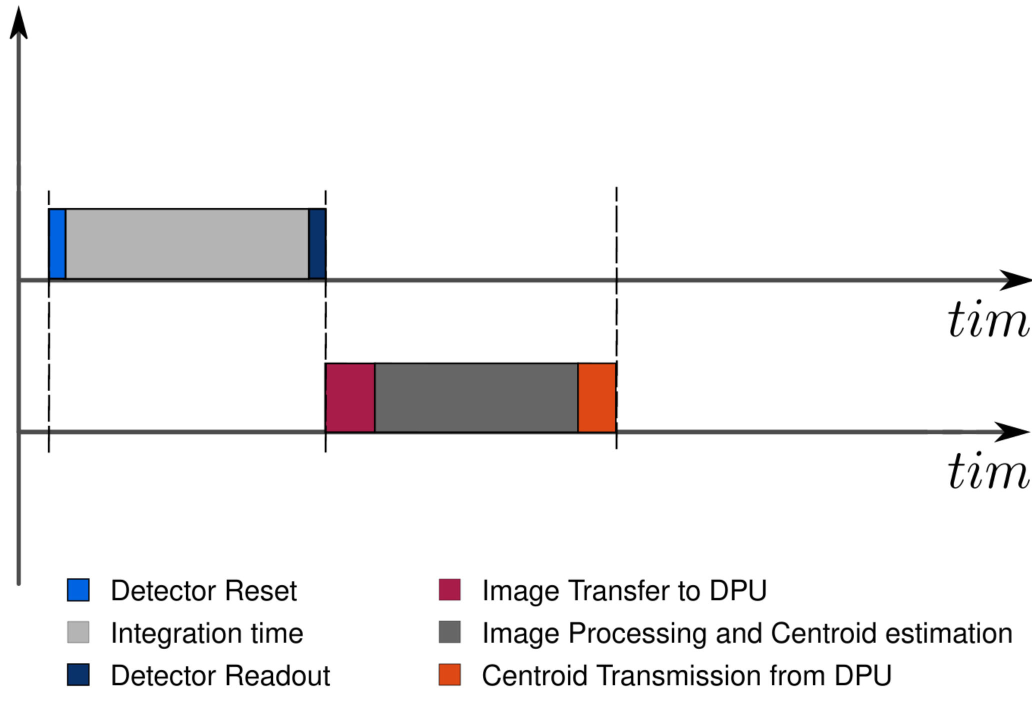


Figure 2.1: Timing diagram for the operation mode Acquisition

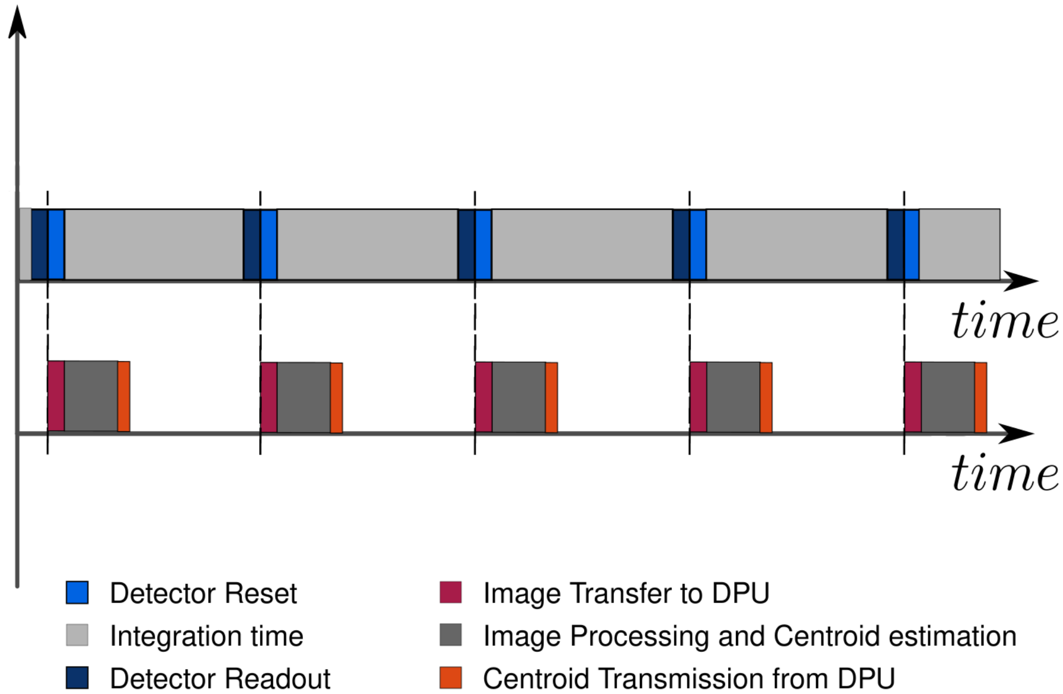


Figure 2.2: Timing diagram for the operation mode Tracking

2.2 Algorithm Design: Guiding

The main task of the centroiding algorithm is the determination of the center of a target star inside the image provided by the detector of the FGS. The resulting positional information is then relayed to the AOCS, where the data are used to correct the alignment of the telescope during use.

As a basis for the development of the algorithm, a basic IWCoG method has been chosen. This is due to the high precision and stability this method provides. As can be seen in Figure 2.5, the basic IWCoG method using ten iterations provides sufficient precision for the ARIEL FGS based on requirement R-GENPERF-15/P.

In the case of the algorithm used for ARIEL, the weighting functions tested consist of an analytical two-dimensional Gaussian and an approximation using a linear approximated Gaussian (Figure 2.3) that is rotated around its center (Figure 2.4). The linear approximation is meant to provide a reduction in computational cost. The achieved speed-up can be seen in Section 3.4.1. The exact formula for the 2D Gaussian used for IWCoG algorithms can be written as depicted in Equation 2.1.

$$W^n(x, y) = a\ exp\left[-\frac{(x - x_b^{n-1})^2}{2\sigma_x^2} + \frac{(y - y_b^{n-1})^2}{2\sigma_y^2}\right] \tag{2.1}$$

The values of x_b^{n-1} and y_b^{n-1} hereby denote the centroid of the previous iteration of the algorithm.

In both cases, the functions are rotational symmetric and have the same Full Width Half

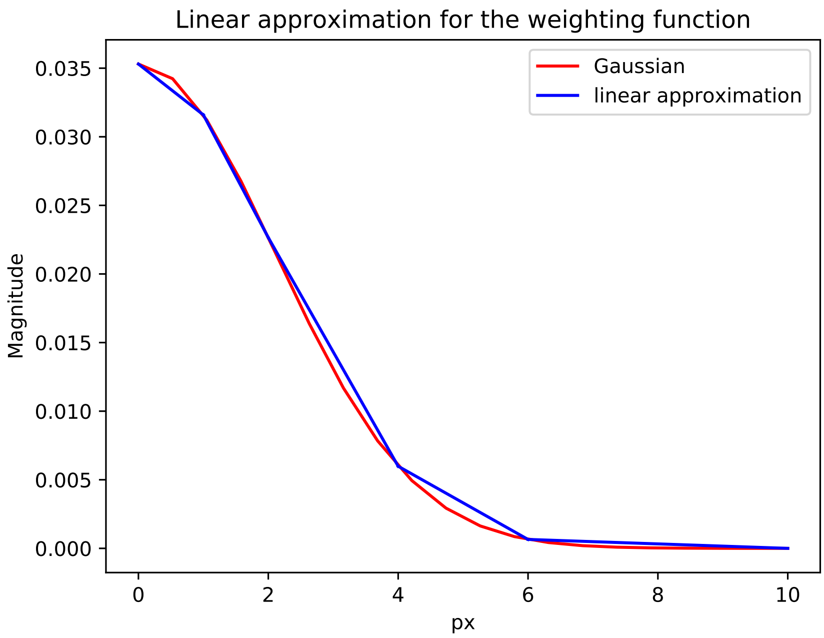


Figure 2.3: Radial profile of both weighting functions for FGS1

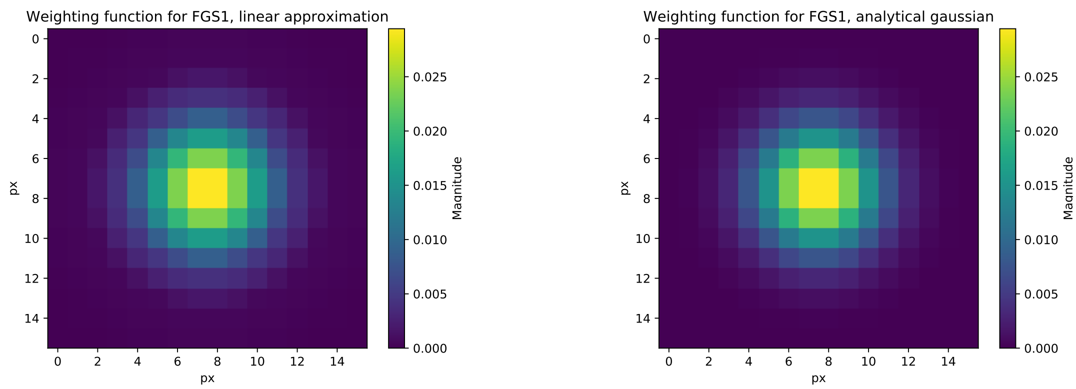


Figure 2.4: 2D weighting functions for FGS1. The left plot shows the rotation of a linear approximated Gaussian while the right plot presents the analytical 2D Gaussian.

Maximum (FWHM) of 5 px for FGS1 and 6 px for FGS2. However, it is possible to provide elliptically shaped weighting functions with both methods, if the shape of the PSF calls for it.

In order to utilize the IWCoG algorithm effectively, it is necessary to provide a sufficiently precise positional estimation, in which the weighting function is placed at the beginning of the iterative steps. This estimate is usually taken from the centroid measurement of the previous Acquisition procedure. The quality of this guess determines the needed amount of iterations to reach a given precision, which in turn directly influences the processing time of the algorithm. This means that the number of iterations can be drastically reduced by using an estimate that is already close to the position of the star. In the case of the Tracking algorithm for the ARIEL FGS a simple and quick centroiding algorithm is used instead of the previous measurement. This eliminates convergence issues due to errors in the previous measurement. It is to note that the algorithm used for this purpose shouldn't need an initial centroid estimation, which limits the selection to more basic CoG methods. The algorithm that was chosen for this purpose is an IWC algorithm as it already was applied in the CHEOPS space mission with great success (Loeschl et al., 2016).

As stated in Section 1.2.2.1, this algorithm is rather sensitive to glitches due to the squaring of the entire frame. This can cause large centroiding errors, if not addressed properly. In this case, a median filter is implemented before the IWC step in order to eliminate the influence of glitches.

To have a synthetic comparison of the performance of these algorithms, simulations that apply the same parameters as covered in Section 3.1.1 and 3.1.2 with the PSF of FGS1 were performed. In the case of the IWCoG, 10 iterations with a Gaussian weighting function with a FWHM of 5 pixels in both the x- and y-axis were used. The weighting function was positioned randomly inside of a circle with a radius of 1.5 pixels around the center of the PSF. The results are depicted in Figure 2.5. The implementation of an initial, faster algorithm results in a performance that is comparable to a standalone IWCoG with twice as much iterations, which in return means that the processing time is almost cut in half. This is further discussed in Chapter 3.

It is to note that the computational cost of the CoG algorithm is dependent on the size of the ROI. Given that the smallest ROI size of the mode Tracking is defined as a 64 px square, the multiple iterations of the IWCoG are inefficient if they are performed on the entire window. This is solved by applying a ROI refinement after the initial IWC step.

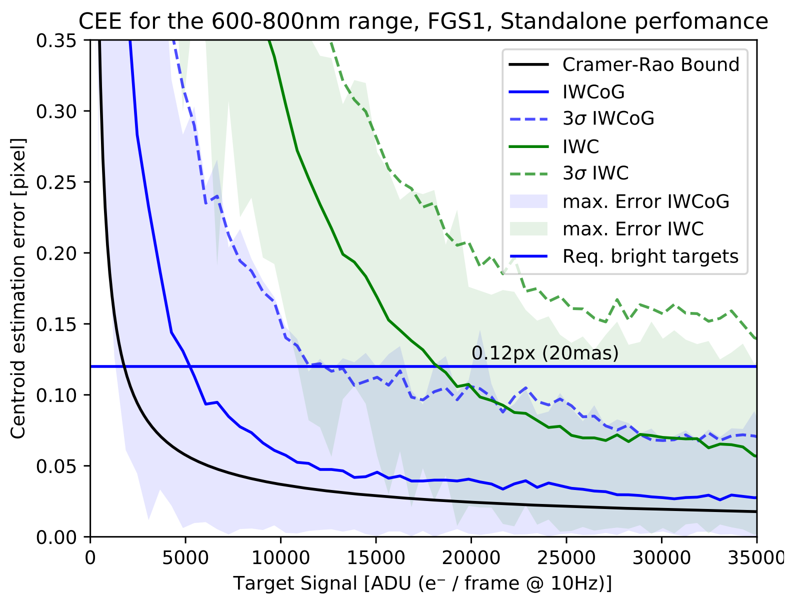


Figure 2.5: Stand-alone performance of the IWC (green) and IWCoG with 10 iterations (blue) algorithms. The solid line denotes the mean CEE, the dashed lines show a 3σ confidence interval while the shaded regions mark the envelope.

Here, a smaller window with a size of 21 × 21 px is cut around the initial estimated center. Since this can lead to a cut-off of parts of the PSF due to a large error, this window moves with the estimations of the IWCoG. In this case, the weighting function is still positioned separately instead of being kept at the center, as the missing sub-pixel shift would introduce unwanted errors.

After the calculation of the centroid, the algorithm performs a Centroid Validation, which is discussed in Section 2.5. An overview of the entire Tracking procedure is given in Figure 2.6.

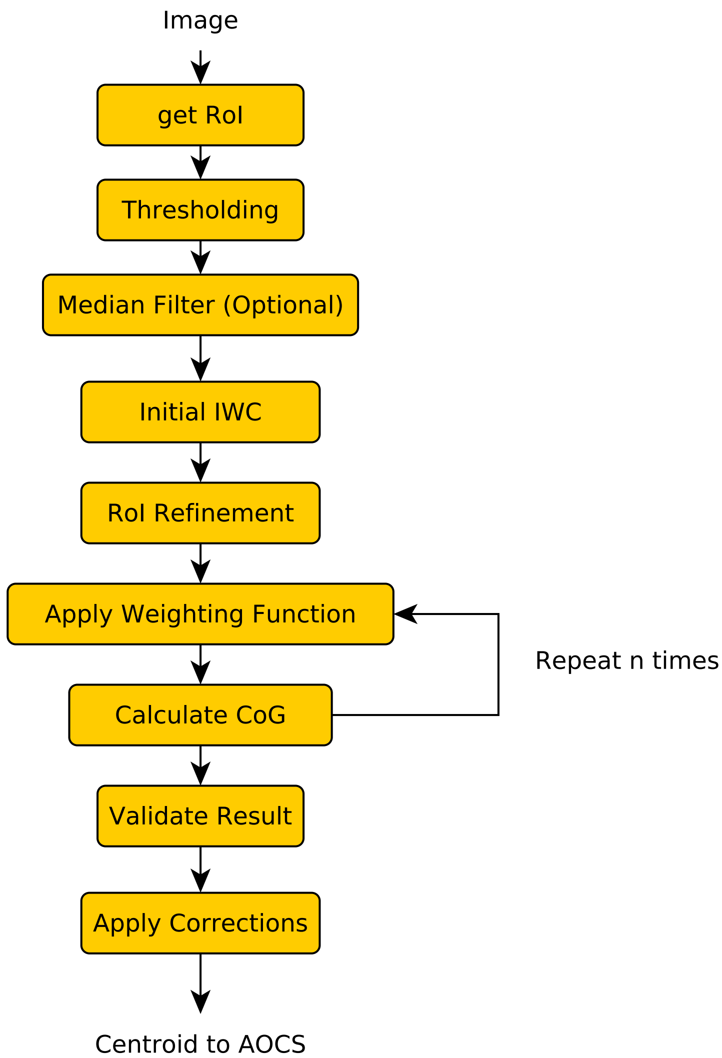


Figure 2.6: Flowchart of the algorithm used for the Tracking mode of the ARIEL FGS.

2.2.1 Influence of Thresholding

As is stated in Section 1.2.2, algorithms of the CoG family can show unfavourable behaviour when they operate on images with larger background signals and higher noise. The proposed algorithm mitigates this effect to a degree by the use of weighting functions, but subpar thresholding can still have an impact on the measured centroid. Therefore, this section covers a quick analysis of the effects of increasing bias-signals on the performance of the base IWC.

It order to study this effect, a short simulation was created that observes the one dimensional case. The set of images was simulated by creating a one dimensional array of a given length and setting the central pixel to the defined signal value. This initial mask is then convolved with a kernel that depicts a Gaussian in order to mitigate the influence of an asymmetrical PSF. As a last step, a constant offset and a random noise component is added to every pixel in order to resemble the presence of a bias.

In its main loop, the simulator creates images with decreasing signal to bias ratio. A basic IWC is then performed on the image with different thresholding methods applied. The methods consist of no threshold, the minimum of the image and a fixed value that is above the combined bias and noise. In the case of the method involving the local minimum, two different methods are tested as the proper minimum can be an outlier which would lead to faulty thresholds. In order to correct this behaviour, the second method uses the third-lowest value in the image as the thresholding value.

The results for this analysis can be seen in Figure 2.8. It is apparent that no threshold

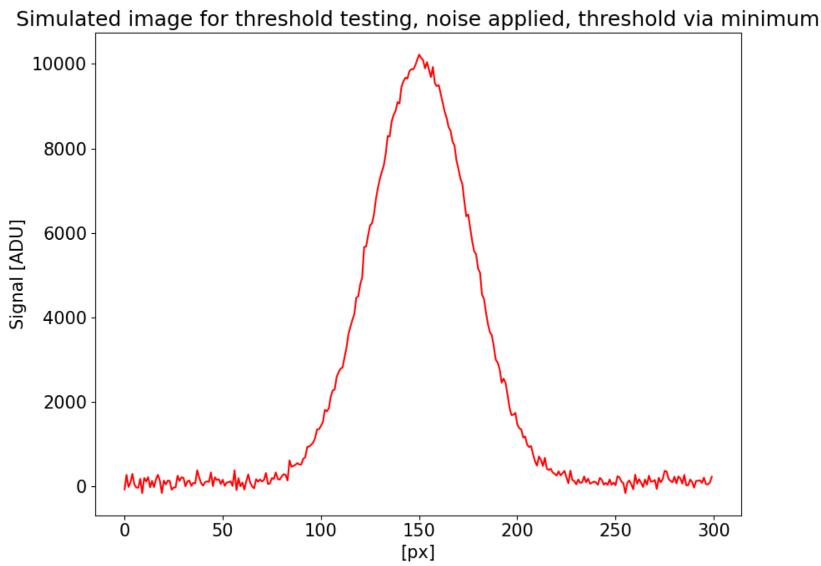


Figure 2.7: Simulated image for the thresholding tests with a bias of 100 ADU and a stellar signal of 100000 ADU.

introduces an additional knowledge error in the sub-pixel scale with increasing bias. All other tested methods show similar performance in mitigating this effect with minor shifts in the mean error. In order to provide a better overview of the performance, a zoomed-in region is shown in Figure 2.9. It can be seen that the method using a fixed value shows the best performance out of the three. This is due to the fact that most of the noise from the background and readout is cut from the image. It is to note that this is only the case if the thresholding value is properly chosen which makes this method ill-suited for the autonomous operation of a spacecraft.

Based on this, one can determine that either form of thresholding has the same effect. However, the fixed value method is unsuitable as it requires precise knowledge of the bias, noise and background signal. Therefore, the methods using the local minimum of the image are preferred. Here, thresholding using the third minimum shows a slight improvement over the basic minimum due to the minor reduction in noise.

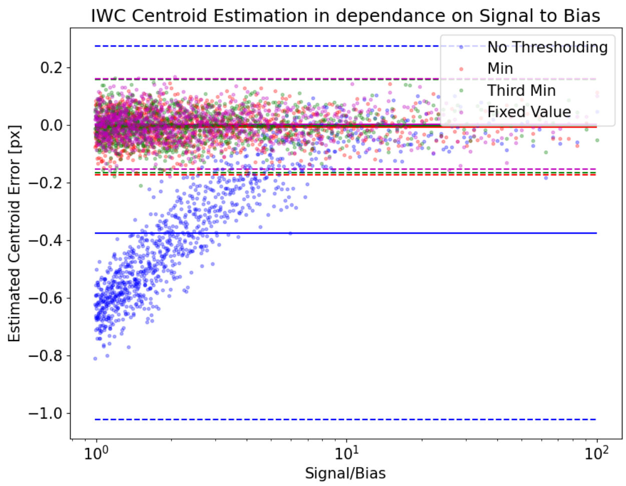


Figure 2.8: Results for the analysis of the different thresholding methods.

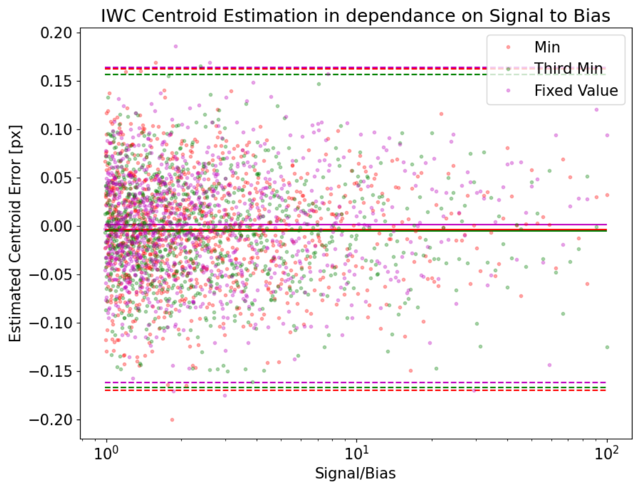


Figure 2.9: Plot showing a zoom-in on the performance of the thresholding methods.

2.3 Algorithm Design: Target Acquisition

As can be seen in the requirements, the mode Acquisition is performed on a much larger ROI when compared to the mode Tracking. Furthermore, this mode must be able to distinguish between multiple stars inside of the imagette and select the correct target from the present sources. This restraint on image size and the more generous timing requirement makes an approach of a combined source extraction and a centroiding algorithm a promising option.
The source extraction consists of two parts. The first step is to find possible targets. This is done by applying 3×3 binning and searching for the brightest pixels inside of the initial FOV. The binning uses the total signal inside of the bin instead of the mean, which helps with target validation in the later steps. Furthermore, the binning helps with reducing the computational cost of the source extraction on the large FOV.
In order to find the brightest pixels inside of the ROI, a non-destructive sorting algorithm is used that writes the n brightest pixels into an additional array, where n is specifiable by the user. The current model used for the purpose of testing uses a bubble sort (Yu et al., 2012), that is modified to produce the desired array of arbitrary length. It is to note, that this approach will be changed for a more performant algorithm in the final version of the algorithm, given the bubble sorts' complexity of $\mathcal{O}(n^2)$.
This array is part of a structure that also contains information about the pixels' position, the resulting signal of the later brightness validation and a flag that specifies, if the pixel is the target. In the initial state, this flag is set as true for all found pixels.
In the next step, the extraction checks, if the found pixels are neighbours. If that is the case, every found neighbour except for the brightest one is marked as a false target.
After this step, the source extraction checks the extent of the source. This is done by looking at the eight neighbouring pixels of the found target. Then it is counted how many pixels are above a given threshold. This is a necessity, since the size of the FOV does not permit the usage of a median filter. This means that a method to detect cosmic ray hits and glitches on the detector is necessary. Since cosmics and glitches are often confined to single pixels or trails, it will result in a lower signal of the neighbouring pixels than what is expected for an extended source.
As a last part of the source extraction process, the brightness of the target is measured and compared to the target signal defined by the user. The remaining target that is closest to the defined brightness is then used for the determination of the position on subpixel precision.
For the process of obtaining a positional measurement of the target position, a small ROI is cut from the original, unbinned image around the previously determined target. On

this ROI, the same algorithm as is described in Section 2.2 is performed. In this case three iterations are used for the IWCoG, given the lower required precision.

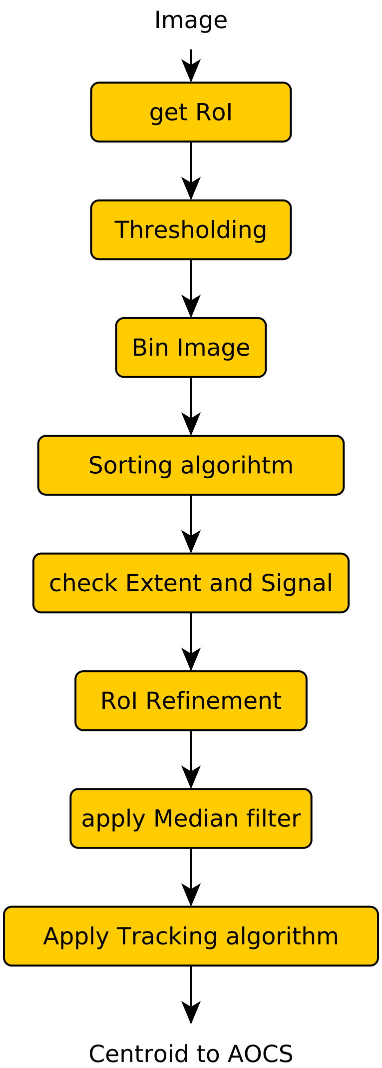


Figure 2.10: Flowchart of the algorithm used for the Acquisition mode of the ARIEL FGS.

2.4 Correction Functions

2.4.1 Offset-Correction

The PSF of the ARIEL FGS is quite asymmetrical in it's nature (Figure 3.2). This and the Gaussian weighting function used by the algorithm can lead to shifts of the center of gravity in regard to the stars' actual position. This has no implication on the overall performance of the algorithm, if this is a static effect. However, it could impact the precision if the offset is dependent on the signal of the target star.
In order to study this effect, a regular dataset that covers a range of up to 250000 ADU is taken and the algorithm is performed on the generated images. The error in the x- and y-axis is then plotted separately to analyse the dependency on signal.

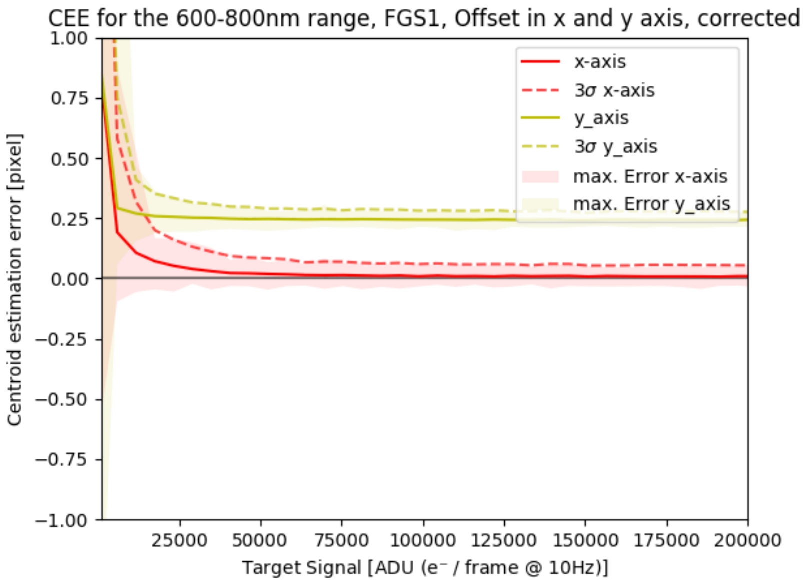


Figure 2.11: Result of the simulation run used for the offset analysis for FGS1. The solid line depicts the mean error while the shaded region describes the maximum error.

One can see in Figure 2.11 that the offset quickly converges to a constant value but shows a drastic increase in the lower signal range. This behaviour can impact the precision of the algorithm quite heavily which can be fatal.
Therefore, a correction function with a signal dependency is required. For this purpose an exponential function is fitted to the original measurements of the dataset:

$$CEE_{corr} = e^{-a*x+b} + c. \tag{2.2}$$

The resulting coefficients are given in Table 2.1.

Table 2.1: Fit parameters of the exponential offset correction.

Parameter	FGS1		FGS2	
axis	x	y	x	y
a	2.603×10^{-4}	4.942×10^{-4}	2.509×10^{-4}	3.904×10^{-4}
σ_a	6.611×10^{-6}	3.400×10^{-5}	5.727×10^{-5}	2.041×10^{-5}
b	-7.014×10^{-2}	-3.018×10^{-1}	1.068×10^{-1}	-3.963×10^{-1}
σ_b	9.122×10^{-3}	1.149×10^{-2}	8.406×10^{-3}	1.260×10^{-2}
c	1.507×10^{-2}	2.454×10^{-1}	-1.238×10^{-1}	2.714×10^{-1}
σ_c	1.353×10^{-3}	1.300×10^{-3}	1.491×10^{-3}	1.320×10^{-3}

2.4.2 S-shaped noise correction

Figure 3.5 shows periodic fluctuations of the CEE for higher signals, which can be attributed to the aliasing effect and the subpixel sensitivity (Figure 3.1). Since these effects falsify the performance of the tested algorithm, an in-depth analysis was performed in order to obtain a correction function.

For this purpose, a simulation run was performed for both channels, in which the PSF was moved step-wise across an area of four pixels in a serpentine motion. The step-size for this process was chosen to be 0.04 px in order to give us 2500 points of measurement. The centroiding algorithm is then performed on the images, which gives a map of the CEE in dependency of intra-pixel position (Figure 2.12).

The error was plotted against the measured centroid position for each axis. As can be seen in Figure 2.13, the function of the error in each axis proves to be a combination of two sinusoidal functions (Equation 2.3), which provides the basis for a least-squares fit. The resulting fit parameter are shown in Table 2.2. The correction function is then applied to the results shown in 3.2.1 which gives us Figure 2.14.

$$CEE = A_x * sin(k_x * x + \phi_x) + A_y * sin(k_y * y + \phi_y) \tag{2.3}$$

Table 2.2: Fit parameters of the S-shaped noise correction for both spectral bands.

Parameter	FGS1		FGS2	
axis	x	y	x	y
A_x	2.130×10^{-2}	1.629×10^{-2}	1.284×10^{-2}	1.649×10^{-2}
σ_{Ax}	2.619×10^{-4}	2.179×10^{-4}	2.335×10^{-4}	1.346×10^{-4}
k_x	6.349	6.323	6.600	6.394
σ_{kx}	2.337×10^{-2}	2.410×10^{-2}	3.292×10^{-2}	1.529×10^{-2}
ϕ_x	0.1395	-1.797	-0.9154	-0.8493
$\sigma_{\phi x}$	3.063×10^{-2}	3.023×10^{-2}	3.539×10^{-2}	1.607×10^{-2}
A_y	5.358×10^{-2}	9.164×10^{-3}	4.219×10^{-2}	5.737×10^{-3}
σ_{Ay}	3.728×10^{-4}	1.968×10^{-4}	4.398×10^{-4}	6.887×10^{-4}
k_y	6.334	12.656	6.499	6.482
σ_{ky}	1.275×10^{-2}	3.816×10^{-2}	1.765×10^{-2}	2.090×10^{-1}
ϕ_y	0.5701	-1.222	0.7222	-1.255
$\sigma_{\phi y}$	1.620×10^{-2}	5.106×10^{-2}	2.313×10^{-2}	2.964×10^{-1}

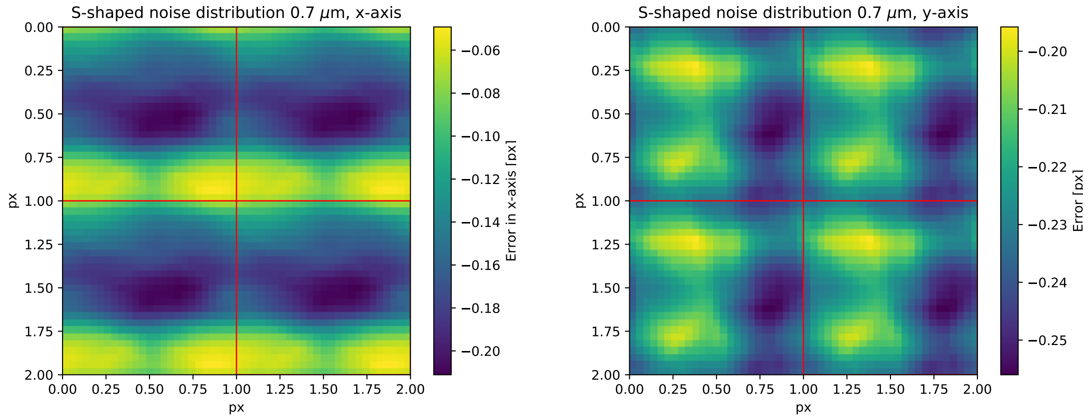


Figure 2.12: CEE in dependency of sub-pixel position for the x- and y-axis. The red square denotes the area of a single pixel. The distribution of the error clearly shows a periodic behaviour, which was expected by the nature of the aliasing.

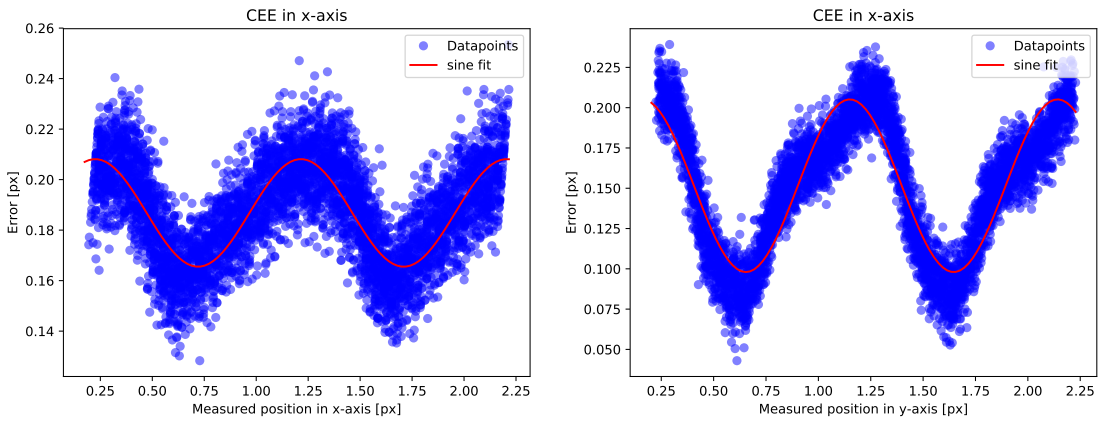


Figure 2.13: Fit of the error in the x-axis along both the x- and y-axis for FGS1.

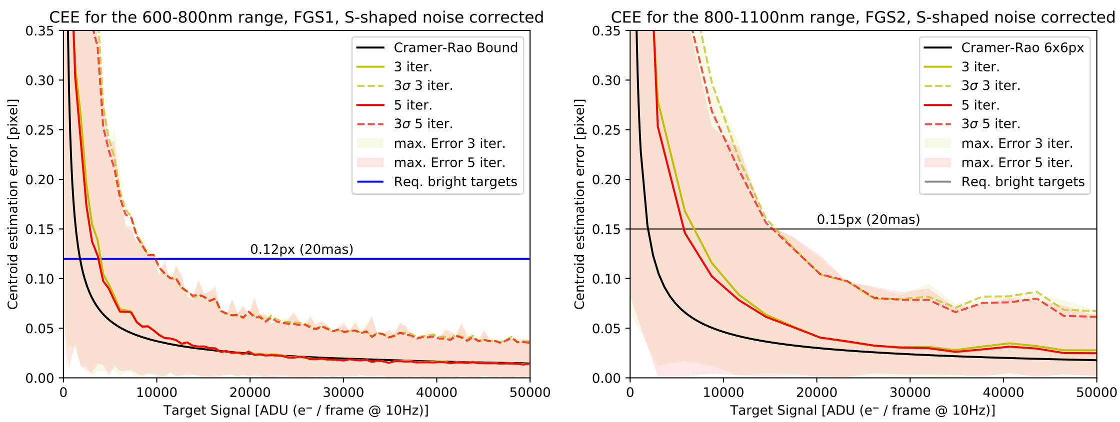


Figure 2.14: S-shaped noise corrected CEE for the centroiding algorithm with 3 (yellow) and 5 (red) iterations for FGS1 (left) and FGS2 (right).

2.5 Centroid Validation

This section gives a quick overview of the methods used to validate the measured stellar positions from the centroiding algorithm. For this purpose, a validation procedure is implemented that builds on the centroid validation used in CHEOPS (Loeschl et al., 2016). In the context of this process, the ROI is checked in regard to the following six rules:

1. The image must not be constant.
2. The image must contain a star.
3. The star must be a distinct feature.
4. There must be a signal.
5. The sigma of the signal must be large.
6. The signal inside of a defined aperture around the measured center must surpass a given percentile of the image.

The first five rules are basic validity checks that are performed on a binned image and will void the centroid if they aren't met. In the context of ARIEL, the image is binned to $6{\times}6$ px, but the factor can be adjusted in the future. The last rule is additionally used as the quality index which will flag the centroid either as valid, compromised or void. This test is performed on the original ROI after a median filter was applied.

2.5.1 Quality Index

As stated, the usage of the percentage of the signal inside of a defined aperture around the center provided by the centroiding algorithm is proposed as the quality index. This test provides a metric of the degeneracy of the PSF and furthermore performs the required magnitude measurement of the target star. The size of the aperture can be arbitrarily set to match the ideal size of the PSF. In the context of this analysis, the aperture radius is set at 5 px for FGS1 and 6 px for FGS2. Furthermore, the thresholds at which the centroid will be flagged as valid, compromised or void is dependent on the magnitude of the desired target star.

3. Performance Testing

3.1 Simulated Data

3.1.1 FGS Data Simulator

The images used for further performance analyses were generated using a designated data simulator called StarSim, which is a standalone Python program. This simulator was designed with the guiding process in mind but can be used for a multitude of applications, with the main task of StarSim being the generation of FITS-images that resemble real observational data. This means that StarSim provides both hardware features and secondary signals in addition to stars with varying source signals. The hardware features consist of detector specific features such as bias, read-out noise, flatfield on the sub-pixel scale and hotpixels, as well as optical features like the distinct shape of the PSF. On the other hand, secondary signals include the Poisson-noise and background signals. This is achieved by creating an initial star mask, that is convolved with the desired PSF. The position and brightness of the stars are read from an external xml file which further contains information about the target star that is used for the guiding process (Ferstl, 2016). In order to take the different sensitivities inside a single pixel into account, the images are upsampled with the oversampling factor that's being determined by the size of the sub-pixel flatfield. In the context of the simulations depicted in this document, an oversampling factor of five is applied with the used sub-pixel flat being shown in Figure 3.1. It is to note, that this is only a rough estimate and it will likely be changed in the future, if more detailed information about the Teledyne H2RG is provided.

After the creation of a frame containing nothing but the stars with a signal corresponding to the config file and the exposure time, the background and photon noise are added. The artificial image is then multiplied with the quantum efficiency and master-flatfield containing the intra-pixel sensitivity variations. The resulting image is then downsampled to the final image resolution. Lastly, effects like glitches, bias- and darkframes are added to the final image.

In addition to the images, StarSim also provides metainfo in form of an extra xml file. This file contains information about the used parameters and furthermore provides the position of the target star for each image, which is needed for the determination of the centroiding error in the context of the FCU Simulator.

Table 3.1: Simulation Parameters of the FGS Data Simulator.

Parameter	Value
Sample size [images]	26000
Minimum signal step [p/s]	1000
Maximum signal step [p/s]	10000
Samples per step [images]	200
Detector size [pixel]	200×200
Avg. Background [p/s/px]	1.10
Avg. Read-noise [e-/s]	20
Avg. Dark current [e-/s/px]	0.01
Quantum efficiency	0.58
Exposure Time [s]	0.1
Full-well capacity	100000
Flat mean	0.97
Flat σ	0.03
Flat gradient lower limit	0.99
Flat gradient upper limit	1.0
Flat gradient changes	0.0001
Flat gradient angle	0.0
Bias value	2000

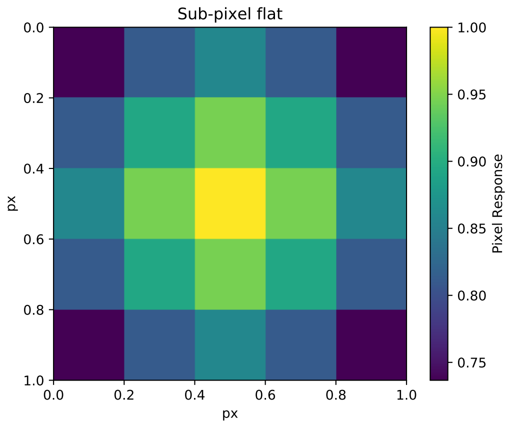


Figure 3.1: Sub-pixel flatfield used in the FGS Data Simulator.

The Data Simulator was run with the parameters given in Table 3.1 for both spectral bands with their respective PSF. These were provided by the University of Rome and were rescaled according to the F-number, pixelscale and pixel-pitch, which are given in Table 3.2 (Figure 3.2). The PSF were then oversampled by a factor of five in order to comply with the oversampling introduced with the sub-pixel flatfield (Figure 3.3).

Table 3.2: Optical parameters for both FGS channels.

Parameter	FGS1		FGS2	
Pixelscale	0.1249		0.1249	
Pixel pitch	18		18	
F-number	−29.57	−19.67	−37.59	−25.00

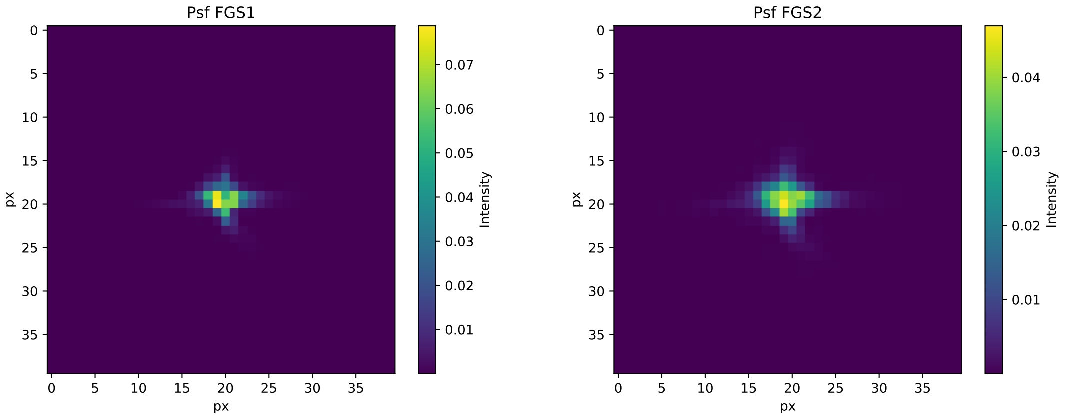


Figure 3.2: PSF for FGS1 (left) and FGS2 (right)

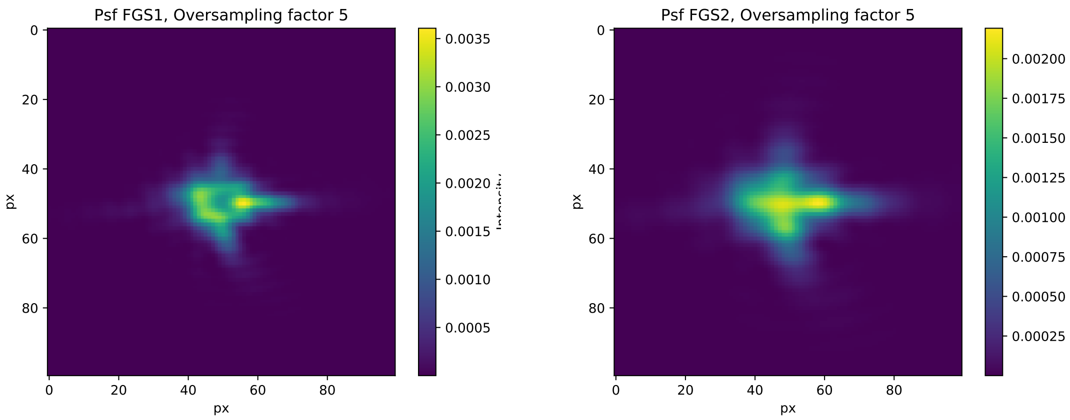


Figure 3.3: Oversampled PSF for FGS1 (left) and FGS2 (right).

3.1.2 FCU-Simulator

The FCU-Simulator is a standalone Python program used to emulate the data processing steps that are handled by the DPU during normal operation. In the scope of this simulator, these task can be summarized as windowing, applying thresholds and performing the respective centroiding algorithm. The first step that is performed is windowing, which is the definition of the subframe of the image, on which the centroiding shall be performed. The FCU-Simulator provides windowing modes where the ROI is positioned in the center of the image, in the location of the last centroid or in the exact position of the target star. The size of the ROI is variable and can either be rectangular or round in shape.

The next step is the optional application of a threshold. Thresholding is a useful tool to improve the accuracy of noise-dependent centroiding algorithms, where pixels with an intensity below a defined value are set to zero. The FCU-Simulator provides multiple thresholding modes, where the value of the threshold can be predefined or the minimum intensity inside of the ROI.

Lastly, the simulator performs the centroiding algorithms, that are specified in the config file. In case of the algorithm used for ARIEL, the configurable parameters consist of the number of iterations, the used weighting function and the dimensions of the weighting function. In addition to the centroiding algorithms, it is possible to calculate other relevant parameters such as the Cramer-Rao bound (Section 1.2.3).

The main parameter used to determine the performance of the algorithm in regard to centroiding precision within the FCU-Simulator is the CEE, which represents the distance between the estimated and true centroid. The CEE can be defined with following formula:

$$CEE = \sqrt{(x_c - x_c^*)^2 + (y_c - y_c^*)^2} \tag{3.1}$$

where (x_c, y_c) denotes the actual position of the star and (x_c^*, y_c^*) is the position provided by the algorithm.

The resulting output of the simulator contains the signal of the star, the true position of the PSF, the measured position, the CEE in both axes and additional information depending on the used algorithm.

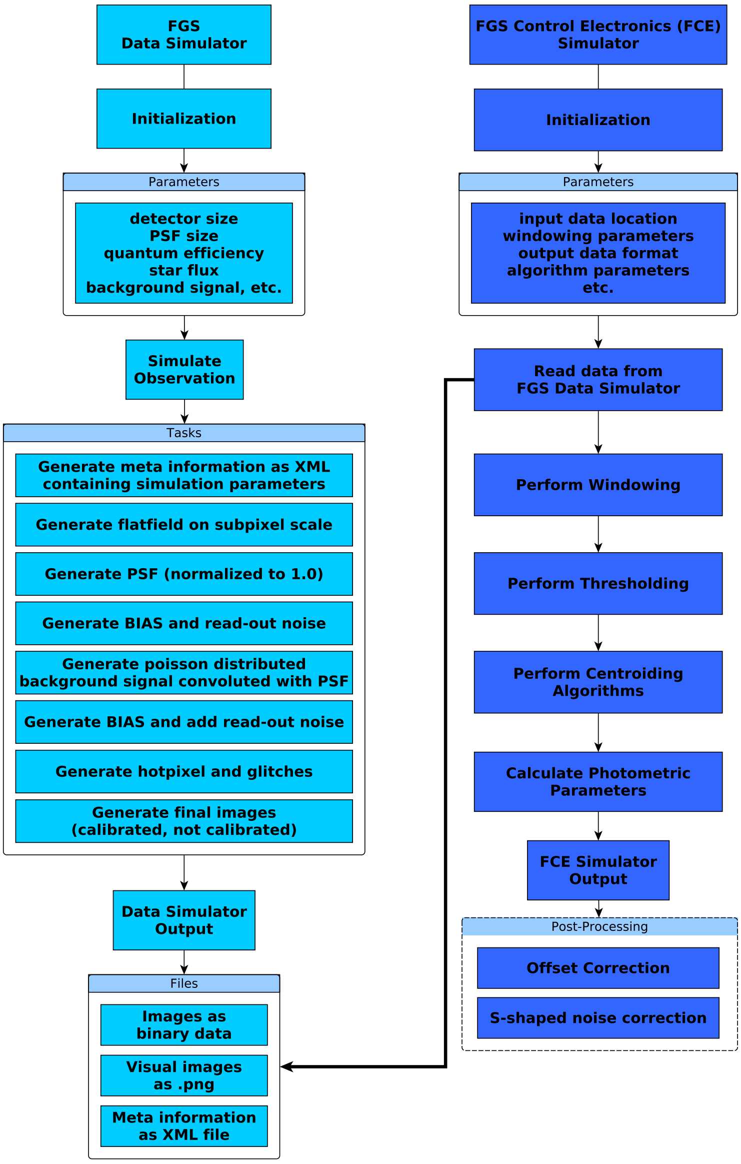


Figure 3.4: Architectural design of the simulation framework used for the ARIEL FCU centroiding simulations. The two parts consist of the Data-Simulator (light blue) that is used for the generation of images consistent with the instrumentation and the FCU-Simulator, which uses the generated images to perform the chosen algorithms.

3.2 Tracking Performance

3.2.1 Baseline Precision

In the scope of this analysis, the CEE was determined for 130 samples with signals ranging from 60 ADU to 73260 ADU with each sample containing 200 images. For the initial tests, the performance for the algorithm using both 3 and 5 iterations after the initial intensity weighted step were analysed for both spectral bands, using the linear approximation as the weighting function. The results are presented in Figure 3.5 and Figure 3.6, indicating that both implementations provide sufficient accuracy to reach the requirements for both faint and bright targets.

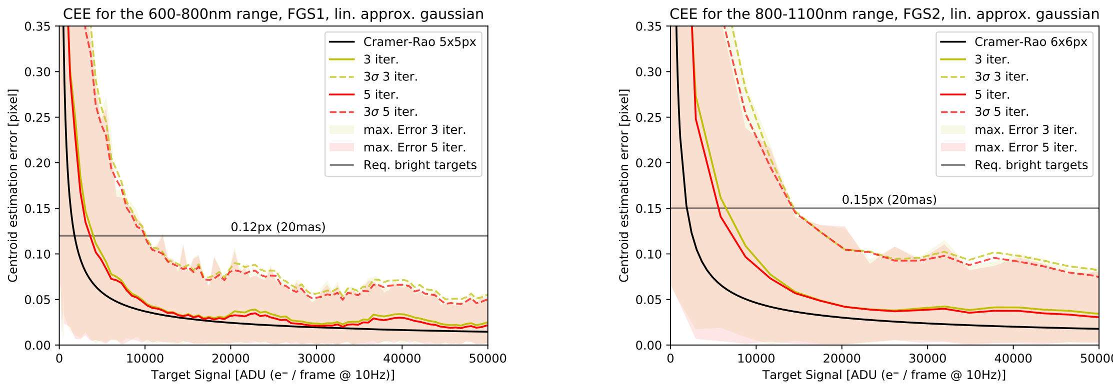


Figure 3.5: CEE for the centroiding algorithm with 3 (yellow) and 5 (red) iterations for FGS1 (left) and FGS2 (right). The solid line shows the mean, the dashed line shows a 3σ confidence interval and the shaded region shows the absolute error. The mean was subtracted in both x- and y-axis.

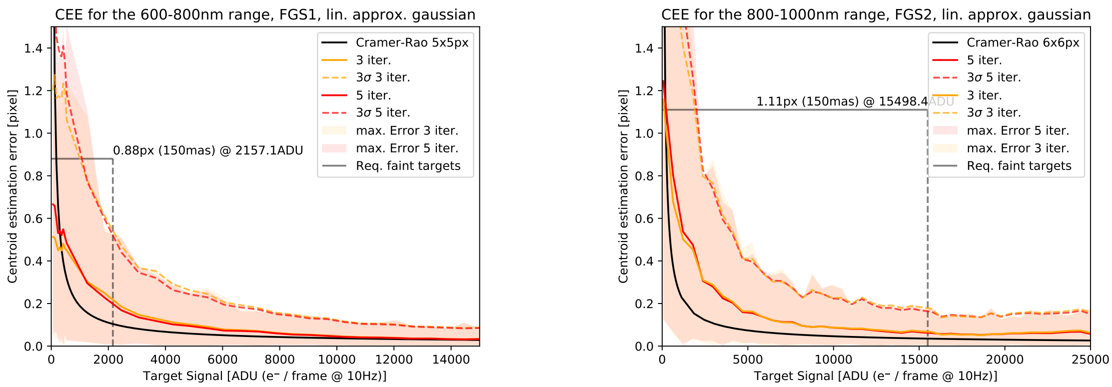


Figure 3.6: CEE for the same simulation runs with a focus on the lower signal range. The vertical dashed lines denote the faint star signal with the lower signal being an additional goal at 50% of the required faint target brightness.

3.2.2 Jitter Analysis

In this section the performance of the algorithm in the presence of spacecraft jitter is discussed. Jitter can be classified by its frequency and the amplitude of the movement. Two cases are differentiated in regard to frequency with the first one being high-frequency jitter with $f > 1/2/T_{exp}$, which causes smearing. This is a change of the spatial distribution of the signal on the detector. The second case is low frequency jitter with $f < 1/2/T_{exp}$, where the jitter causes shifts of the PSF inside of the focal plane.
In the scope of this analysis, six jitter cases given in Table 3.3 and 3.4 were used. In these cases, W_0 and W_1 denote the Power Spectral Density (PSD) of the jitter in the given frequency range.

Table 3.3: Jitter cases for bright targets.

Case	RPE[mas]	W_0[mas/$\sqrt{Hz}$]	RPE[mas]	W_1[mas/$\sqrt{Hz}$]
	90^{-1} Hz<f<10 Hz	90^{-1} Hz<f< 10 Hz	10 Hz<f<20 Hz	10 Hz<f<20 Hz
1	180	17	90	8
2	140	13	140	13

Case	RPE[mas]	W_0[mas/$\sqrt{Hz}$]	RPE[mas]	W_1[mas/$\sqrt{Hz}$]
3	90	8	180	17

Table 3.4: Jitter cases for faint targets.

Case	RPE[mas]	W_0[mas/$\sqrt{Hz}$]	RPE[mas]	W_1[mas/$\sqrt{Hz}$]
	300^{-1} Hz<f<10 Hz	300^{-1} Hz<f<10 Hz	10 Hz<f<20 Hz	10 Hz<f<20 Hz
1	390	36	90	8
2	375	35	140	13
3	357	33	180	17

The jittered PSF is created by convolving the existing PSF with the appropriate jitter-kernel. In the scope of this analysis, this kernel is estimated using two different approaches.

The first model uses a uniformly distributed jitter kernel with the radius R_k. This radius denotes the 99.7% Half-Cone radius as defined in ECS (2008).

The resulting kernel radius for each case is given in Table 3.5. It is to note, that this approach grossly overestimates the low frequency jitter, because the image is only affected by jitter-timescales below its exposure time. This would result in a much lower contribution of W_1.

The second method is based on a combination of the contributors using convolution, with the idea being, that the movement caused by high-frequency jitter sits on top of the shift resulting from the low frequency jitter. Since there aren't any directional dependencies for both error sources, both are modelled using an uniform distribution using the RPE given in Table 3.3 and 3.4. The size of the kernel is hereby given by the radius $R_{3\sigma}$ containing 3σ of the resulting distribution.

Table 3.5: Kernel radius for each case and distribution model.

Case	R_k [mas]	$R_{3\sigma}$ [mas]
Bright		
1	203	250
2	199	262
3	204	250
Faint		
1	400	456
2	405	478
3	403	501

It can be seen that the largest deviation between the two models amounts to 98 mas for case 3 of the faint targets. In order to study the effects of the different models, a simulation run using the case 2 of the bright targets was performed since the discrepancy of the signal distribution is most prevalent in this example. The resulting CEE-plots can be seen in Figure 3.7. It can be seen that there is a negligible difference between the two models. Based on these results the analysis was performed using the 99.7% Half-Cone radius model based on the premise that the normal distribution resembles a worse case in the outer regions of the kernel. The plots for each jitter case can be seen in Figure 3.9 and 3.10. In addition to the provided cases, worst case kernels were modelled in order to address the larger size difference of case 3 of the faint targets.

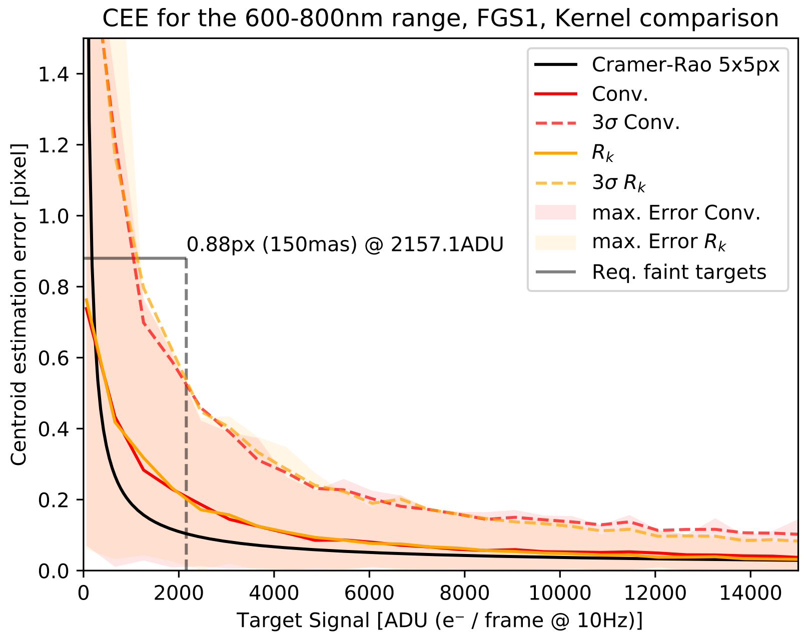


Figure 3.7: Comparison between the two jitter kernel models for case two of the bright targets.

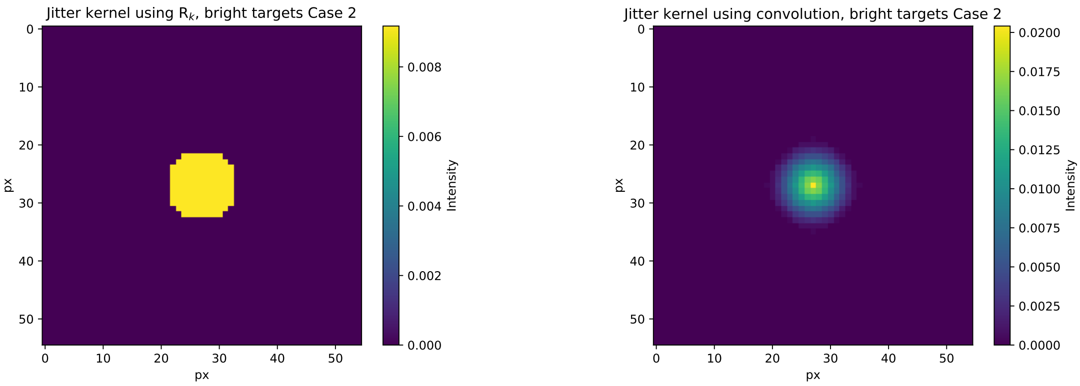


Figure 3.8: Jitter kernels for case 2 of the bright target cases. The left plot shows the 99.7% Half-Cone Radius model while the right image depicts the convoluted model.

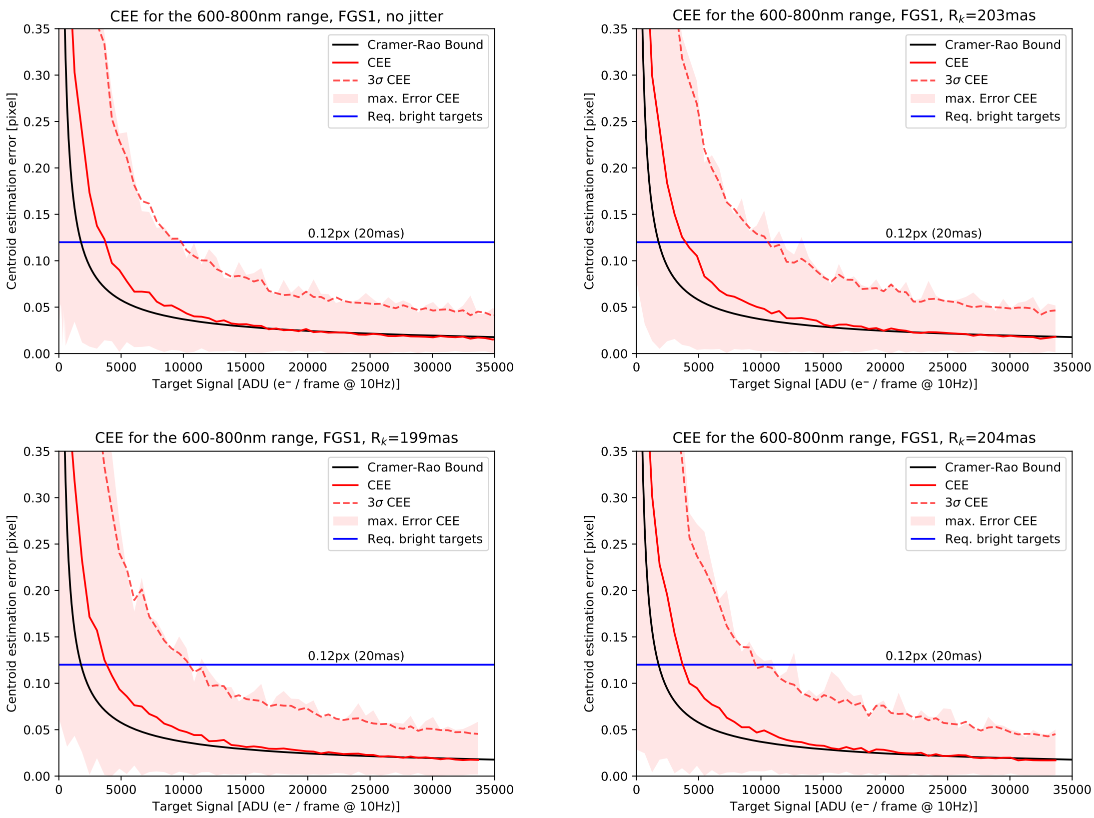


Figure 3.9: CEE plots for the three bright target jitter cases.

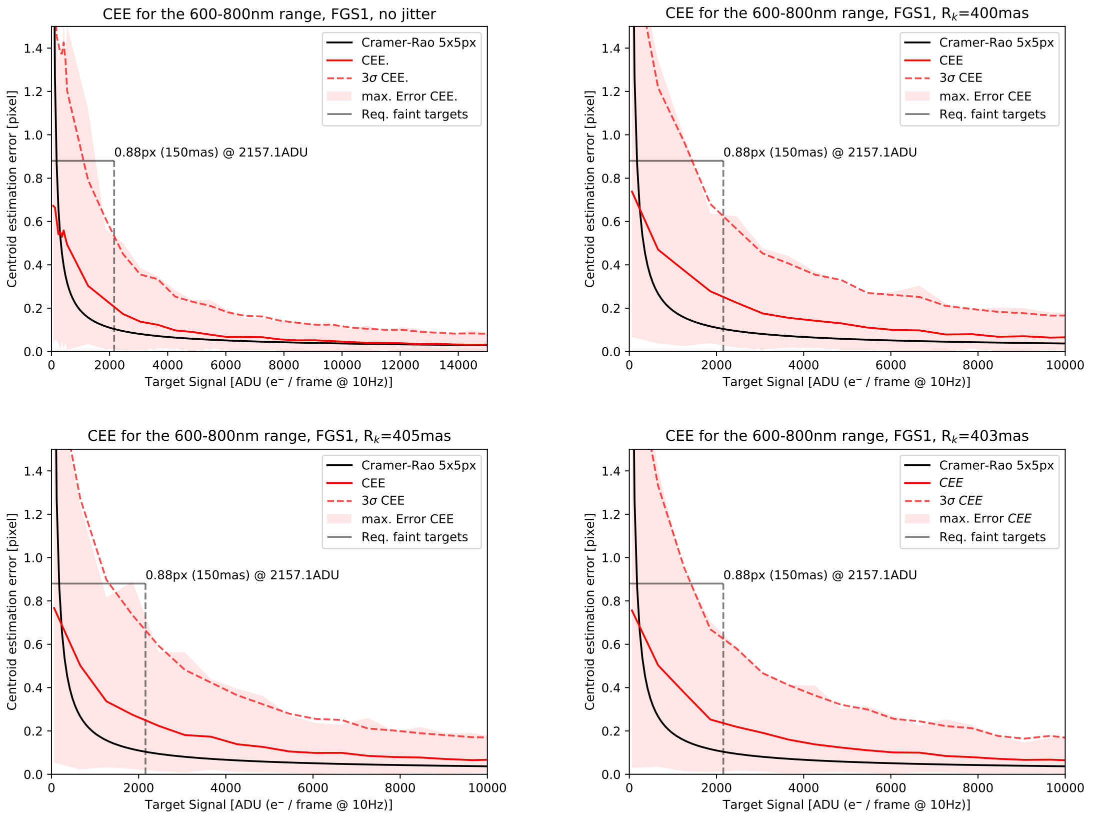


Figure 3.10: CEE plots for the three faint target jitter cases.

3.2.2.1 Modeling of the worst case

In addition to the cases presented in this section, a worst case was modelled in order to study the effects that a large increase in jitter would have on the centroiding performance. For this, case 1 for faint targets with a kernel radius of 400 mas was used and simulations with a size increase of 50% and 100% using the 99.7% Half-Cone RPE model were performed.

It can be seen in Figure 3.11, that the algorithm can still provide the required precision with a jitter kernel with a radius of 600 mas using a Gaussian weighting function with a FWHM of 5 px.

It is to note, that the plot for the 100% size increase shows a large deviation in the total error at 6000 ADU. This is due to a hot pixel inside of the PSF of the target star (Figure 3.12), which causes a large error of the initial estimation using the IWC algorithm because of its inherited sensitivity to glitches. This behaviour will be fixed in future iterations by applying a median filter to the ROI before the IWC step is performed.

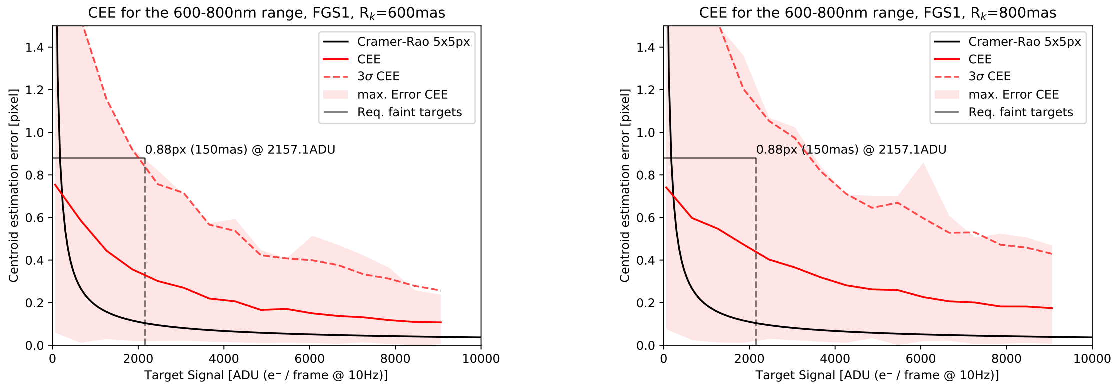


Figure 3.11: CEE plots for the worst case jitter models. The left plot shows the 50% size increase of case 1 for faint targets. The right figure shows the model using a 100% size increase.

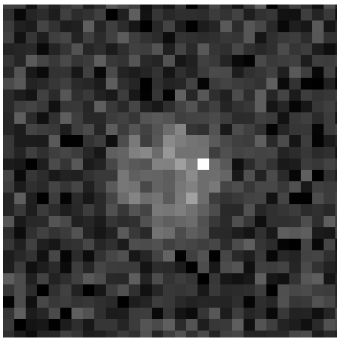


Figure 3.12: Imagette leading to the large maximum error in the worst case model analysis.

3.2.3 PSF-Size

In addition to simulations using the current PSF, runs using a PSF that was enlarged by 20% were also performed due to the results of recent developments in regard to the optics. This analysis was only performed for FGS1 using 5 iterative steps. It was opted to only simulate FGS1, since the PSF-shape is more problematic because of its ring-like nature. Figure 3.13 shows that the algorithm still achieves the required precision using an unmodified weighting function. Additionally, the performance of the algorithm was tested with a combination of a 20% PSF-size increase and both worst case jitter kernels (Figure 3.14). It can be seen, that the algorithm still fulfills the requirement at 2157.1 ADU. As a follow up, the dependency between the performance of the algorithm for the faint targets and the PSF-size was analysed. For this, a samples of 400 images was created with a star signal of 2157.1 ADU with zoom-factors ranging from 1 to 2 with a step-size of 0.25. The results can be seen in Figure 3.15 and it has proven, that the algorithm still performs properly up to a zoom-factor of 1.6 without changing the weighting function. It is to note however, that this simulation didn't account for any changes in the form of the PSF which will be a consequence of a WFE change. This alteration will be investigated in future runs.

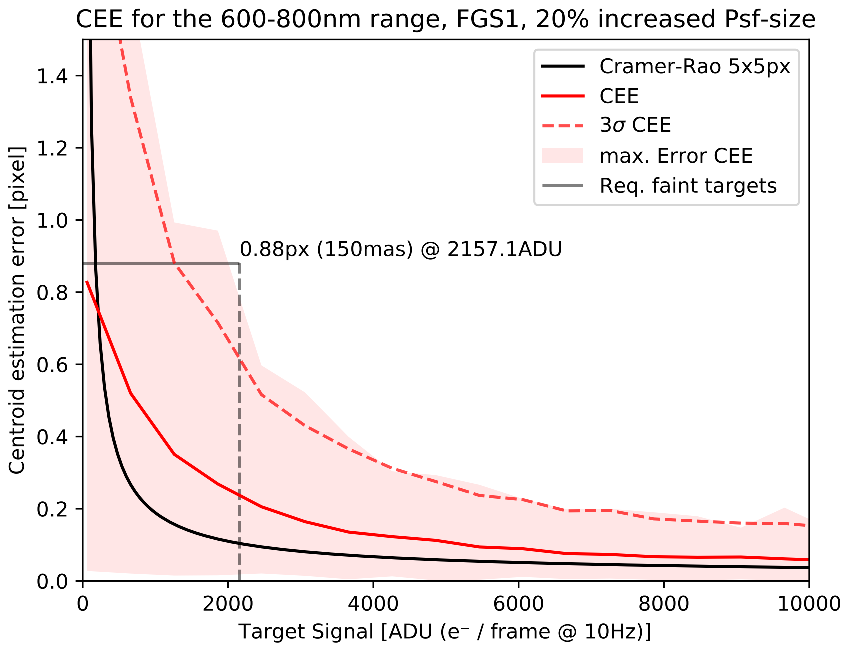


Figure 3.13: CEE plot using the PSF of FGS1 with a 20% size increase.

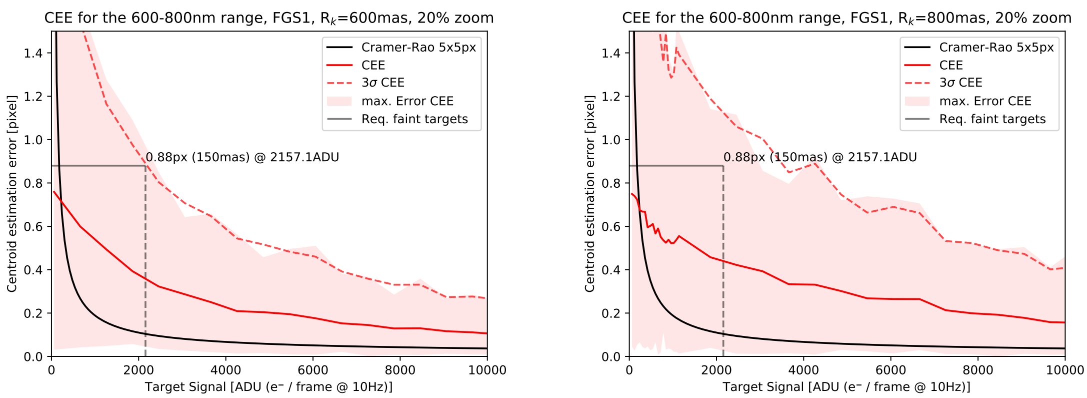


Figure 3.14: CEE plots using the 20% zoomed PSF convolved with either of the worst case jitter kernels. The left plot shows the performance using the kernel with $R_k = 600\,\text{mas}$ for faint targets. The right figure shows the model using $R_k = 800\,\text{mas}$.

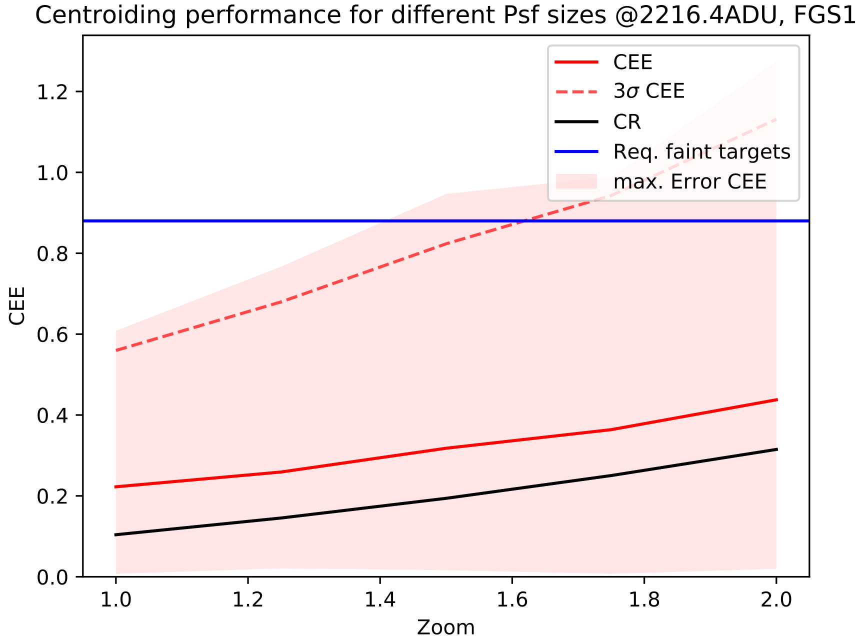
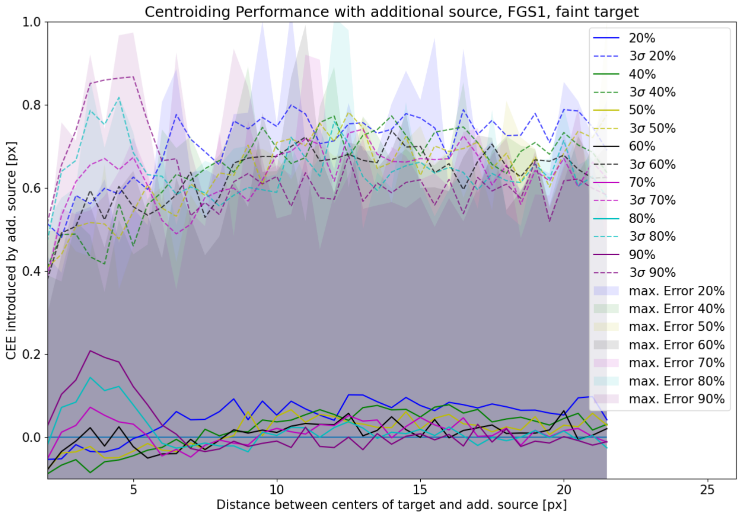


Figure 3.15: Plot showing CEE @2157.1 ADU versus zoom-factor.

3.2.4 Presence of an additional source

In this section, the influence of an additional star inside of the ROI is presented. The initial set of simulations was performed for both FGS channels using the PSF of the 270 nm WFE case. The additional source was positioned at a starting distance of 2 px in x-direction measured from the center of both stars and is moved in 0.5 px intervals away from the target where 200 images were simulated for each position. These simulation-runs were performed for both a bright and faint target with the brightness of the additional source ranging from 20% to 90% of the targets' signal. The CEE difference of performing the centroiding algorithm on these images compared to the baseline performance is given in Figure 3.16 and 3.17.



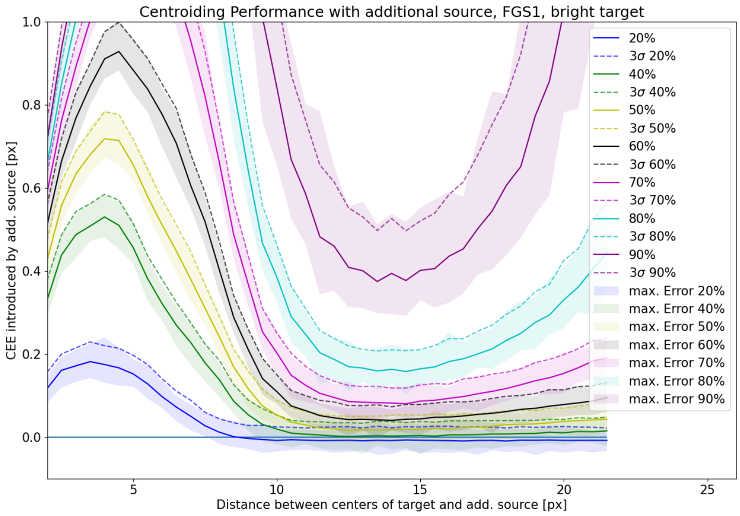
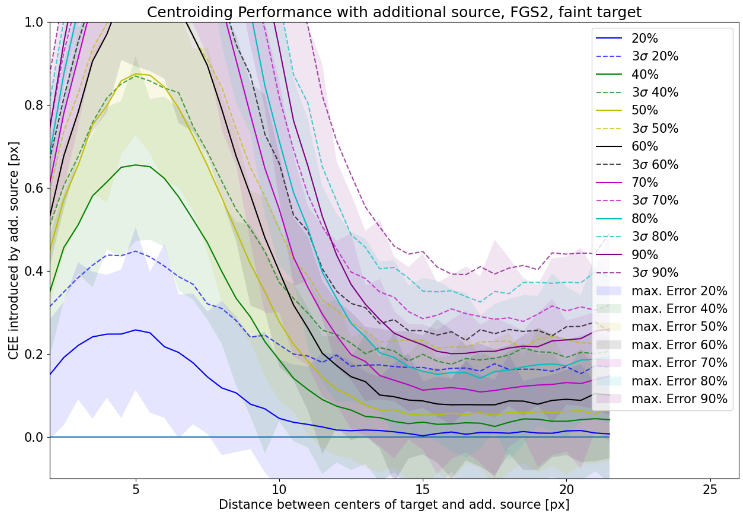


Figure 3.16: CEE plots for faint (top) and bright (bright) target stars for FGS1 with different additional source signals and increasing distance. The CEE shown is the difference between the measured error and the mean baseline performance of the algorithm without the additional source for the corresponding target signal.

Based on Figure 3.16 and 3.17, one can see that any presence of an additional source has an impact on the performance out to a maximum distance of roughly 15 px for bright targets of FGS2.

In the case of a bright target star for both channels, one can observe that additional sources with more than 50% brightness cause a second increase in error at further distances. This behaviour is a result of the first rough centroiding step using the IWC. The IWC weighs the image with itself, which also includes the additional source. When the Center of Gravity is calculated, it will be situated between the two sources in respect to their signal (Figure 3.18). This can lead to a state, where the weighting function for the following iterations is placed in a position where it does not overlap with the signal



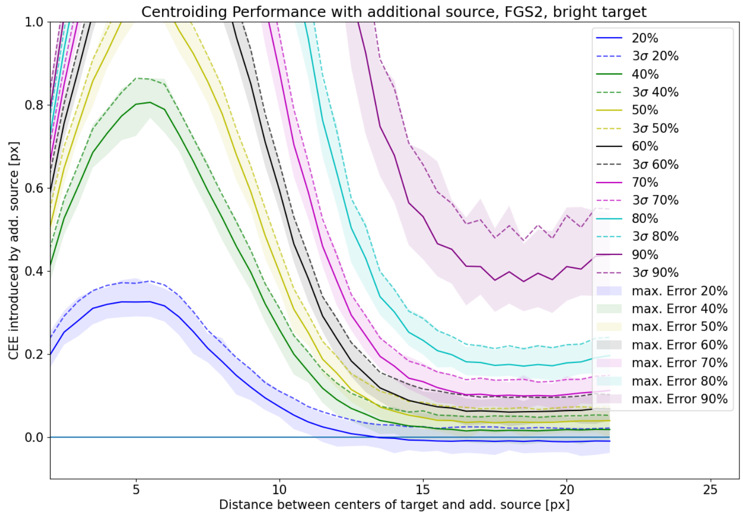


Figure 3.17: CEE plots for faint (top) and bright (bright) target stars for FGS2 with different additional source.

of either of the two stars. In this case, the IWCoG wanders around aimlessly, causing significant errors.

It is to note that this analysis also shows that the presence of an additional source doesn't cause an increase of the spread of the error at larger distances with lower neighbouring signals. With this knowledge, one can assume that an additional source that doesn't overlap with the target star only has minimal impact on the APE of the algorithm. However, this is only valid if the additional source has a signal of 50 % or less, when compared to the target.
In order to test this observation, a simulation run with a secondary star with 50 % brightness at a distance of 18 px center to center was performed with increasing signal. Based on

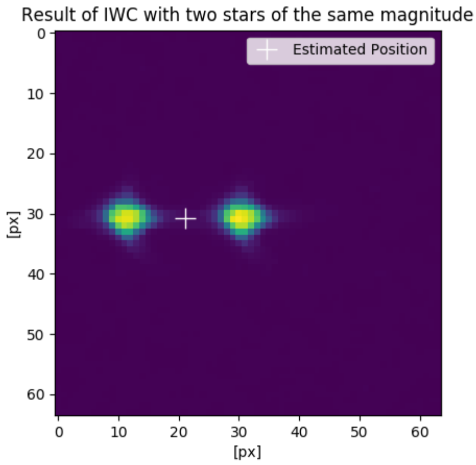
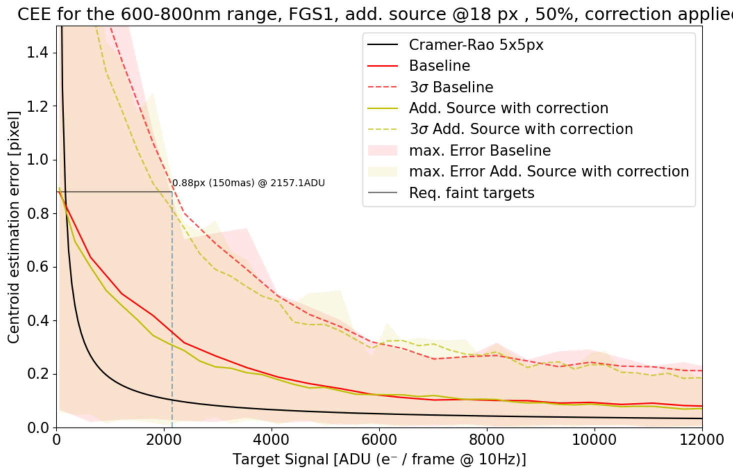


Figure 3.18: Result of the IWC on an image containing two bright stars of the same brightness with a distance of 19 px for FGS1.

the observed behaviour the result should be the same, if the offset correction is adjusted for this case. The result of this analysis is presented in Figure 3.19.

One can see in this plot that the centroiding performance of the algorithm in the additional source case is comparable to the baseline. This means that the presence of a secondary star inside of the ROI at a reasonable distance doesn't have an impact on the overall performance error of the measurement, but it does increase the knowledge error. This unchanged performance error means that there is no influence on the guiding process. However, the knowledge error does have negative effects on the centroid validation as this could result in a lost signal in the aperture photometry. Therefore, it is necessary to correct this error, which is possible if the position and brightness of the second source is known. There are multiple solutions to this issue, where one could measure the position in the Acquisition process or simulate the position on the ground and send the information with telecommands. These methods will be part of future work on the system and are not covered in more detail in this thesis.



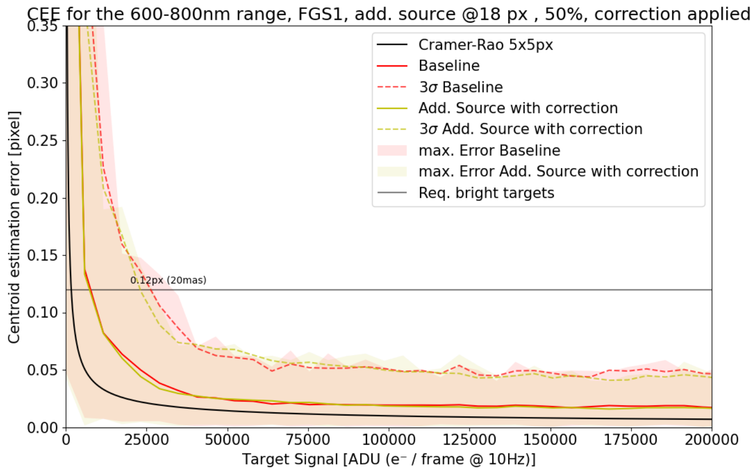


Figure 3.19: Comparison between the centroiding performance of FGS1 on a normal image (red) and an image with a secondary star at 18 px distance. The top image shows the performance for faint targets while the bottom image depicts the bright case.

3.2.5 Presence of hot and dead pixels

In this section, the performance of the Tracking method on images with hot pixels and defective areas of the detector is covered. Here, hot pixels describe pixels with an increased dark current (Freudling, 1995), while dead pixels are defined as pixels that show no response to the incoming photons. While hot pixels are an isolated effect, dead pixels can occur in larger clusters, which will be referred to as dead regions in this analysis.
Both types of detector error can have an impact on the performance of centroiding algorithms. In the case of hot pixels, the singular hotspot can heavily influence the result of the IWC and cause offsets for the IWCoG, if the defect is inside of the PSF. On the other hand, dead regions cause a loss of total signal, if they clip the PSF and therefore also lead to offsets and larger errors, depending on the size and position relative to the target.
In the case of the ARIEL FGS, these detector errors are documented inside of a map that contains the position of the error. This enables us to perform a simple correction for these effects. This is done by replacing the value of the defective pixel with the median of the eight neighbours. By using this process, one is able to reduce the need for the optional median filter.

3.2.5.1 Hot Pixels

In this analysis, the effect of hot pixels inside of the PSF is studied. The initial test consists of 8000 simulated images for both the faint and the bright case with a signal increase after 200 images. Each image contains a hot pixel with a signal value of 15000 ADU in the center of the PSF. Additionally, both of the simulation runs were performed without the hot pixel in order to provide a baseline.
The FCU simulator was then used to perform the Tracking algorithm on all datasets. The images containing the hot pixel were both used with and without the median correction applied. The resulting plot is shown in Figure 3.20.

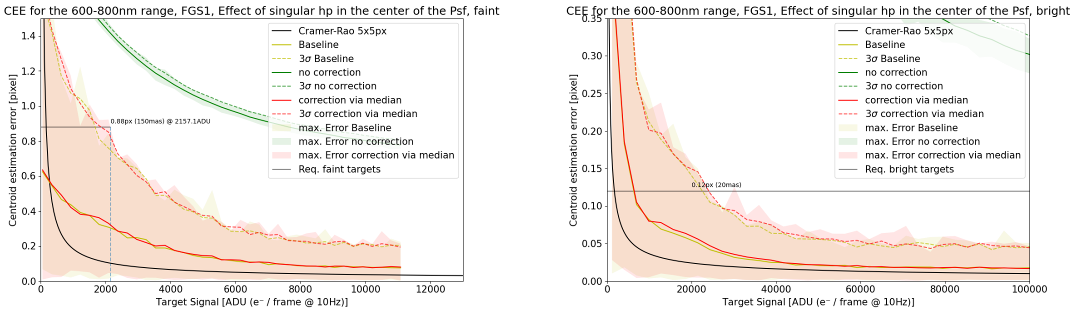


Figure 3.20: CEE plots for the central hot pixel case. The yellow line shows the baseline performance. The hot pixel case is depicted by the green (no correction) and the red line (median correction applied).

Based on the plot one can see that the correction method allows us to remove these features without any noticeable changes in performance. In order to further study this effect, a dataset with nine hot pixels that are randomly distributed inside of the ROI was simulated. The same set of simulations is then run with the median correction method applied.
In this analysis it was apparent that not all hot pixels where sufficiently corrected. Upon further inspection, the first entry of the hot pixel map contained four defective pixels in its neighbourhood. Given that one calculate the median of the eight neighbours, this leads to an insufficient correction as the median is interpolated between a hot pixel and the maximum of the local noise. This effect can be mitigated by sorting the hot pixel map in a way that a majority of the neighbouring pixels is corrected beforehand.
This method proved to correct all pixels sufficiently, so that one doesn't encounter any additional APE. However, a small increase in the AKE can be observed, which requires a new offset correction.

3.2.5.2 Dead Pixels

In this section, the influence of extended regions of dead pixels is covered. Dead pixels have a lower impact on centroiding performance since they don't result in large errors in the IWC step. They do however cause errors in the thresholding of the image if they are not corrected. This can lead to an increase in the APE as the influence of the noise is significantly larger.
In order to analyse the influence of the aforementioned dead regions, a datasets with varying region sizes at different distances to the center of the target star was created.
The first test case covered a dead region with a size of 2×2 px or 3×3 px that is positioned at the upper right corner of the PSF, covering a quarter of the target. These images were then used inside the FCU Simulator with the previously mentioned correction method and a basic replacement with the mean of the image.

One can see in Figure 3.21 that the presence of a dead region in the corner of the image has little to no impact on the APE. However, a small increase in AKE can be seen in both cases, where the larger offset error is caused by the bigger region. This is not present in the plots, as these only cover the APE.
In the next dataset, a closer look is taken at the influence of such a region in the center of the target star.

Here, one can start to see increases in the APE. In addition, the introduced offset in the AKE is substantial and well above a pixel in the 3×3 px case. Based on this result,

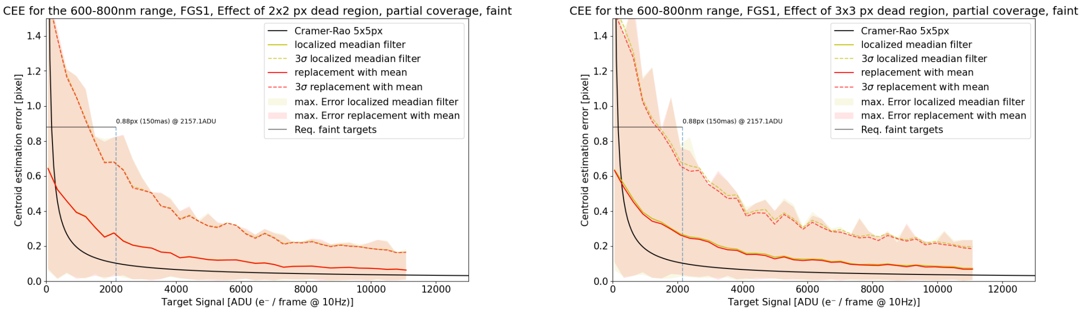


Figure 3.21: CEE plots for the case of a dead region covering a quarter of the PSF. The left plot shows the case with a 2×2 px region while the right image shows the impact of the 3×3 px case. The red line denotes the replacement with the mean, while the yellow line shows the method of the localised median filter.

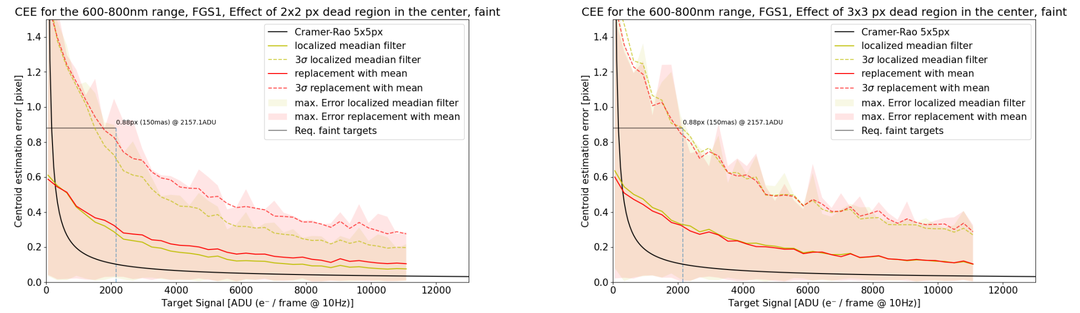


Figure 3.22: CEE plots for the case of a dead region in the center of the PSF. The left plot shows the case with a 2×2 px region while the right image shows the impact of the 3×3 px case. The red line denotes the replacement with the mean, while the yellow line shows the method of the localised median filter.

the use case where a dead region is positioned at the same position as the target star is unsuited for the Tracking mode, which has also been used as a basis for further requirements.

The last analysis was a baseline test with a dead region being present in the ROI but not overlapping with the PSF. The results are given in Figure 3.23.

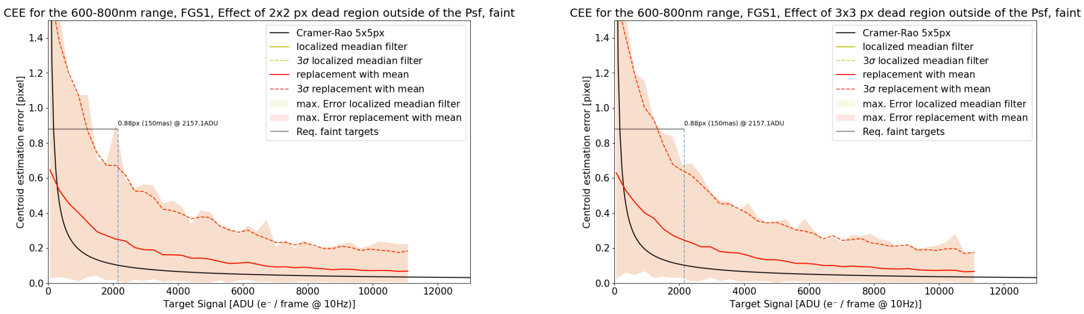


Figure 3.23: CEE plots for the case of a dead region inside the ROI without any interaction with the PSF. The left plot shows the case with a 2×2 px region while the right image shows the impact of the 3×3 px case. The red line denotes the replacement with the mean, while the yellow line shows the method of the localised median filter.

In this case one can see no increase in both APE and AKE. This means that the dead regions only have an impact on centroiding performance, if the region interacts with the target star in any way. Furthermore, one can see that smaller overlaps only lead to minor offset errors, but don't impact the performance of the overall pointing stability.

3.3 Acquisition Performance

3.3.1 Baseline Performance

The baseline performance of the Acquisition algorithm was tested on two datasets consisting of 10000 images each for both channels. These datasets represent the faint and the bright target star case, respectively. Similar to the datasets used for the Tracking performance tests, the signal is increased after 200 images. The increment is 50000 photons/s for the bright datasets, 2500 photons/s for the faint case of FGS1 and 5000 photons/s for the faint range of FGS2.
The operational parameters used for the performance testing of the Acquisition algorithm are given in Table 3.6.

Table 3.6: Simulation parameters for the Acquisition algorithm

Parameter	Value
ROI	200×200 px
Thresholding	Minimum
Binnsize	3×3 px
Target Pixel count	5
Target Signal	taken from simulation
Sigma Threshold	290 ADU (faint), 400 ADU (bright)
Extension Threshold	3
CoG ROI	64×64 px
Weighting mode	Approximation
Iteration Count	3

In the first set of simulations it was tested if the algorithm achieves the required baseline performance of a maximum error of 200 mas. The results are shown in Figure 3.24.

Based on the results of this initial analysis, one can see that the Acquisition algorithm performs adequate for the faint case of FGS1. There are some minor outliers in the centroid estimation but the majority lies within the tolerance. This behaviour is the same with the Tracking algorithm which performs slightly better under low signal cases due to the increased iteration count.
It is to note that the algorithm suffers from wrong source detections in cases where the signal is even lower. In this case, the sigma threshold has to be set quite low, which leads to the source extraction starting to return local noise maxima as stars. If one then increases the threshold by a minor amount, the source detection will fail to detect any

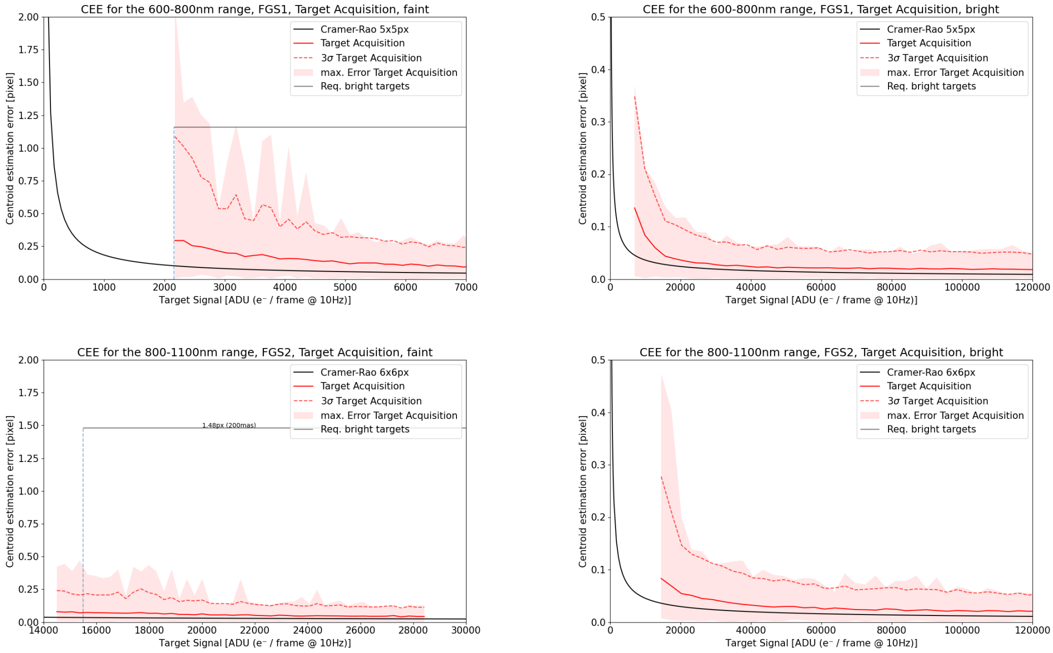


Figure 3.24: CEE plots for the baseline performance of the Acquisition algorithm using the settings given in Table 3.6. The two images on the top show the performance for the faint (left) and the bright (right) case of FGS1. The lower images show the analogue for FGS2. The bright cases are limited to 0.5 px in the y-axis in order to give a better overview of the performance. The requirement is the same as with the faint case.

star. However, this behaviour is an unrealistic case where the results wouldn't be used for the acquisition process as the Centroid validation will always return a "no star inside image" error for these low signals.

For all other cases, one can see that the algorithm performs in a similar manner as the Tracking algorithm with some small deviations due to the aforementioned lower iteration count. This behaviour leads to a substantial safety margin for cases with brighter signals as the algorithm outperforms the requirement by a factor of 5 to 10.

3.4 Timing Performance

3.4.1 Tracking

In this section, a discussion about the limitations on computation time for the centroiding algorithm is given and the time needed for a centroid computation with a varying amount of iterations is analysed.
As discussed in section 2.1, the requirements state that the FGS must provide centroiding data at a rate of either 8 Hz or 10 Hz, where this time-frame also includes the exposure time, the detector readout and the data transfer. It can roughly be estimated that the time available for the calculation of the centroid amounts to 20 ms for the 10 Hz case.
In order to measure the computation time of the centroiding, timing tests using the CHEOPS EM on ten images with an increasing signal were performed . The processes tested in the scope of this analysis consist of functions used for the centroiding process such as the IWC and IWCoG algorithms and the basic WCoG and CoG processes. It is to note, that the entire analysis is based on the algorithm using 5 iterative steps following the initial IWC result.
As an addition to the performance of the algorithm itself, the effect of the different weighting functions and the impact of larger ROIs on the processing time was also studied. The weighting functions tested consist of the previously mentioned linear approximation of a two-dimensional Gaussian, as well as an analytical Gaussian. With regards to the ROI, sizes of both 17×17 px and 31×31 px were tested.
The results of this analysis are shown in Table 3.7 which depicts the maximum of the processing time needed within the sample of the ten images. It is to note, that the results were scaled from the actually measured numbers due to the fact, that the GR712RC processor runs at 25 MHz in CHEOPS compared to the 40 MHz used in ARIEL.

Table 3.7: Results for the timing analysis. The first line (Centroiding) is the overall duration. The other lines are not supposed to exactly add up to that number.

Function	approx. Gauss		analytical Gauss	
	17 px [ms]	31 px [ms]	17 px [ms]	31 px [ms]
Centroiding	8.813	23.625	13.500	43.125
Weighting Function	1.369	3.344	2.275	7.313
CoG	0.126	0.454	0.126	0.349
IWC	0.381	1.194	0.383	1.075
IWCoG (5 iter.)	7.750	20.813	12.438	40.500

It can be seen in Table 3.7 that the algorithms using a ROI of 17×17 px lie below the 20 ms given by the requirements, whereas a ROI of 31×31 pixels is just above. If the design of the the algorithm is taken into account where the IWC is used the larger ROI (optionally corrected for glitches) and the iterations are performed on the refined ROI, the execution time totals up to 15 ms.

Figure 3.25: Execution time of various functions used in the tracking process on a 17×17 px ROI for both an analytical and approximated weighting function.

Figure 3.26: Execution time of various functions used in the tracking process on a 31×31 px ROI for both an analytical and approximated weighting function.

3.4.2 Acquisition

The timing tests for the Acquisition algorithm use the same setup as the timing measurements for the Tracking algorithm. In this case, only the source extraction and target identification steps of the target acquisition is tested as the last step for inter-pixel precision is identical to the Tracking method with a reduced iteration count for the IWCoG step.

As stated in Section 2.1 (R-GENPERF-8/P), the Acquisition method shall support a sampling rate larger or equal to 1 Hz. This is much more relaxed when compared to the timing restrictions for the mode Tracking but the method should still be´ as fast as possible in order to reduce the added time, if follow-up measurements are necessary.

The results of the timing analysis for the target acquisition are given in Table 3.8.

Table 3.8: Results for the timing analysis of the Acquisition method. The method was configured to find the five brightest pixels inside of the binned image. The initial image size is 200×200 px. The refined ROI has a size of 63×63 px.

Function	Execution time [ms]
Acquisition w/o centroiding	19.10
Binning	14.91
Source Extraction	1.36
Identify Star	0.082
Extract ROI	1.32

The baseline execution time for the target acquisition amounts to roughly 20 ms. If one adds a rough estimate for the Tracking method of 20 ms, it adds up to a total measurement time of 40 ms. This allows for a sampling rate of well over 1 Hz. It is to note, that thanks to a lower execution time, it is possible to increase the integration time, which would result in larger signals for faint cases. As a result, one would see an increase in pointing precision.

4. Conclusion

In the context of this thesis, the current development status of the algorithms implemented for the pointing correction of the ARIEL FGS instrument has been presented. Both the algorithm presented for the continuous guiding process and the target acquisition have proven to be sufficient for the demanding photometric stability criteria given by the ARIEL science case.

In the case of the guiding algorithm, a combined method of an initial IWC step for a rough positional estimation followed by an IWCoG method has been proposed. This method has proven quite promising in both the processing time and precision. The algorithm replaces the first half of the IWCoG steps with a fast IWC, which drastically cuts down on the execution time. This is further reduced by the use of a ROI refinement between the two processes. This results in the performance that is comparable to an IWCoG with 10 iterations with the computational time of approximately 6 iterative steps.
The performance of this algorithm has been tested in multiple cases, where it has shown sufficient precision in almost every case. The exception to this is the case of a faint target for the channel FGS1. Here, the incoming signal of the target star in the given wavelength range is significantly lower when compared to the other FGS channel. This leads to a worst case scenario where the performance is still adequate, but the margin is significantly smaller. This could lead to problems if multiple effects like smearing and jitter further reduce the signal to noise ratio by spreading the signal over a larger area.

The algorithm used for the target acquisition builds on the algorithm used in the guiding process. Here, the guiding method is used in order to gain sub-pixel precision after an initial source extraction. The source extraction is performed on a binned image by searching for the n brightest pixels inside of the image and determining the extent and signal of each. If the found target is an extended source and has a signal that's within a given margin, a small ROI is cut from the original, unbinned image and the guiding algorithm is performed with 3 IWCoG iterations.
in regard to precision and computational cost, the algorithm shows satisfactory performance. Similar to the guiding algorithm, the faint targets of FGS1 tend to be a borderline case for the pointing precision of the algorithm, but in all other cases, the algorithm provides a large safety margin when compared to the requirement. Furthermore, the achievable sampling rate is well above the required 1 Hz.

As a last point it is to mention that this is an early performance and design study of the algorithms. Since the mission is scheduled to start in 2029, there are still a lot of potential improvements and changes which will occur over the next years. In this context, some planned work is presented here which will be done in the following years.

- In the course of this analysis it became apparent that the current version of the simulator suite needs an update. This is due to the fact, that both the FCE and Data Simulator had to undergo significant modifications in order to be used for the FGS. Therefore, a new simulator suite is planned, which should be more adaptive and designed for easier modification for different instruments in order to reduce the workload for future missions. This is already important in the case of the ARIEL NIRSpec science data processing, since Starsim doesn't support the creation of spectroscopic data yet.
- As discussed in the performance section of the thesis, the algorithms are tested in regard to instantaneous performance. However, it is necessary to test these algorithms in a long term, closed-loop environment, which requires further knowledge about the design of the AOCS and the behaviour of the correction loop. These tests will be performed as soon as these factors are known.

Bibliography

(2008). ECSS-E-ST-60-10C. ESA Requirements and Standards Division.

Akondi, V. and Vohnsen, B. (2013). Myopic aberrations: Impact of centroiding noise in hartmann shack wavefront sensing. Ophthalmic and physiological optics : the journal of the British College of Ophthalmic Opticians (Optometrists), 33:434–443.

Apai, D., Radigan, J., Buenzli, E., Burrows, A., Reid, I. N., and Jayawardhana, R. (2013). HST Spectral Mapping of L/T Transition Brown Dwarfs Reveals Cloud Thickness Variations. apj, 768(2):121.

ARIEL Science Study Team (2017). ARIEL Assessment Study Report, ESA/SCI(2017)2 Issue 2.0.

ARIEL Science Study Team (2020). ARIEL Payload Requirements Document, Issue 4.0.

Baker, K. L. and Moallem, M. M. (2007). Iteratively weighted centroiding for shack-hartmann wave-front sensors. Opt. Express, 15(8):5147–5159.

Charbonneau, D., Knutson, H. A., Barman, T., Allen, L. E., Mayor, M., Megeath, S. T., Queloz, D., and Udry, S. (2008). The Broadband Infrared Emission Spectrum of the Exoplanet HD 189733b. apj, 686(2):1341–1348.

Eccleston, P. (2017). ARIEL Payload Design Description.

Ferstl, R. (2016). Optimised cenrtroiding of stars for space applications. Master's thesis, University of Vienna, Institute for Astrophysics.

Freudling, W. (1995). WFPC 2 hot pixels and image restoration. Space Telescope European Coordinating Facility Newsletter, 22:20.

Knutson, H. A., Charbonneau, D., Noyes, R. W., Brown, T. M., and Gilliland, R. L. (2007). Using Stellar Limb-Darkening to Refine the Properties of HD 209458b. apj, 655(1):564–575.

Lisa A. Poyneer, Kai La Fortune, C. C. (2003). Scene-based wavefront sensing for remote imaging. Proc.SPIE, 5162:5162 – 5162 – 12.

Loeschl, P., Ferstl, R., Kerschbaum, F., and Ottensamer, R. (2016). On-board target acquisition for CHEOPS. In Chiozzi, G. and Guzman, J. C., editors, Software and Cyberinfrastructure for Astronomy IV, volume 9913 of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, page 99132W.

Majeau, C., Agol, E., and Cowan, N. B. (2012). A Two-dimensional Infrared Map of the Extrasolar Planet HD 189733b. apjl, 747(2):L20.

Ottensamer, R., Rataj, M., Schrader, J.-R., Ferstl, R., Güdel, M., Kerschbaum, F., and Luntzer, A. (2014). EChO fine guidance sensor design and architecture. In Space Telescopes and Instrumentation 2014: Optical, Infrared, and Millimeter Wave, volume 9143 of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, page 91434O.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (2007). Numerical Recipes, Third Edition. Cambridge University Press.

Rataj, M., Wawer, P., Skup, K., and Sobiecki, M. (2019). Design of fine guidance system (FGS) for ARIEL mission. In Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2019, volume 11176 of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, page 111763E.

Roh, K.-W., Uhm, T.-K., Kim, J.-Y., Youn, S.-K., and Lee, J. H. (2008). Noise-insensitive centroiding algorithm for a shack-hartmann sensor. J. Korean Phy. Soc., 52(1):160–169.

Sing, D. (2018). Observational Techniques with Transiting Exoplanetary Atmospheres. Astrophysics of Exoplanetary Atmospheres. Astrophysics and Space Science Library, Springer, 450.

T Ott, A Benoit, P. V. d. B. (2011). ESA pointing error engineering handbook, ESSB-HB-E-003-Issue1.

Tinetti, G., Drossart, P., Eccleston, P., Hartogh, P., Leconte, J., Micela, G., Ollivier, M., Pilbratt, G., Puig, L., Heske, A., Turrini, D., Vandenbussche, B., Wolkenberg, P., and Ariel Consortium (2017). The science of ARIEL. In European Planetary Science Congress, pages EPSC2017–713.

Tinetti, G., Eccleston, P., Haswell, C., Lagage, P.-O., Leconte, J., Lüftinger, T., Micela, G., Min, M., Pilbratt, G., Puig, L., Swain, M., Testi, L., Turrini, D., Vandenbussche, B., Osorio, M. R. Z., Aret, A., Beaulieu, J.-P., Buchhave, L., Ferus, M., Griffin, M., Guedel, M., Hartogh, P., Machado, P., Malaguti, G., Pallé, E., Rataj, M., Ray, T., Ribas, I., Szabó, R., Tan, J., Werner, S., Ratti, F., Scharmberg, C., Salvignol, J.-C., Boudin, N., Halain, J.-P., Haag, M., Crouzet, P.-E., Kohley, R., Symonds, K., Renk, F., Caldwell, A., Abreu, M., Alonso, G., Amiaux, J., Berthé, M., Bishop, G., Bowles, N., Carmona, M., Coffey, D., Colomé, J., Crook, M., Désjonqueres, L., Díaz, J. J., Drummond, R., Focardi, M., Gómez, J. M., Holmes, W., Krijger, M., Kovacs, Z., Hunt, T., Machado, R., Morgante, G., Ollivier, M., Ottensamer, R., Pace, E., Pagano, T., Pascale, E., Pearson, C., Pedersen, S. M., Pniel, M., Roose, S., Savini, G., Stamper, R., Szirovicza, P., Szoke, J., Tosh, I., Vilardell, F., Barstow, J., Borsato, L., Casewell, S., Changeat, Q., Charnay, B., Civiš, S., du Foresto, V. C., Coustenis, A., Cowan, N., Danielski, C., Demangeon, O., Drossart, P., Edwards, B. N., Gilli, G., Encrenaz, T., Kiss, C., Kokori, A., Ikoma, M., Morales, J. C., Mendonça, J., Moneti, A., Mugnai, L., Muñoz, A. G., Helled, R., Kama, M., Miguel, Y., Nikolaou, N., Pagano, I., Panic, O., Rengel, M., Rickman, H., Rocchetto, M., Sarkar, S., Selsis, F., Tennyson, J., Tsiaras, A., Venot, O., Vida, K., Waldmann, I. P., Yurchenko, S., Szabó, G., Zellem, R., Al-Refaie, A., Alvarez, J. P., Anisman, L., Arhancet, A., Ateca, J., Baeyens, R., Barnes, J. R., Bell, T., Benatti, S., Biazzo, K., Błęcka, M., Bonomo, A. S., Bosch, J., Bossini, D., Bourgalais, J., Brienza, D., Brucalassi, A., Bruno, G., Caines, H., Calcutt, S., Campante, T., Canestrari, R., Cann, N., Casali, G., Casas, A., Cassone, G., Cara, C., Carmona, M., Carone, L., Carrasco, N., Changeat, Q., Chioetto, P., Cortecchia, F., Czupalla, M., Chubb, K. L., Ciaravella, A., Claret, A., Claudi, R., Codella, C., Comas, M. G., Cracchiolo, G., Cubillos, P., Peppo, V. D., Decin, L., Dejabrun, C., Delgado-Mena, E., Giorgio, A. D., Diolaiti, E., Dorn, C., Doublier, V., Doumayrou, E., Dransfield, G., Dumaye, L., Dunford, E., Escobar, A. J., Eylen, V. V., Farina, M., Fedele, D., Fernández, A., Fleury, B., Fonte, S., Fontignie, J., Fossati, L., Funke, B., Galy, C., Garai, Z., García, A., García-Rigo, A., Garufi, A., Sacco, G. G., Giacobbe, P., Gómez, A., Gonzalez, A., Gonzalez-Galindo, F., Grassi, D., Griffith, C., Guarcello, M. G., Goujon, A., Gressier, A., Grzegorczyk, A., Guillot, T., Guilluy, G., Hargrave, P., Hellin, M.-L., Herrero, E., Hills, M., Horeau, B., Ito, Y., Jessen, N. C., Kabath, P., Kálmán, S., Kawashima, Y., Kimura, T., Knížek, A., Kreidberg, L., Kruid, R., Kruijssen, D. J. M., Kubelík, P., Lara, L., Lebonnois, S., Lee, D., Lefevre, M., Lichtenberg, T., Locci, D., Lombini, M., Lopez, A. S., Lorenzani, A., MacDonald, R., Magrini, L., Maldonado, J., Marcq, E., Migliorini, A., Modirrousta-Galian, D., Molaverdikhani, K., Molinari, S., Mollière, P., Moreau, V., Morello, G., Morinaud, G., Morvan, M., Moses, J. I., Mouzali, S., Nakhjiri, N., Naponiello, L., Narita, N., Nascimbeni, V., Nikolaou, A., Noce, V., Oliva, F., Palladino, P., Papageorgiou, A., Parmentier, V., Peres, G., Pérez, J., Perez-Hoyos, S., Perger, M., Pestellini, C. C., Petralia, A., Philippon, A., Piccialli, A., Pignatari, M., Piotto, G., Podio, L., Polenta, G., Preti, G., Pribulla, T., Puertas, M. L., Rainer, M., Reess, J.-M., Rimmer, P., Robert, S., Rosich, A., Rossi, L., Rust, D., Saleh, A., Sanna, N., Schisano, E., Schreiber, L., Schwartz, V., Scippa, A., Seli, B., Shibata, S., Simpson, C., Shorttle, O., Skaf, N., Skup, K., Sobiecki, M., Sousa, S., Sozzetti, A., Šponer, J., Steiger, L., Tanga, P., Tackley, P., Taylor, J., Tecza, M., Terenzi, L., Tremblin, P., Tozzi, A., Triaud, A., Trompet, L., Tsai, S.-M., Tsantaki, M., Valencia, D., Vandaele, A. C., der Swaelmen, M. V., Vardan, A., Vasisht, G., Vazan, A., Vecchio, C. D., Waltham, D., Wawer, P., Widemann, T., Wolkenberg, P., Yip, G. H., Yung, Y., Zilinskas, M., Zingales, T., and Zuppella, P. (2021). Ariel: Enabling planetary science across light-years.

Tinetti, G., Vidal-Madjar, A., Liang, M.-C., Beaulieu, J.-P., Yung, Y., Carey, S., Barber, R. J., Tennyson, J., Ribas, I., Allard, N., Ballester, G. E., Sing, D. K., and Selsis, F. (2007). Water vapour in the atmosphere of a transiting extrasolar planet. Nature, 448(7150):169–171.

Tremsin, A. S., Vallerga, J. V., Siegmund, O. H. W., and Hull, J. S. (2003). Centroiding algorithms and spatial resolution of photon counting detectors with cross-strip anodes. In Siegmund, O. H. W., editor, UV/EUV and Visible Space Instrumentation for Astronomy II, volume 5164 of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, pages 113–124.

Vyas, A., Roopashree, M. B., and Prasad, B. R. (2010). Noise reduction in the centroiding of laser guide star spot pattern using thresholded Zernike reconstructor. In Ellerbroek, B. L., Hart, M., Hubin, N., and Wizinowich, P. L., editors, Adaptive Optics Systems II, volume 7736, pages 1556 – 1565. International Society for Optics and Photonics, SPIE.

Willert, C. E. and Gharib, M. (1991). Digital particle image velocimetry. Experiments in Fluids, 10(4):181–193.

Winick, K. A. (1986). Cramér–rao lower bounds on the performance of charge-coupled-device optical position estimators. J. Opt. Soc. Am. A, 3(11):1809–1815.

Yu, P., Yang, Y., and Gan, Y. (2012). Experiment analysis on the bubble sort algorithm and its improved algorithms. In Zhu, R. and Ma, Y., editors, Information Engineering and Applications, pages 95–102, London. Springer London.

Zellem, R. T., Swain, M. R., Cowan, N. B., Bryden, G., Komacek, T. D., Colavita, M., Ardila, D., Roudier, G. M., Fortney, J. J., Bean, J., Line, M. R., Griffith, C. A., Shkolnik, E. L., Kreidberg, L., Moses, J. I., Showman, A. P., Stevenson, K. B., Wong, A., Chapman, J. W., Ciardi, D. R., Howard, A. W., Kataria, T., Kempton, E. M. R., Latham, D., Mahadevan, S., Meléndez, J., and Parmentier, V. (2019). Constraining Exoplanet Metallicities and Aerosols with the Contribution to ARIEL Spectroscopy of Exoplanets (CASE). pasp, 131(1003):094401.