Semester thesis

# An algorithm for exoplanet search with the HPP telescope

Thomas Birbacher

May 24, 2019

Advisers: Silvan Hunziker, Prof. Dr. Sascha P. Quanz

## Abstract

The most successful method of exoplanet detection is the transit method, where the light curve of a star is examined to find periodic drops caused by an orbiting body. In this thesis, an algorithm for exoplanet search and the extraction of orbital parameters is presented. The `Python` routine takes as input a time series of CCD images of the host star and performs dark- and flat-correction, astrometry, photometry, decorrelation of the extracted light curve, transit-detection, and an MCMC-fit to find transit parameters such as the period and semi-major axis of the orbit and radius of the planet. The algorithm is tested with observations of TrES-3b from the 0.5 m telescope at ETH Hönggerberg and finds that most parameters agree within 15 % with the existing research. The biggest limitation in finding the parameters is that only single-transit observations are available from this telescope.

## Contents

# 1. Introduction

There are four major methods to detect extrasolar planets: the radial velocity method, the transit method, the direct imaging method, and the microlensing method. This work focuses on the transit method, which is comparatively easy to perform: A planet transiting its host star can be detected by a drop in its light curve, which is the star's flux as a function of time. [5]

This work builds on the observations of A. Bohn [4] and A. Gheorghe [7], who were the first to detect exoplanet transits with the 0.5 m telescope at ETH Hönggerberg and developed a `Python` routine for data analysis. In particular, they showed that transits can be detected even under the light polluted skies in Zürich.

The aim of this thesis is to improve the quality and speed of the data analysis. It now includes the calculation of the astrometric solution of every image to facilitate the tracking of stars, aperture photometry with an automatic choice of aperture, extraction of the light curves of all stars that are visible in the image, detrending of the light curves with principal component analysis, an automatic search for exoplanet transits in the light curves, and the extraction of several transit parameters, such as radius of the planet, period of the transit, and the length of the semi-major-axis of the orbit, using an MCMC sampler.

The algorithm is tested with observations of a transit of TrES-3b. Except for the period and the semi-major axis length, all the fitted transit parameters agree with the existing research.

# 2. Theory

A simple model for the light curve of a planetary transit uses the following parameters:

- the time of inferior conjunction $t_0$,
- the orbital period $P$,
- the planet radius in units of host-star radii $R_p/R_\star$,
- the semi-major axis in units of host-star radii $a/R_\star$,
- the orbital inclination $i$ or the impact parameter $b$,
- the orbital eccentricity $e$,
- and the argument of periastron $\omega$.

These values suffice to determine the light curve of a non-radiating body in an orbit around a star with uniform brightness. However, as seen from an outside observer, most stars are significantly brighter at the centre of the visible disk than at the outside. This fall-off in brightness is called limb-darkening and can be modelled with the light curve by supplying a set of limb-darkening parameters $\{u_1, \dots, u_n\}$. For this work, a quadratic limb-darkening model [10] is used, because it can reproduce the light curve without requiring many parameters that could lead to numerical instability.

# 3. Algorithm

## 3.1. Overview

The Python program presented in this work consists of multiple parts, which are mostly handled by already existing software using established and well-tested algorithms. The light frames are corrected with dark and flat frames, and their astrometric solutions is computed to track the stars between images. Aperture photometry is performed to extract the light curves, and systematic errors of the light curve are removed with principal component analysis. Transits are found with a quick gradient-descent fit, and orbital parameters extracted with an MCMC sampler.

## 3.2. Preparation

The `FITS` files are read and edited with `astropy` [1, 2]. Light frames are corrected with flat and dark frames according to

$$\text{corrected light} = \frac{\text{light} - \text{dark}}{\langle\text{flat}\rangle},$$

where the angle brackets denote that the flat frame has been normalized. To reduce noise, multiple dark and flat frames are median-stacked.

### 3.3. Astrometry

To track the stars between different frames, the astrometric solution is computed for every frame. This is done with the `astrometry.net` solver [13], which, despite its name, does not require internet access and can run locally. It was chosen both for its speed, which is mostly limited by the recognition of sources in the image, and its great reliability. The solver requires a catalogue of stars, called an index file, to compare the image to. Depending on the aperture and field of view of the telescope, different index files can be downloaded from the `astrometry.net` website.

The stars that the solver detects are written into a database which is later used to indicate to the photometry-routine where to look for stars. This reduces the risk of accidentally detecting galaxies, comets, cosmic rays, or other unwanted sources as stars.

### 3.4. Photometry

A light curve is the stellar flux as a function of time. To find the fluxes of all stars in the database in each image, `SExtractor` [3] is used. The source extractor automatically determines an aperture size for each star by fitting an elliptical Gaussian profile to the source and determining the optimal aperture radius based on the Kron radius [12]. The influence of different aperture sizes on the light curve is examined in appendix B. The flux is only obtained for stars in the database to avoid generating light curves of unwanted sources.

### 3.5. Decorrelation

Due to atmospheric extinction, wind, changing humidity, light pollution, and other effects, the raw light curves found by `SExtractor` are heavily influenced by systematic errors. To remove these systematic errors, it is assumed that all stars in the image are affected by the same effects to an equal extent. While this assumption is not exactly true[1], a significant part of the systematic trend can be removed with principal component analysis. This part of the algorithm is adapted from Kepler's Presearch Data Conditioning [19] routine and consists of two steps.

First, a set of cotrending basis vectors is generated, which are light curves that represent the principal components of the systematic errors. For this, all light curves are classified according to their variability, and only "quiet" stars that exhibit low variability are used. This excludes many eclipsing binaries and noisy stars with low fluxes. The correlation of these quiet light curves is calculated, and the most correlated light curves are then analysed with singular value decomposition. The singular vectors corresponding to the highest singular values are used as cotrending basis vectors.

Second, these cotrending basis vectors are fitted to each light curve with a least-squares gradient-descent fit. It was found that by fitting more than one cotrending

---

[1] For example, stars of different emission spectra undergo different extinction in the atmosphere, cosmic rays can effect a single star's flux on the image, and clouds can obscure isolated parts of the field of view.

basis vector to a light curve, some features of the light curve are lost by over-fitting. Still, most of the systematic errors are removed from the light curve by fitting only one basis vector. The remaining errors seem to be random, as can be seen by the residual in fig. 6. Depending on the quality of the data, more basis vectors can be fitted. The Kepler manual recommends to use eight basis vectors for their own observations.

If further systematic errors need to be removed without impairing features of the transit, I recommend to implement the maximum a-posteriori (MAP) approach [18] of Kepler's PDC-routine. It reduces the risk of overfitting the cotrending basis vectors to light curves with high intrinsic variability.

I initially attempted to use the Sys-Rem algorithm [15] to detrend the light curves, since it was easier to implement. However, Sys-Rem removes more intrinsic features of the light curve, while removing less of the systematic errors compared to the Kepler algorithm.

### 3.6. Transit detection

A common method of transit search involves light curves that were captured regularly over long time-scales, usually several days or weeks. These can be analysed with the Box Least Squares algorithm [11], which requires a periodic light curve. Since observations in Zürich are usually impaired by bad weather, and contain only a single transit event, this approach is not feasible.

In the detrended light curve, transits whose depth exceed the noise are detected with a gradient-descent least-squares fit that uses a uniform limb-darkening model. The gradient-descent fit is fast enough to analyse hundreds of light-curves within few seconds, but the fit is ill conditioned when using more complicated limb-darkening models. Therefore, this fit is only used to search for transits, not to find parameters of the transit.

### 3.7. Parameter fitting

To finally extract meaningful parameters of the transit, a quadratic limb-darkening law is used, because it successfully models the light curve in fig. 6, while not requiring many parameters that could slow down the fit or lead to numerical instability. To circumvent the problems associated with a gradient-descent fit, the global minimizer from `scipy.optimize.minimize` is used. The model also includes a Gaussian noise component.

To find the correlation between different parameters, refine the model parameters, and obtain a probability distribution of the parameters, the default MCMC-sampler of `PyMC3` [17, 20] is used. It is conveniently wrapped in the `exoplanet` module [6, 14], along with priors [9] that are tailored to the needs of planetary transit fitting.

The posterior distributions and their correlations are compared with values that are automatically downloaded from the Exoplanet Orbit Database [8].

# 4. Usage

## 4.1. Installation

In order to use the program for exoplanet search, several software packages have to be installed first. The `Python` routine requires `astrometry.net` and `SExtractor` to be configured on the system, such that they can be launched from the command line. The following guide is written for Ubuntu and similar Linux distributions. Since the source code for all the software that is used is publicly available, all programs could also be compiled to run on macOS. On Windows 10, the integrated Linux VM can probably be used to do this, but I do not know whether this solution works.

**Python** The program runs on `Python 3.6.8` and requires the following modules:

- `astropy 3.0.5` to handle the FITS files of the CCD-camera
- `astroquery 0.3.9` to obtain reference values for parameters of the transit for comparison with the fitted values
- `batman-package 2.4.6` to generate light curve models from transit parameters
- `exoplanet 0.1.5` to conveniently control the MCMC-fit
- `matplotlib 2.2.3` to generate plots
- `numpy 1.16.2` to efficiently perform calculations on arrays
- `pymc3 3.6` to perform MCMC-fits
- `scipy 1.1.0` to perform quick gradient-descent fits
- `corner 2.0.1` to plot the correlations of posteriors of the MCMC sampler

Different versions of `Python 3` or the modules may work, but have not been tested. All the above packages can be installed with `pip`.

**Astrometry** To identify and distinguish different stars in the light frames, the equatorial coordinates of each image are found with the `astrometry.net` software. Despite the name, a local version is installed that requires no internet access.

On Ubuntu, `astrometry.net` can be installed with

```
sudo apt-get install astrometry.net
```

Installers for Windows and macOS are also available. Instructions for building the solver from source are found at `https://astrometrynet.readthedocs.io/en/latest/build.html#build`.

The solver compares the sources found in the light frame with a reference catalogue, which is stored in index files. Depending on field of view of the image, different index files are needed. For the 40′ × 40′ field of view of the HPP telescope, the files in the `4200` directory are recommended.

On Ubuntu, the index files can be downloaded and installed following the instructions on `https://indilib.org/about/ekos/alignment-module.html`.

In case this fails, the index files can be downloaded from `http://data.astrometry.net/`. To download all the files from the `4200` directory, use

```
wget --recursive --no-parent http://data.astrometry.net/4200/
```

The downloaded index files must be copied to `/usr/share/astrometry`.

After a successful installation, the solver can be started from the shell with the command

```
solve-field image.fits
```

**Photometry** `SExtractor 2.19.5` is used to perform photometry. If `astrometry.net` was installed with `apt-get`, `SExtractor` is already installed.

Otherwise, it can be downloaded from `https://www.astromatic.net/software/sextractor`. The downloaded source contains the file `INSTALL`, which explains how to install it. After a successful installation, `SExtractor` can be launched from the shell with the command

```
sextractor image.fits
```

The program also requires the configuration files `default.conv`, `default.param`, and `default.sex` to be in the same directory. These are provided with the `Python` program and can be modified as needed, but the default settings should work well.

## 4.2. Running the program

Even though the program can execute all parts of the data analysis, except for the noise estimate, without user input, it is recommended to run through the program in small steps. At the least, the light curves of candidates and the results of the curve fit that generates the guess for the MCMC sampling should be checked by the user, since the MCMC sampling can take from several minutes to an hour. Intermediate results are saved in the `output` directory, and plots are saved in the `figures` directory.

**Preparation** Copy the light, dark, and flat frames of an observation into the directories `input/light`, `input/dark`, and `input/flat`. If the `FITS`-files do not contain a rough guess of the RA and DEC of the middle of the frame (accurate to within 1°) in the header, edit the `RAguess` and `DECguess` variables. This is required for images that were acquired with the HPP telescope before September 2018. If the `FITS` header contains RA and DEC, leave `RAguess` and `DECguess` as `None`.

**Photometry** The file `default.sex` contains the settings for photometry. A detailed description of all parameters can be found in the documentation of `SExtractor`. The default options are optimized for the HPP telescope. If another telescope is used, the following options should be adjusted:

- `DETECT_MINAREA`: The minimum number of pixels above threshold for a source to be detected. If the stars in the image are smaller than 5 pixels, this should be lowered.
- `SATUR_LEVEL`: The ADU level above which a pixels is considered as saturated.
- `BACK_SIZE`: The size of the background mesh in pixels. The width and height should be considerably larger than the diameter of every unsaturated star in the image.
- `DETECT_THRESH`: The minimum signal-to-noise ratio for a source to be detected.

The aperture diameter for photometry is determined automatically, but can be scaled manually by modifying the `PHOT_AUTOPARAMS` option. Different factors are explored in appendix B, but the default value of 2.5 should work well in most situations.

**Removing trends** The light curves of all stars are detrended using principal component analysis. A set of cotrending basis vectors is generated and fitted to each light curve. The best results are achieved by fitting only the highest-singular-value basis vector to each light curve, which is the default option in the code. Fitting more basis vectors increases the risk of removing important features of the light curves, such as exoplanet transits and stellar variability.

**Transit search** A simple gradient-descent fit with a uniform limb-darkening model is executed to find transits. A transit is only reported if its depth is larger than the noise of the light curve. After candidates are found and saved in `figures/candidates`, the user should check which of those should be used for further analysis. In particular, eclipsing binary stars are frequently detected as candidates, and should be excluded from further analysis.

**Finding the starting point for MCMC** Transit parameters are calculated with an MCMC sampler. A starting point for the sampling is generated with a `scipy.optimize` fit. After the fit, the result should be checked by the user to determine whether the fit was successful. If the fit has failed, the most important option to modify is the noise estimate, which should be within 10 % of the actual value.

**MCMC sampling** The MCMC sampler takes approximately 15 min to sample one light curve with the default settings on an i7-6700K processor. The algorithm can detect whether any problems occur during sampling, and proposes ways to mitigate them. After sampling, the user should check the diagnostic plots to assess the reliability of the

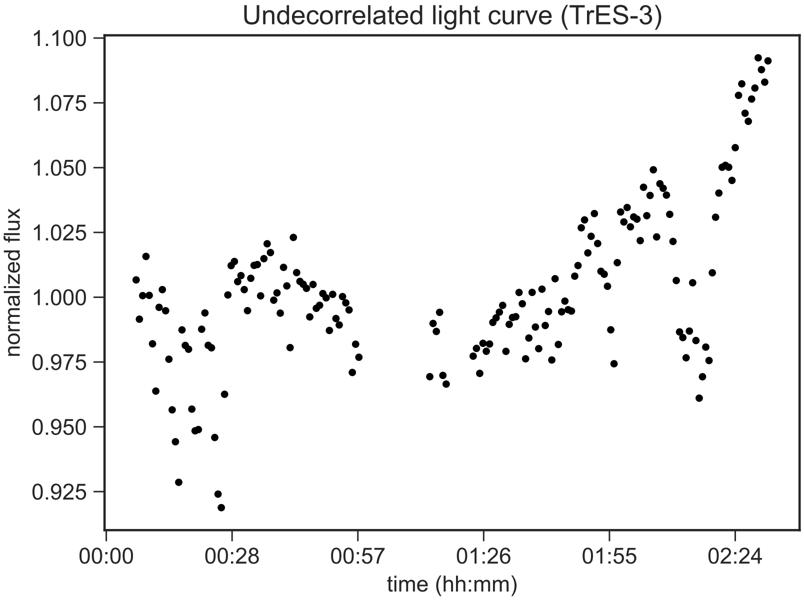

**Fig. 1:** The unmodified light curve of TrES-3. A transit during the middle of the observation is visible.

results. In particular, the trace-plots should show a "fuzzy caterpillar" shape centred around the median, and the different chains should have converged to approximately equal shape. Examples are discussed in appendix A.

## 5. Results

The algorithm is tested with transit observations of the exoplanets TrES-3b and HAT-P-44b from the HPP telescope at ETH Zürich.

### 5.1. TrES-3b

A single TrES-3b transit was captured by A. Gheorghe [7] in 2018. TrES-3 is a magnitude 12.4 star and hosts a confirmed exoplanet with a transit depth of 2.8 % [16].

The observation consists of 166 × 30 s frames from an optical CCD with no band-filter. 100 stars per frame were detected with SNR $>$ 40, and 33 were used to generate the cotrending basis vectors for detrending the light curves. Since no autoguider was used, TrES-3 drifted by approximately 300 pixels during the observation.

**Light curve extraction** The extraction of the light curves of all the stars in the images takes approximately 3 s per light frame. The resulting light curve of the candidate star TrES-3 is shown in fig. 1. The light curve shows high systematic errors, which are removed in the next step.

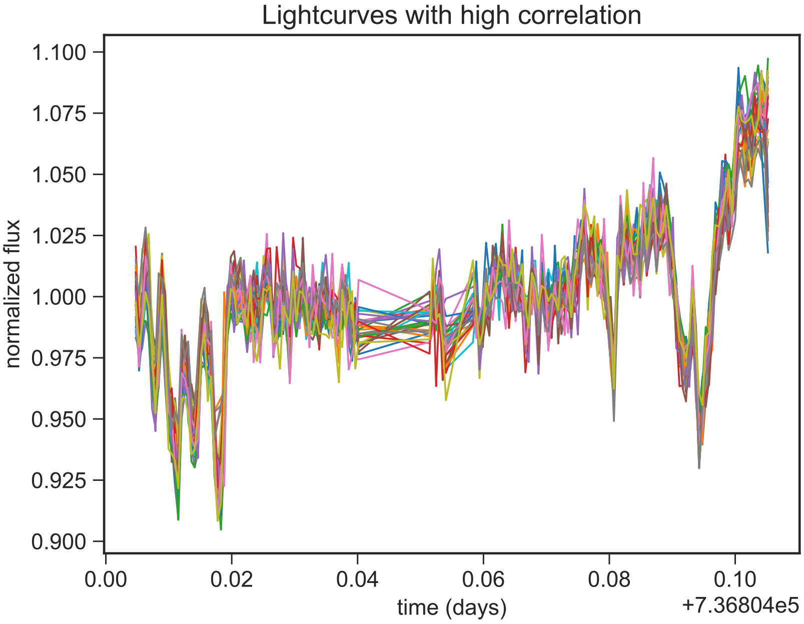

**Fig. 2:** The highly correlated "quiet" light curves that are used to generate the basis vectors for decorrelation.

**Decorrelation** To mitigate systematic errors that are shared by the light curves of most stars, the light curves are decorrelated using principal component analysis. The light curves used to generate the basis vectors for decorrelation are selected automatically and shown in fig. 2, and the decorrelated light curve of TrES-3 is shown in fig. 3. The transit is now clearly visible.

**Transit search** Every star's light curve is fitted to a transit model with uniform limb-darkening to find potential candidates for exoplanet hosts. Two candidates are found using this routine: One is the confirmed exoplanet host TrES-3, and the other is GSC 03089-01247 (fig. 4), which is an eclipsing binary star. Only the light curve of TrES-3 is used for further analysis.

**Parameter fitting** Parameters of the transit, such as orbital period, length of the semi-major axis, and transit depth, are obtained with an MCMC-sampler. Four chains with 3000 samples each were calculated. The posteriors for these parameters and their correlation are shown in fig. 5. In appendix A, the success and reliability of this sampling is analysed.

From the samples, the median, and the 10th and 90th percentiles are calculated and compared to the measured data in fig. 6.

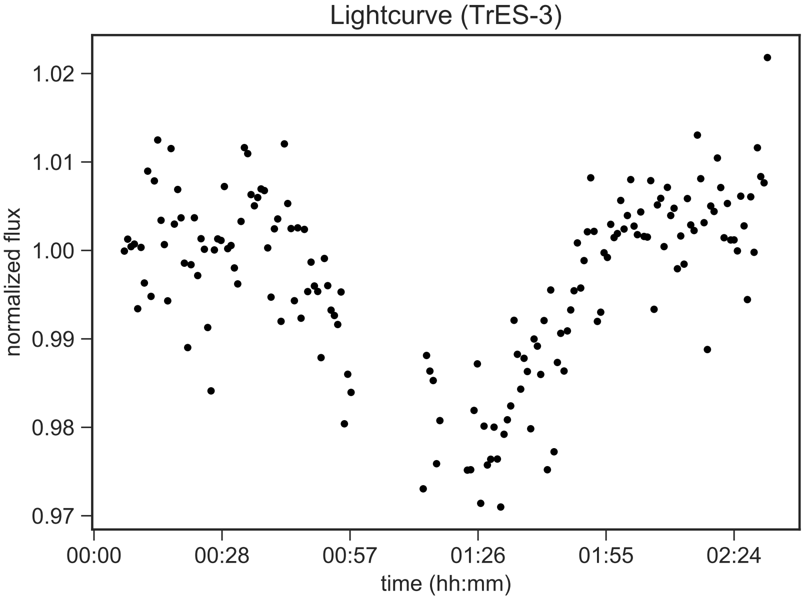

**Fig. 3:** The decorrelated light curve of TrES-3. The transit is clearly visible.

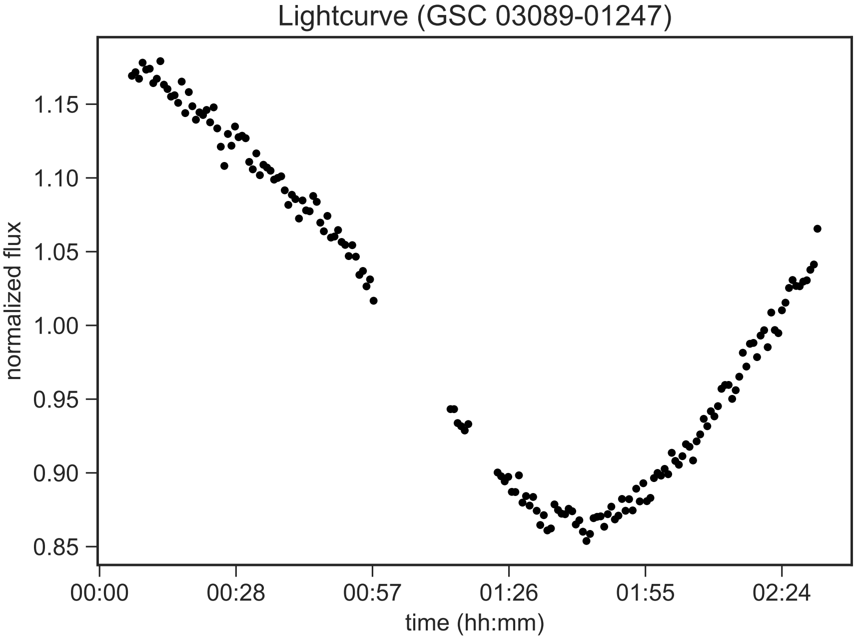

**Fig. 4:** Light curve of the eclipsing binary star GSC 03089-01247.

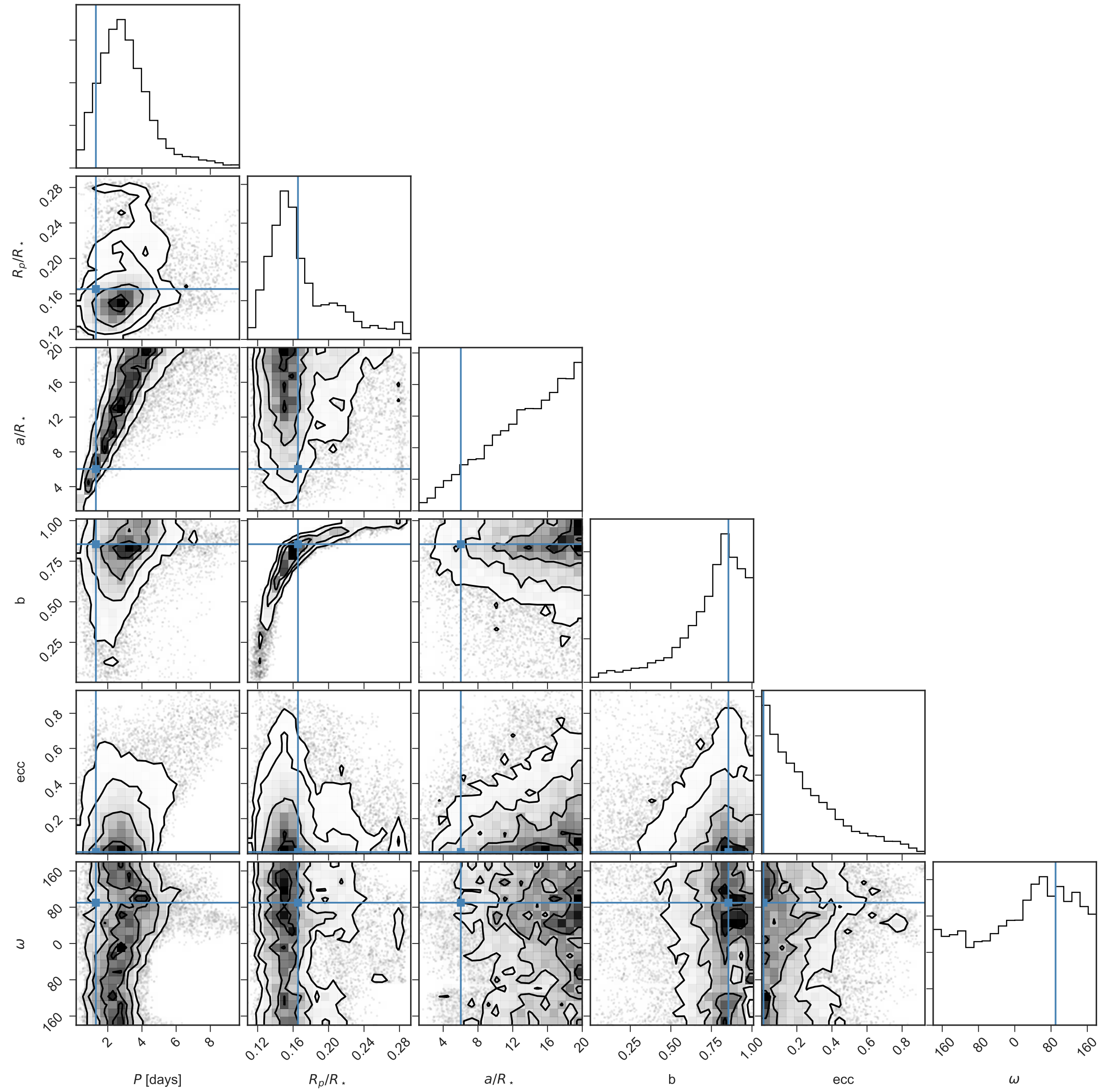

**Fig. 5:** Posterior distributions and correlation for the orbital parameters of TrES-3b. The reference values from the exoplanet orbit database are shown in blue. Except for the semi-major axis length $a$, the posterior distributions agree with the reference values.

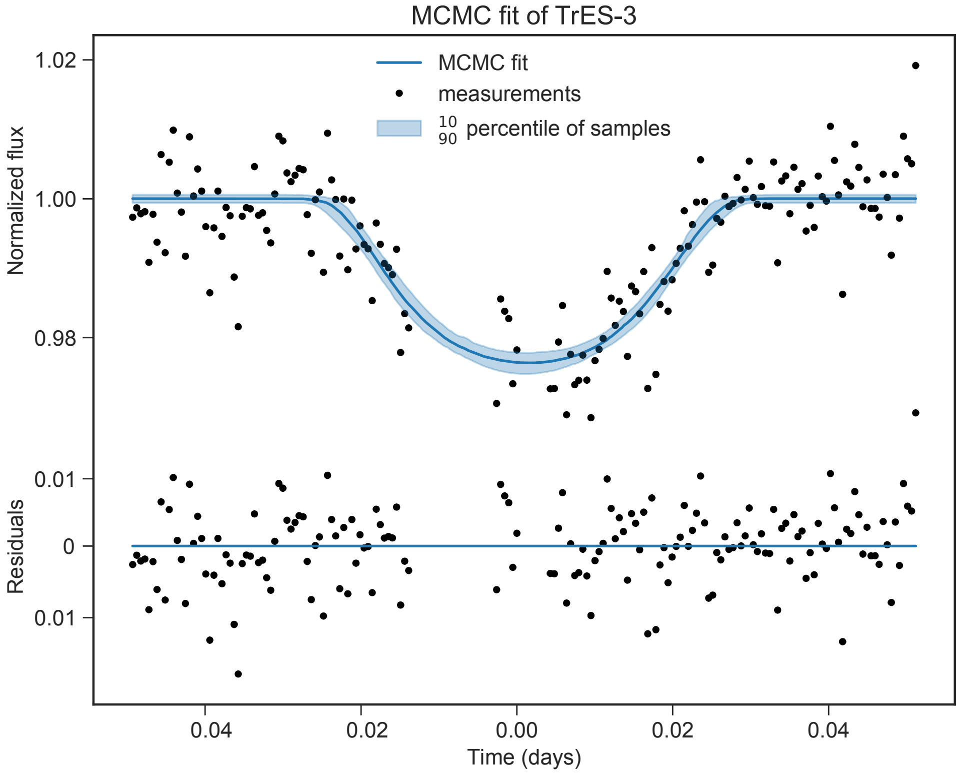

**Fig. 6:** The decorrelated TrES-3b light curve and the result of the MCMC sampling. The residual is shown below the fit, and the blue band reaches from the 10th to the 90th percentile of all MCMC samples.

### 5.2. HAT-P-44b

I observed a single HAT-P-44b transit in 2019. HAT-P-44 is a magnitude 13.2 star and hosts a confirmed exoplanet. The observation consists of 233 × 60 s frames from an optical CCD with no band-filter. The same data analysis was used as in the previous section. The detrending algorithm could not remove all systematic errors, and the MCMC sampler encountered numerical problems, including approximately 4000 divergences that were encountered during sampling. The results of the MCMC sampling are shown in fig. 7, and the decorrelated light curve with the fit is shown in fig. 8.

## 6. Discussion

**Decorrelation** The detrending routine from the Kepler pipeline can successfully remove most of the trend by fitting one cotrending basis vector to the light curve. The resulting light curve still shows a high amount of random noise, which can not be removed by detrending. This is probably due to atmospheric conditions at the observing site.

**Transit search** The algorithm can reliably find transit-like events in a light curve. However, it can not distinguish an exoplanet transit from other types of variable sources, such as eclipsing binary stars. The user has to manually select which star should be used for further analysis.

**Parameter fitting** The median samples from the MCMC sampling of the TrES-3b transit agree within 15 % with the reference values, except for the semi-major axis length and the period. From the correlation-plots, it can be seen that the semi-major axis length is steeply correlated with the period. Since only a single transit was observed, the uncertainty in the period is high. Due to the aforementioned correlation, the posterior distribution for the semi-major axis length is very unreliable.

To mitigate this effect, multiple observations of the same transit have to be conducted within a short time. Because of the variable weather conditions in Zürich, this is usually not possible.

## A. Diagnostics of MCMC-sampling

The reliability and success of an MCMC-sampling can be estimated with various parameters. Many of these parameters are calculated after the sampling with `PyMC3` and a warning is issued to the user if they indicate problems. However, it is advisable to manually check the trace-plots of the sampling to detect problems that might have not been found by the automatic checks. The trace-plots are automatically saved in this program.

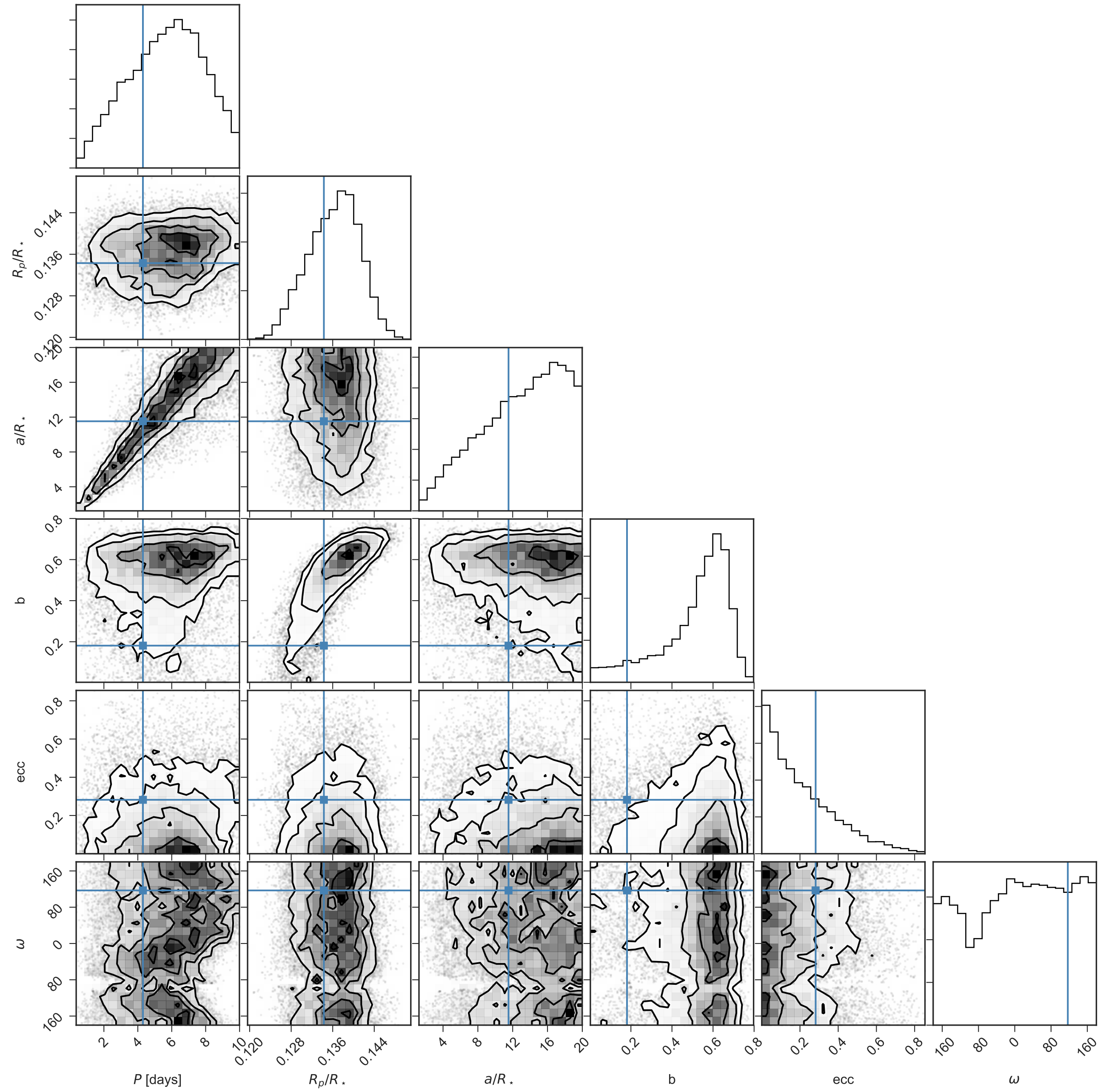

**Fig. 7:** Posterior distributions and correlation for the orbital parameters of HAT-P-44b. The reference values from the exoplanet orbit database are shown in blue. Due to numerical problems during sampling, the results are not reliable.

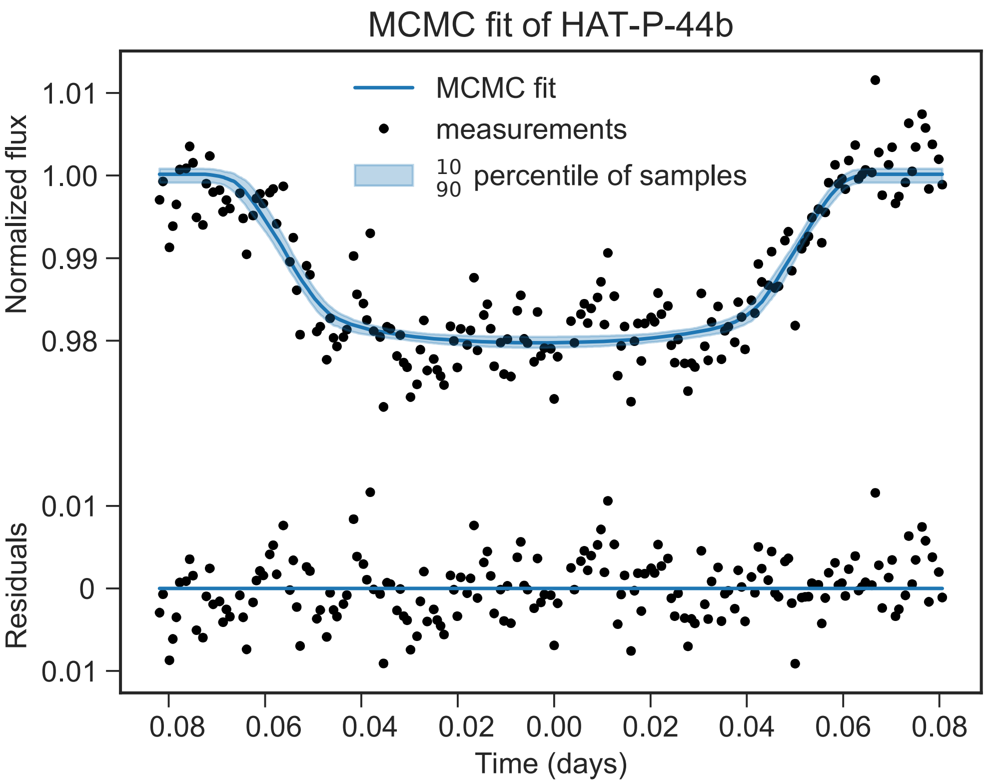

**Fig. 8:** The decorrelated HAT-P-44b light curve and the result of the MCMC sampling. The residual is shown below the fit, and the blue band reaches from the 10th to the 90th percentile of all MCMC samples.

As an example, the trace-plot of an MCMC-sampled TrES-3b transit is shown in fig. 9. Indicators of successful sampling are agreement between different chains, and a good exploration of the parameter space.

The agreement between chains can be estimated from the posteriors in the left column. All chains of the period $P$ and the semi-major axis length $a$ show approximately the same posteriors, indicating a reliable value. However, the orange chain of the impact parameter $b$ deviates clearly from the other chains, which indicates a problem during sampling. Since the three other chains all agree, this should not have much influence on the result. The chains for the argument of periastron $\omega$ all show very different posteriors. Therefore, the value for $\omega$ is very unreliable.

The exploration of the parameter space in the right column is harder to analyse. A good sampling will result in a trace that seems to fluctuate consistently around the median, while also briefly, regularly, and frequently exploring values that are far from the median. The desirable shape is commonly described as a “fuzzy caterpillar”. It is no problem if different chains have different traces in the right column, as long as their posteriors in the left column agree. All traces look good, except for the orange chain of the radius, which strays from the median for 1000 samples. Assuming the model is correct, the trace-plots can be improved by running the sampler for a longer time.

## B. Aperture size

The aperture size for photometry is automatically determined by `SExtractor`, but can be scaled by a factor by the user if desired. The photometry was tested with factors 1, 2.5 (default value), and 4. The decorrelated light curves are shown in fig. 10.

The light curve does not change significantly for a small radius, so the factor 2.5 can be lowered if needed. If a larger radius is chosen, the noise increases, and two nearby stars might influence each other’s photometry. Therefore, the default value of 2.5 is a good choice and was used for the data analysis.

## References

[1] Astropy Collaboration et al. “Astropy: A community Python package for astronomy”. In: *Astronomy and Astrophysics* 558 (Oct. 2013), A33. DOI: `10.1051/0004-6361/201322068`.

[2] Astropy Collaboration et al. “The Astropy Project: Building an Open-science Project and Status of the v2.0 Core Package”. In: *The Astronomical Journal* 156 (Sept. 2018), p. 123. DOI: `10.3847/1538-3881/aabc4f`.

[3] E. Bertin and S. Arnouts. “SExtractor: Software for source extraction.” In: *Astronomy and Astrophysics Supplement* 117 (June 1996), pp. 393–404. DOI: `10.1051/aas:1996164`.

[4] A. J. Bohn. “Detection of an Exoplanet Transit Using the 0.5 m Telescope on HPP”. In: *ETH Zürich semester thesis* (2016).

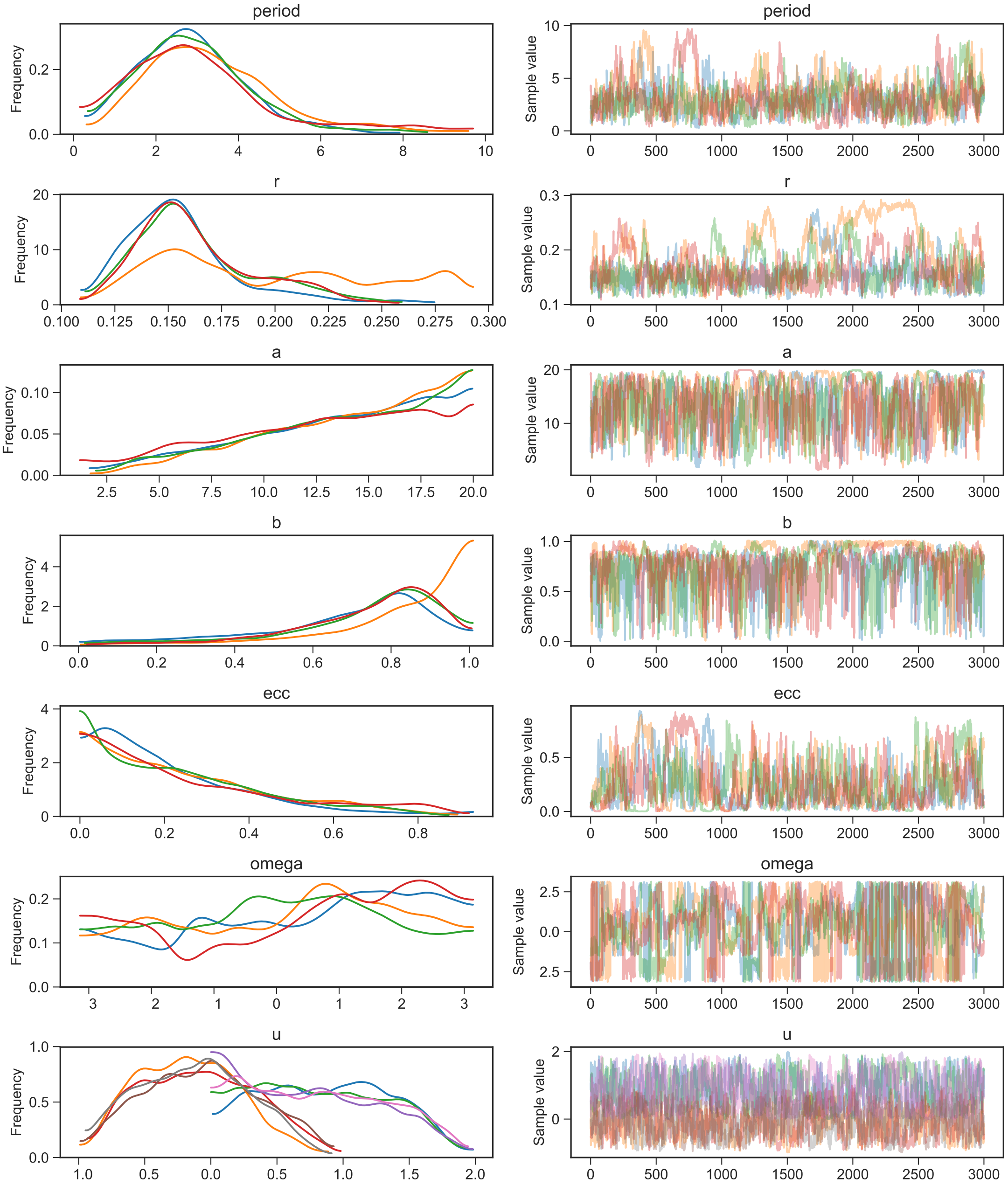

**Fig. 9:** The trace-plots of an MCMC-sampled exoplanet transit. The left column shows the posterior distributions of all four chains for all parameters. The right column shows the values that the different chains take over time.

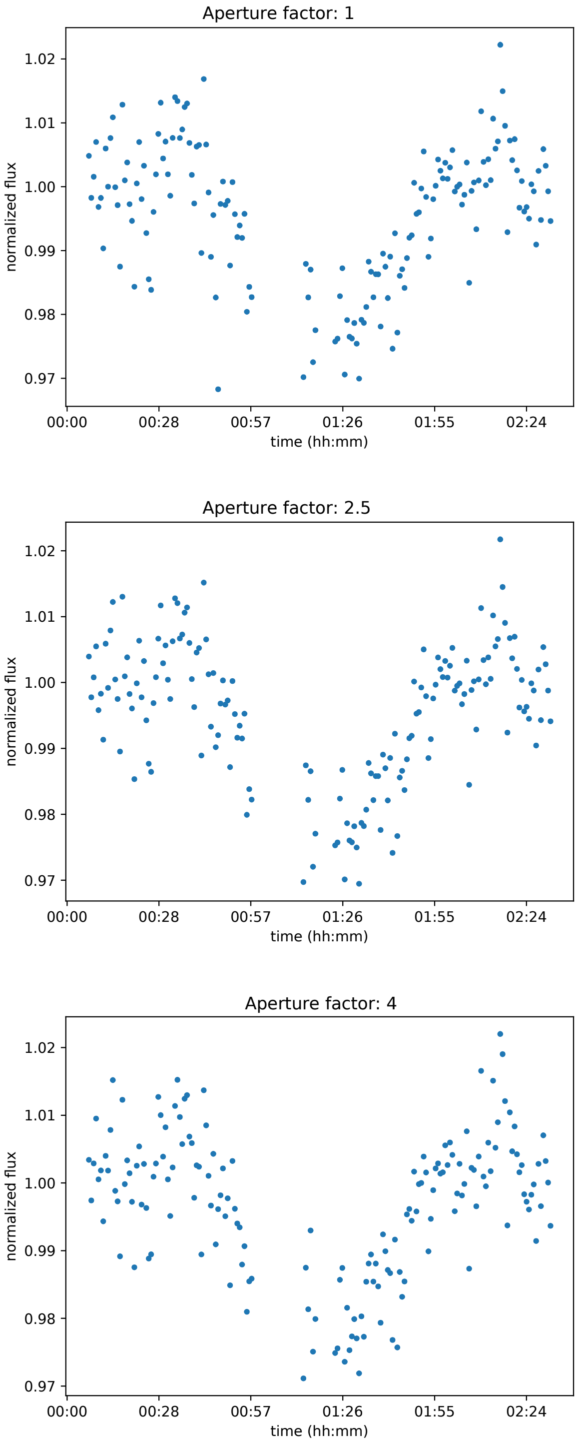

**Fig. 10:** Photometry with different aperture scaling-factors.

[5] A. C. Cameron. “Extrasolar Planetary Transits”. In: *Methods of Detecting Exoplanets: 1st Advanced School on Exoplanetary Science*. Ed. by V. Bozza, L. Mancini, and A. Sozzetti. Vol. 428. Astrophysics and Space Science Library. 2016, p. 89. DOI: `10.1007/978-3-319-27458-4˙2`.

[6] Dan Foreman-Mackey, Geert Barentsen, and Tom Barclay. *dfm/exoplanet: exoplanet v0.1.5*. Mar. 2019. DOI: `10.5281/zenodo.2587222`. URL: `https://doi.org/10.5281/zenodo.2587222`.

[7] A. A. Gheorghe. “Towards Measuring Exoplanetary Transits Using the HPP Telescope and Developing a Standalone Program for Data Analysis”. In: *ETH Zürich semester thesis* (2018).

[8] E. Han et al. “Exoplanet Orbit Database. II. Updates to Exoplanets.org”. In: *Publications of the Astronomical Society of the Pacific* 126 (Sept. 2014), p. 827. DOI: `10.1086/678447`. arXiv: `1409.7709 [astro-ph.EP]`.

[9] D. M. Kipping. “Efficient, uninformative sampling of limb darkening coefficients for two-parameter laws”. In: *Monthly Notices of the Royal Astronomical Society* 435 (Nov. 2013), pp. 2152–2160. DOI: `10.1093/mnras/stt1435`.

[10] Z. Kopal. “Detailed effects of limb darkening upon light and velocity curves of close binary systems”. In: *Harvard College Observatory Circular* 454 (1950), pp. 1–12.

[11] G. Kovács, S. Zucker, and T. Mazeh. “A box-fitting algorithm in the search for periodic transits”. In: *Astronomy and Astrophysics* 391 (Aug. 2002), pp. 369–377. DOI: `10.1051/0004-6361:20020802`. eprint: `astro-ph/0206099`.

[12] R. G. Kron. “Photometry of a complete sample of faint galaxies”. In: *The Astrophysical Journal Supplement Series* 43 (June 1980), pp. 305–325. DOI: `10.1086/190669`.

[13] D. Lang et al. “Astrometry.net: Blind astrometric calibration of arbitrary astronomical images”. In: *The Astronomical Journal* 137 (2010). arXiv:0910.2233, pp. 1782–2800.

[14] R. Luger et al. “starry: Analytic Occultation Light Curves”. In: *The Astronomical Journal* 157 (Feb. 2019), p. 64. DOI: `10.3847/1538-3881/aae8e5`.

[15] Tsevi Mazeh, Omer Tamuz, and Shay Zucker. “The Sys-Rem Detrending Algorithm: Implementation and Testing”. In: (Jan. 2007).

[16] Francis O’Donovan and David Charbonneau. “TrES-3: A Nearby, Massive, Transiting Hot Jupiter in a 31 Hour Orbit”. In: *The Astrophysical Journal Letters* 663 (Dec. 2008), p. L37. DOI: `10.1086/519793`.

[17] John Salvatier, Thomas V Wiecki, and Christopher Fonnesbeck. “Probabilistic programming in Python using PyMC3”. In: *PeerJ Computer Science* 2 (2016), e55.

[18] Jeffrey C. Smith et al. “KeplerPresearch Data Conditioning II - A Bayesian Approach to Systematic Error Correction”. In: *Publications of the Astronomical Society of the Pacific* 124.919 (Sept. 2012), pp. 1000–1014. DOI: `10.1086/667697`. URL: `https://doi.org/10.1086%2F667697`.

[19] Martin C. Stumpe et al. “Kepler Presearch Data Conditioning I—Architecture and Algorithms for Error Correction in Kepler Light Curves”. In: *Publications of the Astronomical Society of the Pacific* 124.919 (Sept. 2012), pp. 985–999. DOI: 10.1086/667698. URL: https://doi.org/10.1086%2F667698.

[20] Theano Development Team. “Theano: A Python framework for fast computation of mathematical expressions”. In: *arXiv e-prints* abs/1605.02688 (May 2016). URL: http://arxiv.org/abs/1605.02688.