Validation of exoplanets exhibiting dynamical interaction

Warsame Mohamed Ali

Division of Astrophysics
Department of Physics

LUND UNIVERSITY

Abstract

The Transiting Exoplanet Survey Satellite (TESS) uses transit photometry to detect exoplanets. To confirm if it is a planet, a second detection method is used to determine the planet's mass. But with the number of candidates waiting for confirmation, statistical tools have been developed to validate exoplanets so that false positives could be ruled out quickly. One such tool is the Tool for Rating Interesting Candidate Exoplanets and Reliability Analysis of Transits Originating from Proximate Stars (`TRICERATOPS`) [Giacalone et al., 2021].

`TRICERATOPS` works well for systems with strict periodicity but I investigated its performance in validate systems exhibiting dynamical interactions or Transit Timing Variations (TTVs). For a system to be deemed as validated, it needs a false positive probability (FPP) below 0.015. An FPP between 0.015 and 0.5 indicates a likely planet while an FPP above 0.5 indicates a false positive. By varying the TTV amplitude, TTV period, and impact parameter of the transit of a mock system, I tested how well `TRICERATOPS` worked.

I discovered that `TRICERATOPS` struggles to validate a planet with a relative TTV amplitude of 5.38% of the transit duration and it increases until a relative TTV amplitude of 17.9% where the FPP reaches 1. Varying the TTV period at an amplitude below this limit does not seem to have an effect on the FPP. There is no reliable interpretation of the FPP when above this limit. The impact parameter caused the FPP to increase at about $b = 0.5 - 0.6$ as it increased.

The conclusion of this test is that `TRICERATOPS` is not a viable tool for systems with TTVs unless the TTVs have been carefully corrected. Otherwise, the system may be classified as a false positive.

Popular science description

When people think about space they may think about something practical like the satellites that orbit the Earth and help us in our day-to-day lives. Or they think about something more fantastical like aliens on another world, like from Star Trek. This project leans toward the fantastical side of the spectrum but is not quite the same. It is about validating exoplanets around distant stars. These planets have been detected using the transit method. This means we measure the light coming from a star and when that light dims, it means an object has eclipsed it. If this happens at regular intervals, we know there is the possibility of there being an exoplanet and we can find its size from how much light is blocked. The problem is then, how do we know if it is a planet we are seeing and not maybe a binary system where two stars orbit each other? We cannot see the planet, so we use another detection method like radial velocity, which detects massive planets close to the star by seeing how much the gravitational forces from the planets affect the star. This is perfect because transits are easier to detect for massive planets close to the Sun! That means these planets are likely to be gas giants like Jupiter where a year on the planet takes on average 5 days. Hence, it is very unlikely there is any alien life on these planets.

Scientists discover many candidates, so they created tools to determine which of these candidates are the most interesting to focus on. One of these tools is the Tool for Rating Interesting Candidate Exoplanets and Reliability Analysis of Transits Originating from Proximate Stars or `TRICERATOPS` like the dinosaur. This tool analyzes data from a transit and calculates the probability of the object being a false positive. It allows scientists to determine what object to do follow-up studies on.

But what about the dynamical interaction? This could mean that the star has more than one planet orbiting around it, just like our own solar system. These planets will affect each other with their gravity, slowly varying their orbital period. They are no longer at regular intervals, which means that the length of a year on the planets is going to vary. Now the question this project seeks to answer is if `TRICERATOPS` can be used to accurately validate planets that have these irregular orbits. This is important to further help scientists not waste time doing follow-up observations on objects that cannot be exoplanets so that they can focus on discovering new worlds in this galaxy. And perhaps discover some aliens along the way.

Contents

Chapter 1

Introduction

The search for exoplanets has been fascinating, with new technologies and methods being developed to discover more and more about exoplanets and planetary systems. Simply put, exoplanets are planets that are outside our solar system, orbiting other stars. Just like the planets in our system, they come in different varieties from small rocky planets to large gas giants and even possible Earth-like planets. A definition of an exoplanet is given by the *International Astronomical Union's (IAU) Commission F2: Exoplanets and the Solar System* in 2018. It defines the upper mass limit of exoplanets as the mass limit for deuterium burning at $13M_J$[1] and the stability of the Lagrange points L4 and L5. Lagrange points are positions where the gravitational influence from two more massive objects on a smaller mass object, is in equilibrium. The L4 and L5 points are stable Lagrange points. For these points to be stable, the mass of the orbiting object needs to be less than 1/25th of the central object [Cornish, 1998]. Lastly, the lower mass limit of planets is sufficient mass to assume hydrostatic equilibrium and a round shape, like Ceres. The definition has been updated from its original 2003 version to include new insights gained over the years, and it will continue to be revised in the future.[Lecavelier des Etangs and Lissauer, 2022].

As of 2nd of April 2024, 5602 planets have been confirmed[2], the first detected in 1988 and confirmed in 2003 was γ Cephei Ab using the radial velocity method [Hatzes et al., 2003]. This is a method where the wobbles of the star as it is being slightly moved by the orbiting planet can be detected through spectroscopy. The spectrum of the star shifts as it moves away or towards the observer. This is called redshift and blueshift respectively. From these movements, the mass of the planet causing the wobbles can be determined. The first exoplanet to be found and confirmed around a main-sequence star was around 51 Pegasi in 1995, called 51 Pegasi b, using the radial velocity method [Mayor and Queloz, 1995]. 51 Pegasi b is what is known as a Hot Jupiter which are massive gas giants that orbit very close to their host star. Due to the close proximity to the star, the atmosphere of the gas giant gets heated, leading to its name Hot Jupiter. Gas giants are the most massive planets

---

[1]This is only for objects with solar metallicities. This mass limit will be different depending on the metallicity of the object.

[2]https://exoplanets.nasa.gov/ Accessed: 2 April 2024

so the upper mass limit of a Hot Jupiter is around 13 Jupiter masses [Lecavelier des Etangs and Lissauer, 2022]. The lower mass limit is generally seen as 0.1 M_J[Wang et al., 2015]. Hot Jupiters have short orbital periods, generally seen as 10 days or less[Wang et al., 2015].

The transit photometry and radial velocity methods are ideal for discovering Hot Jupiters because of their size and proximity to their star. Transit photometry is observing a star in broadband photometry at high cadences to detect a decrease in brightness coming from the star as a planet transits, blocking the light from reaching us. It is the most successful and most frequently used method to detect exoplanets, having detected 74.4% of the exoplanets discovered according to NASA as of 2nd of April 2024 [3]. The planet HD 209458 b was first discovered using the radial velocity method but was the first planet to be detected using transit photometry. This could prove that this was a planet and not a binary or stellar variability by calculating the density of the planet from the mass and radius [Charbonneau et al., 2000]. This is part of the confirmation of planets where both methods are used to get the mass and radius [Landau, 2018]. The order in which they are discovered does not matter. This is why Hot Jupiters are perfect for the radial velocity and transit methods due to their large mass and radius. However, the transit made by a Hot Jupiter could be confused with a transit coming from an eclipsing binary in the background behind a brighter star. There are also a lot of candidates already detected but have not been confirmed[4]. This has shown the need for validation of the candidates.

Validation is the process of ruling out the candidates that are not exoplanets [Giacalone et al., 2021]. These unwanted candidates are known as false positives and include scenarios where eclipsing binaries or instrumental errors cause the signal. Thousands of possible exoplanets have been detected[5], far too many to do follow-up investigations on all of them. There are ways to validate exoplanets like multiband photometry where the light intensity from an object is measured at multiple wavelengths to find different transit depths at different bands [Giacalone et al., 2021]. A faster way to distinguish between false positives and actual exoplanets is using validation tools. Validation tools are statistical tools used for the validation process by analyzing the light curve data from telescopes, statistically determining the probability of there being a planet there. The tool I will use is the Tool for Rating Interesting Candidate Exoplanets and Reliability Analysis of Transits Originating from Proximate Stars (`TRICERATOPS`) [Giacalone et al., 2021]. It determines the likelihood of the signal coming from a planet or not by assuming a linear periodicity in the transit, meaning that the transit happens at a fixed time interval. With strict periodicity and Bayesian statistics, `TRICERATOPS` can accurately determine the likelihood of the signal being from a planet or not and if there are unresolved stars blended in with the light we see. Now we are presented with a new challenge, however. Can `TRICERATOPS` determine if the transit signal is an exoplanet or a false positive if we have a dynamical interaction

---

[3] https://exoplanets.nasa.gov/ Accessed: 2 April 2024

[4] https://exoplanetarchive.ipac.caltech.edu/docs/counts_detail.html Accessed: 2 April 2024

[5] https://exoplanets.nasa.gov/ Accessed: 2 April 2024

between multiple bodies orbiting the same star? This dynamical interaction is the gravitational forces of the bodies affecting each other, causing a variation in the orbital period and breaking the periodicity. The variations in the periods are called Transit Timing Variations (TTVs) [Agol et al., 2005] and that is what I aim to answer in this thesis by creating my own mock TTV transit data and see if TRICERATOPS can validate the candidate.

In Chapter 1, the necessary background information, theory, and idea behind validation and TRICERATOPS will be provided. In Chapter 2, I will explain the methodologies used in the thesis. In Chapter 3, I will present and analyze the results to evaluate the impact of TTVs on TRICERATOPS. In Chapter 4, I will end with a summary and a conlcusion.

1.1 Transit Photometry

1.1.1 How transit photometry works

The way the transit method works is that telescopes equipped with photometers measure the brightness of stars. When an object passes by between the star and the telescope, a portion of the light will be blocked which is detected by the photometer. The flux at that moment in time will be lowered and is called a primary transit. When the object orbits behind the star, any light it would reflect or emit is now blocked from reaching the telescope which can be seen on the light curve as a smaller transit curve. This is called a secondary eclipse [Deeg and Alonso, 2018]. The object orbiting the star has to orbit right between the observer and the star itself to be detected. This is denoted by the impact parameter, b, with $b = 0$ for a central transit, and $b = 1$, for when the object grazes the top of the star. The impact parameter is related to the inclination i by

$$b \equiv \frac{a}{R_s} \cos i, \tag{1.1}$$

with the semi-major axis a and the radius of the star R_s[Deeg and Alonso, 2018] and can be seen in Fig.1.1. The inclination is the measure of the tilt of the planet's orbit around the star. In addition, the impact parameter will also affect the look of the transit light curve. A central transit will have a steep ingress and egress with a slightly rounded bottom to the curve due to limb darkening as the intensity of light is weaker at the limbs of the star, while a grazing transit will have a more gradual ingress and egress leading to a shallower and pointier curve due to the limbs darkening and a shorter transit time as seen in Fig. 1.2.

Models are often used to simulate light curves and one such model that I will use is the quadratic transit model created by [Mandel and Agol, 2002] included in the PyTransit package[Parviainen, 2015]. This model describes the limb darkening as a quadratic function and it is a good approximation. It is described as

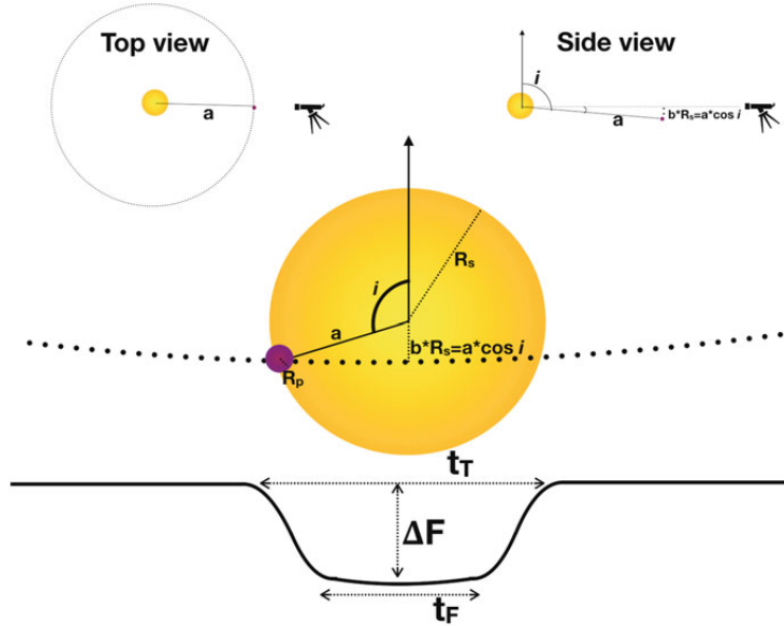$$I(r) = 1 - \gamma_1(1 - \mu) - \gamma_2(1 - \mu)^2, \tag{1.2}$$

Figure 1.1: Figure showing the transit with the total transit duration t_T, the full transit duration t_F and depth of the transit ΔF. A side view shows the inclination i of the orbit and a top view shows the semi-major axis a of the orbit[Deeg and Alonso, 2018].

where γ_1 and γ_2 are the limb darkening coefficients with $\gamma_1 + \gamma_2 < 1$. These coefficients determine how much the limb darkens where $I(r)$ is the normalized specific intensity where $I(0) = 1$. r is the normalized radial distance from the center, $r = 0$ is the center of the star, $r = 1$ is at the limb of the star, μ is the cosine angle between the line of sight and the normal to the stellar surface, $\mu = \cos\theta$. At the center of the star $\mu = 1$ and at the limb $\mu = 0$ where μ can be written in terms of r as $\mu = \sqrt{1 - r^2}$ [Mandel and Agol, 2002]. The width of the transit curve is the duration of the transit. When the planetary disk first touches the star, it is called first contact. When the planetary disk has fully gone over the edge is called the second contact. Similarly, there will be third and fourth contacts as the planet passes the star. As seen in Fig. 1.1, the total duration of the transit, t_T, is from the first to the fourth contact while the time of totality, t_F, is between the second and third contacts [Deeg and Alonso, 2018].

From the depth of the light curve ΔF seen in Fig.1.1, we can determine the size of the object orbiting the target star because the change in flux is related to the sizes of the star and the planet. Assuming the flux coming from the object itself is negligible and a spherical shape for both star and planet, we can see that the change in flux is given by,

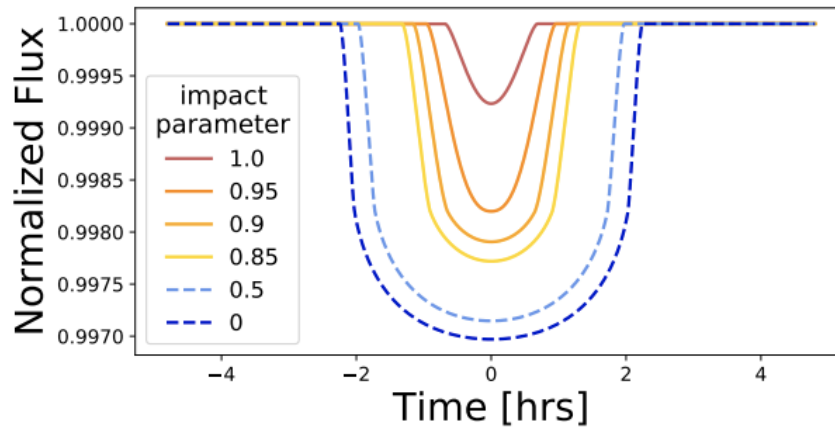$$\Delta F \approx \left(\frac{R_p}{R_s}\right)^2 = k^2, \tag{1.3}$$

Figure 1.2: Figure showing the effect on the light curve caused by the differing impact parameters of a warm Neptune around a Sun-like star. The solid lines are grazing or near grazing, while the dashed lines are full transits. [Gilbert, 2022].

where k is the ratio between the radius of the planet R_p and the star R_s. It is only exactly true for a uniform stellar disk with no limb darkening. This equation is used to determine the radius of the planet. With a follow-up investigation using the radial velocity method, the mass can be determined as well, which along with the radius can tell us more about the planet and its composition. A planet is deemed confirmed if both of these parameters have been measured and thus, its density can be measured [Deeg and Alonso, 2018].

The time between transits is the orbital period of the planet. Two or more transits are required to determine the orbital period. That is why it is easier to use the transit method for planets with short orbital periods in the range of a few days where the object can be seen regularly like Hot Jupiters for example. A smaller orbital period results in a larger transit probability for the object.

Variations in the orbital period or TTVs occur and can be caused by gravitational forces from other bodies orbiting the same star. The planets in these systems accelerate and decelerate each other, breaking up the strict periodicity a lone planet would have. The variations can be likened to an oscillator with a TTV amplitude, TTV period, and a TTV phase. The orbital period of the planet will oscillate around the mean period so we can approximate this using a sinusoidal wave. The y-axis will be the orbital period of the planet in question and the x-axis is time. The amplitude of the sinusoidal wave is the maximum deviation of the orbital period or TTV amplitude, while the period of the sinusoidal wave shows how frequently these deviations in the orbital period occur. This is shown in Fig. 1.3.

TTVs can be used as a detection method. Since TTVs come from planets accelerating
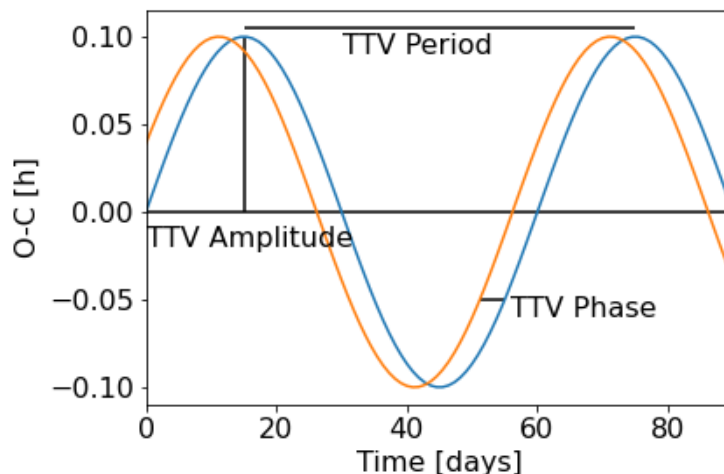
Figure 1.3: Figure showing what the TTV amplitude, TTV period, and TTV phase are.

and decelerating each other, these perturbations can be used to gain knowledge about the perturber. As an example of a similar situation, we have Neptune. Before Neptune was discovered, astronomers noticed that Uranus sped up and slowed down. Leverrier used these perturbations to calculate the position of Neptune. So from TTVs, we can infer that there could be one or more planets perturbing the planet we are seeing. It is also possible to discover an unseen planet using this method, like Kepler-46c. Additionally, more can be inferred about the eccentricity of the orbits and the resonances of the planets [Agol and Fabrycky, 2018].

1.1.2 Transiting Exoplanet Survey Satellite

The TESS mission, led by the Massachusetts Institute of Technology (MIT), is an all-sky surveying mission with the main objective of finding exoplanets smaller than Neptune that can be followed up with spectroscopic analysis to determine the mass and atmospheric compositions. The sky is divided into sectors for TESS to observe. TESS focuses on main-sequence dwarfs with spectral types F5-M5 which are not too bright as to obscure any potential exoplanets due to the small transit depths the planets will make. M-type is of special interest because these stars are abundant and their small size makes a transit deeper. TESS also focuses on planets with a period of less than 10 days, because of the preference of short period targets for transit photometry. However, it searches for planets with periods above 40 days at the ecliptic poles.

TESS uses four identical cameras with $f/1.4$ lenses giving a 24x24 degree field of view each with an effective aperture of 10 cm with a 2-minute cadence for a selection of stars with full-frame images returned every 30 minutes. TESS uses silicon charge-coupled devices (CCDs) sensors to measure the brightness of a star with a bandpass of 600 to 1000

nm. Sensitivity to red light is preferred since planets are easier to detect around smaller, cooler, and redder stars. The orbit of TESS maximizes the sky coverage and is a stable platform for the observations. The orbit of TESS is elliptical with the closest point to Earth being 17 Earth radii away, so that it can transfer the data back to Earth[6][Ricker et al., 2015].

1.2 False positives

Planets are not the only objects to orbit stars. The false positives can come from something like an eclipsing binary which can look similar to a signal from a planet under the right circumstances [Deeg and Alonso, 2018]. Normally an eclipsing binary will have a deeper transit curve than a transiting planet. But there are cases where the transit curves can be mistaken for each other like when the transiting object is a brown dwarf or a star with the radius of a large gas giant. Another potential origin for a false positive signal could be when an eclipsing binary is near a brighter star or is in the background of a star we are measuring. This could cause the transit curve of an eclipsing binary to be shallower like a transiting planet curve, because the brighter star adds to the total flux. Another way to get a false positive is when we have a grazing binary, which is a transit with a high impact parameter. Just like for the transit of planets, the transit curve for an eclipsing binary will be shallower at a high impact parameter due to the limb darkening of the star. The curve can then be shallow enough to be mistaken for a transiting planet.

There are ways to confirm if the signal is coming from a planet or an eclipsing binary, however. Taking the grazing binary as an example, their transit light curves will have a distinct V-shape to them compared to the U-shape from transiting planets transiting near the center of the star as seen in Fig. 1.2. Conversely, a transiting planet with a high impact parameter will also have this V-shape and could mimic an eclipsing binary with a shallower transit curve like a blended eclipsing binary. Additionally, if the mass of the star is known, the density of the star can be determined. If it is inconsistent with the expected density of a star with that temperature, it could be a false positive. Another method to detect false positives is follow-up investigations with a different detection method like the radial velocity method to determine the mass of the orbiting object and therefore its density to determine what the object is but this is costly and time-consuming [Cameron, 2012]. A possible way to distinguish binaries from transiting planets could be the secondary eclipse. When both stars of a binary have similar surface brightness, the primary and secondary eclipses will have similar depths. This could be confused for a transiting planet that has half the orbital period as the binaries. When the surface brightness of the stars is different, the primary and secondary transits will have visibly different depths. Those can be classified as a false positive eclipsing binary [Sullivan et al., 2015]. There are also statistical validation tools like `TRICERATOPS` that can be utilized to quickly determine if a planetary

---

[6]`https://tess.mit.edu/science/` Accessed: 26 March 2024

candidate is a planet or an eclipse. These statistical tools are faster than multicolor photometry or reconnaissance spectroscopy which can be used to identify false positives.

Eclipsing binaries are not the only type of false positives. Other possible sources for false positives include artifacts from the instruments and starspots traveling across the surface of the star. Validation or confirmation using a second detection method is important to confirm the object and ideally, it would be possible to follow up on every single planet candidate. But the amount of candidates is just too large to follow up on all of them. So tools had to be created to decide which ones to follow up on.

1.3 Validation

1.3.1 Validation tools

Exoplanet validation using statistical tools has become an important part of the exoplanet search. With the large amounts of candidates found by Kepler, TESS, and other telescopes, it became increasingly difficult, expensive, and time-consuming to do follow-up investigations to confirm all of them. Candidates are now validated by algorithms to statistically determine the probability of the exoplanet being a false positive just from the light curve and the known data. Some of these validation tools are `BLENDER`, `VESPA` and `PASTIS`[Torres et al., 2004, 2005, Morton, 2012, Díaz et al., 2014, Santerne et al., 2015, Giacalone et al., 2021]. With these tools, scientists could now prioritize what target stars were likely to have an exoplanet. However, these tools are not suited for use with TESS data. TESS has a wide view over the sky. This means that TESS has more contamination than the Kepler telescope and similar telescopes like the Convection, Rotation and planetary Transits telescope (CoRoT). A new tool was therefore created, made for TESS called `TRICERATOPS`[Giacalone et al., 2021].

1.3.2 TRICERATOPS

`TRICERATOPS` is a validation tool utilizing Bayesian statistics to determine if the signal is an exoplanet or a false positive. It uses the primary transit along with the known data of the target and nearby stars. `TRICERATOPS` will display the TESS Object of Interest (TOI), which is the target star, and any star within a 10-pixel radius where each TESS pixel is 21". Light propagates in all directions so the light from all of these stars is mixed making them all potential origins. `TRICERATOPS` needs to determine the flux from each star to identify which can be the source of the signal. The size of the transiting object will be different for each star because each star has a different flux. The size is therefore needed to determine the probability of the object being a planet and is calculated from the transit depth.

Next is determining what kind of scenarios can produce the given signal. There are 18

scenarios in total which `TRICERATOPS` calculates the relative probabilities for, with 3 of them being scenarios where the transit signal is coming from a nearby star instead of the target star. That is determined by the transit depth for the star being $\Delta F_s < 1$. Table 1.1

Table 1.1: The 18 scenarios `TRICERATOPS` tests for [Giacalone et al., 2021].

Scenarios	Description of the scenarios
TP, EB, EBx2P	No unresolved companion
PTP, PEB, PEBx2P STP, SEB, SEBx2P	Unresolved bound companion transiting primary or secondary star
DTP, DEB, DEBx2P BTP, BEB, BEBx2P	Unresolved background star transiting target or background star
NTP, NEB, NEBx2P	No unresolved companion nearby star is the host

shows there are three main scenarios `TRICERATOPS` considers. A transiting planet (TP), an eclipsing binary (EB), or an eclipsing binary with double the period (EBx2P). The transiting planet or eclipsing binary could be transiting the target star with no unresolved companions (Fig. 1.4 **(a)**), the primary or secondary star of an unresolved bound companion (Fig. 1.4 **(b)**), or an unresolved background star (Fig. 1.4 **(c)**), or a nearby star could be hosting the transiting object with no unresolved companion (Fig. 1.4 **(d)**) [Giacalone et al., 2021].
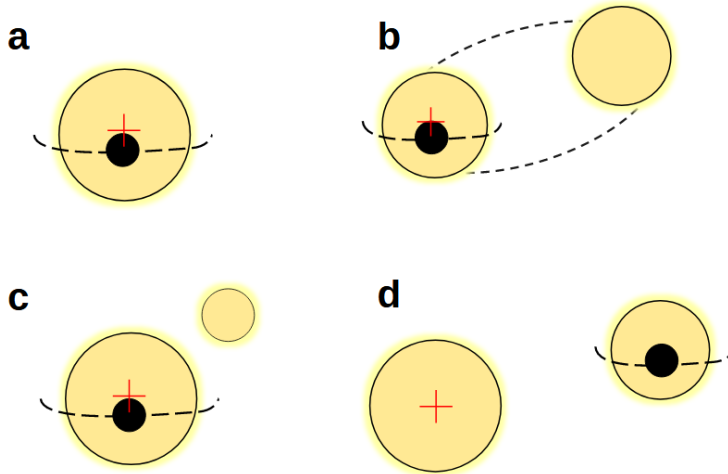

Figure 1.4: Figure showing what the different scenarios from Table 1.1 look like. **(a)** is a transit with no unresolved companions, **(b)** is a transit with an unresolved bound companion where the planet can orbit either star, **(c)** is a transit with an unresolved background star where the planet can orbit either star and **(d)** is the transit with no unresolved companions but the planet orbits a nearby star instead of the target star. The target star is signified by the red cross. Image created by Amelie Bormann.

Chapter 2

Method

2.1 Description of the problem/goal

`TRICERATOPS` as a validation tool does well for transit signals exhibiting a strict periodicity on the orbital period. This is not always the case. There are many systems with multiple planets causing the periods to vary. The change in the periodicity may make it difficult to determine if the transit signal we are seeing is an exoplanet or not. The signal we get will be smeared when phase folding light curves with TTVs. Phase folding is when the transit light curves are layered on each other. This is depicted in Fig. 2.1. This is due to the period changing making it difficult to gain information from the light curve. To get a good signal, the light curves need to be lined up by taking into account the period variations caused by the TTV. This TTV could hypothetically fool `TRICERATOPS` and cause it to think that it is seeing a false positive. This project aims to determine to what extent `TRICERATOPS` can validate exoplanet candidates that exhibit dynamical interaction from other nearby planets. How large can the TTVs be before `TRICERATOPS` misclassifies a transiting planet as a false positive?

2.2 TRICERATOPS probability calculations

The way `TRICERATOPS` calculates the probability of each scenario is through Bayes' theorem,

$$p(S_j|D) \propto p(S_j)p(D|S_j), \tag{2.1}$$

where $p(S_j|D)$ is the posterior probability of the scenario S_j happening given the data D, $p(S_j)$ is the prior probability of scenario S_j and $p(D|S_j)$ is the marginal likelihood of getting the data D if scenario S_j occurs. The data input into `TRICERATOPS` is the light curve, the transit parameters, stellar parameters and populations. After $p(S_j|D)$ has been
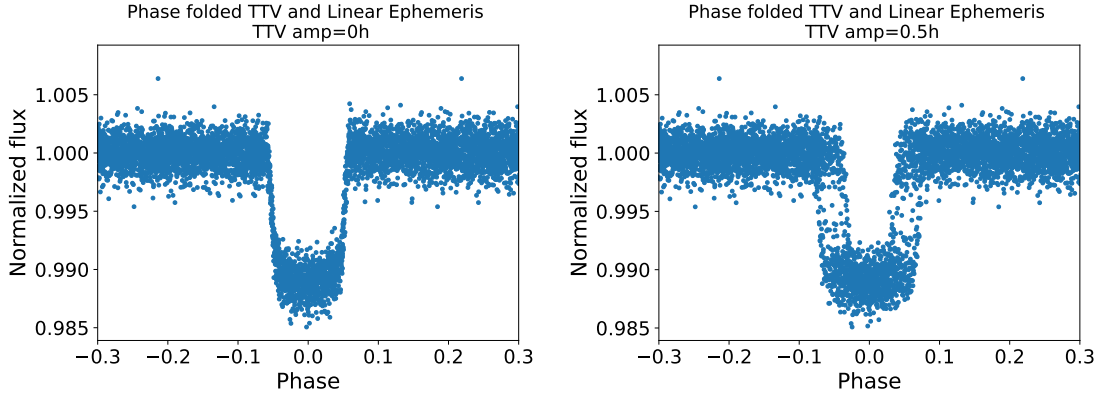
Figure 2.1: Figure showing the difference when phase folding with a TTV of 0.5 hours and without a TTV. The TTV phase fold is more smeared.

calculated for each scenario, the relative probability of each scenario can be calculated.

$$\mathcal{P}_j = \frac{p(S_j|D)}{\sum_j p(S_j|D)}. \tag{2.2}$$

Then we can determine the False Positive Probability (FPP) by subtracting the probabilities of the transits around the target star from 1,

$$FPP = 1 - (\mathcal{P}_{TP} + \mathcal{P}_{PTP} + \mathcal{P}_{DTP}). \tag{2.3}$$

`TRICERATOPS` also calculates the Nearby False Positive Probability (NFPP) for the three scenarios where the transit is coming from a neighboring star[Giacalone et al., 2021].

$$NFPP = \sum(\mathcal{P}_{NTP} + \mathcal{P}_{NEB} + \mathcal{P}_{NEBx2P}). \tag{2.4}$$

2.2.1 Scenario priors

Priors are the previously known knowledge and assumptions. In `TRICERATOPS`, they come in the form of the scenario priors which will give an initial probability for each scenario before data is taken into consideration. An example of a prior that `TRICERATOPS` uses comes from the frequency of transiting planets and eclipsing binaries at different periods essentially determining which of the two scenarios is the most likely at this period. `TRICERATOPS` accomplishes this by assuming the probability distribution as a broken power law with the range 0.1 days to 50 days. For transiting planets, this break occurs at $P_{orb} = 10$ days, and for eclipsing binaries, this break occurs at $P_{orb} = 0.3$ days. This can be seen in Fig. 2.2. `TRICERATOPS` will take these probability densities into account when calculating the probabilities of the scenarios. There are other priors which `TRICERATOPS` takes into account like the radius of the transiting object and the mass ratio between the eclipsing binaries. More detail about the other scenario priors are in Appendix A.
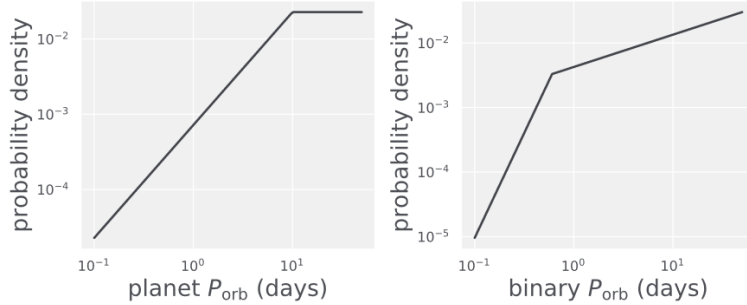
Figure 2.2: Figure showing the priors that `TRICERATOPS` uses. It depicts the probability density of planets and binaries for a given orbital period [Giacalone et al., 2021].

2.2.2   Marginal likelihood

Each scenario `TRICERATOPS` calculates the probability for, has an associated parameter vector θ_j, which needs to be included in the calculations for the marginal likelihood $p(D|S_j)$. The parameters are the inclination i, the radius of the planet R_p, the mass ratio between

Table 2.1: Table showing the parameter vector for each scenario.

Scenario	Parameter vector θ_j
TP, NTP	i, R_p
EB, EBx2P, NEB, NEBx2P	i, q_{short}
PTP, STP	i, R_p, q_{long}
PEB, PEBx2P, SEB, SEBx2P	i, q_{short}, q_{long}
DTP, BTP	i, R_p, simulated star
DEB, DEBx2P, BEB, BEBx2P	i, q_{short}, simulated star

the host star and short-period stellar companion q_{short} like a binary, the mass ratio between the host star and long-period stellar companion q_{long} like an unresolved bound companion and "simulated star" generated by TRILEGAL. TRILEGAL is a galactic model created to simulate different types of stars. In this case, TRILEGAL simulates stars with properties consistent with the target star in a cone around the line of sight of the target. The marginal likelihood is therefore expressed as

$$p(D|S_j) = \int p(\theta_j|S_j)p(D|\theta_j, S_j)d\theta, \qquad (2.5)$$

where $p(\theta_j|S_j)$ is the prior distribution of the model parameters and $p(D|\theta_j, S_j)$ is the marginalization of the likelihood all integrated over the parameter vector θ_j. This integral can not always be solved analytically so it is determined using Monte Carlo sampling. Further details on the marginal likelihood can be found in Appendix A. `TRICERATOPS` creates fits for each scenario to check it against the light curve it has been given. These fits are made using `PyTransit` [Parviainen, 2015] where the best fitting scenario is the
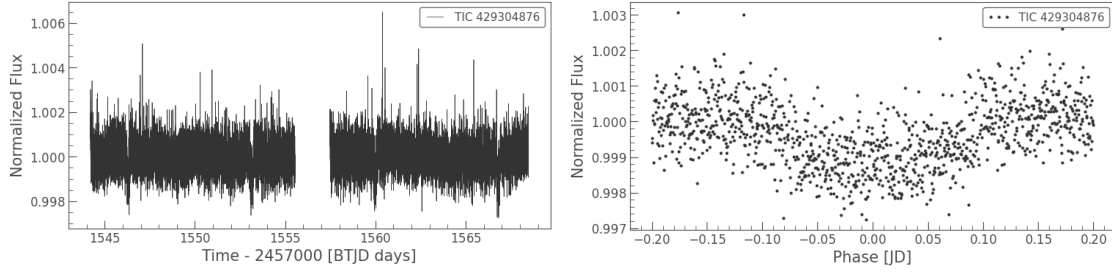
Figure 2.3: On the left, the full normalized light curve for the transit detected around TOI-682 at TESS sector 9. Transits are visible at times 1546, 1553, 1560 and 1562. On the right, the phase folded light curve for the transit detected around TOI-682. The plots are created using lightkurve [Lightkurve Collaboration et al., 2018].

most likely scenario. The scenarios use the quadratic limb darkening coefficients based on effective temperature T_{eff} and $\log(g)$ of the star for the simulated limb darkening.

2.3 Description of methodology

2.3.1 TRICERATOPS used on a non-TTV system

I first decided to use `TRICERATOPS` on a non-TTV system to test the tool and see if it worked properly. I will use this example to demonstrate how to use `TRICERATOPS`. I selected star TOI-682, which is a G-type star [Quinn et al., 2021]. It has the TESS Input Catalogue number, TIC 429304876, and it was observed in sector 9 as the target. I downloaded the light curve with 120 seconds exposure time from the "Barbara A. Mikulski Archive for Space Telescopes" database (MAST)[1] through the `lightkurve` Python package for Kepler and TESS data analysis [Lightkurve Collaboration et al., 2018]. I normalized the light curve and phase-folded it. To phase fold, I needed the epoch time of the center of the transit along with the orbital period. The epoch time can be seen in the plot on the left in Fig. 2.3 and the orbital period is found on the Exoplanet Follow-up Up Observing Program (ExoFOP) website[2]. Using `lightkurve`, I could then fold the light curve and truncate it to reduce the amount of unnecessary data from the baseline. Figure 2.3 shows the normalized and the phase folded light curves.

I defined the star and sector for `TRICERATOPS`. It generates a plot with the target star and all stars within a 10-pixel radius, where each pixel is 21" shown on the left side of Fig. 2.4. All of the stars and any possible unresolved companions could be the origins of the transit signal. Without any high-resolution data `TRICERATOPS` assumes no unresolved stars beyond a 2.2" distance. The plot shows the TESS magnitudes of all of the stars.

---

[1]`https://mast.stsci.edu/portal/Mashup/Clients/Mast/Portal.html` Accessed: 23 April 2024
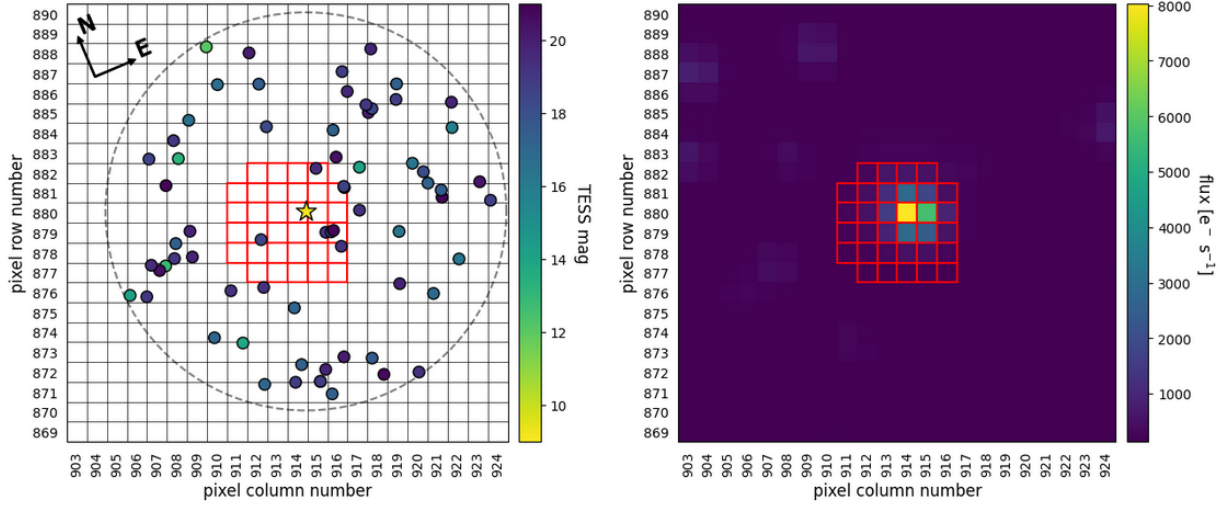[2]`https://exofop.ipac.caltech.edu/tess/` Accessed: 23 April 2024

Figure 2.4: Figure to the left: Plot of the target star, TOI-682, showing all the stars within a 10-pixel radius and their magnitudes. The pixels with red borders are the defined aperture pixels, taken from the figure created by SAOImageDS9. Figure to the right: Plot showing the flux contributed by each pixel. Aperture pixels are shown with red borders in this plot. The plots were generated by `TRICERATOPS` [Giacalone et al., 2021, Giacalone and Dressing, 2020].

Another plot is also shown on the right side of Fig. 2.4 with the flux on each pixel with a color plot. `TRICERATOPS` generates a table with information about all the stars within the circle like TIC ID, stellar parameters, and coordinates. In these plots, `TRICERATOPS` uses an aperture to extract the light curve. I used an imaging and visualization software called SAOImageDS9 to find the shape of the aperture. I entered these pixels into `TRICERATOPS` and they are highlighted with the red borders in Fig. 2.4.

`TRICERATOPS` needs the transit depth for the object to determine which of the stars can be candidates for hosting the transit signal by `TRICERATOPS`. I used the database on ExoFOP to find the transit depth given in parts per million. For TOI-682, the transit depth is 0.001213. After inputting the transit depth, `TRICERATOPS` generates the same table with the information about the stars from before, but now includes the flux ratio of all stars and the possible transit depth if the star was the host star of the transit signal, shown in Fig. B.1. It does this by first calculating the flux ratio X_s for all stars. The flux ratio is calculated by assuming the Point Spread Function (PSF) of each star is a circular 2D Gaussian peak with the area under the peak as the total flux of a star which is determined by the TESS magnitude of the star. The standard deviation is determined using the TESS Pixel Response Function (PRF) which depends on the location of the CCD the star was observed on. `TRICERATOPS` fits it to a 2D Gaussian gives a standard deviation between 0.6 and 0.9 pixels so a deviation of 0.75 pixels was chosen for all stars to make it simple. Most of the light will hit 75% of the pixel. The flux from a star is integrated and divided

over the total flux through the aperture, giving the flux ratio for the star X_s. The possible transit depth for all stars can be determined by

$$\Delta F_s = \frac{\Delta F_{obs}}{X_s},$$
(2.6)

where ΔF_s is the transit depth of the signal around the specified star and ΔF_{obs} is the observed transit depth. Any star that is too faint, where $\Delta F_s > 1$, can then be ignored[Giacalone et al., 2021].

`TRICERATOPS` now only needs the light curve data and the orbital period before it can start calculating the probabilities. I read the phase-folded light curve data I saved earlier and input the orbital period which could be found on ExoFOP. The period for TOI-682 is 6.8... days. I bin the light curve data down to 100 bins to reduce the time it takes for `TRICERATOPS` to complete the calculations. `TRICERATOPS` then calculates the probability for each scenario and displays the results. It shows the FPP, NFPP, and a table of all the scenarios and their probabilities along with other information for each scenario, Fig. 2.5. The information includes the stellar parameters like the mass and radius and fitting parameters like the impact parameter. It also shows the transiting object's estimated radius and the eclipsing binary's mass. The results will be slightly different every time the calculations are made unless the seed is defined and not randomized each time the code is run. For this reason, I ran it 10 times and took an average probability. After this, the fit for each scenario could be overlaid over the transit plot to visually see which scenario fits the best[3][Giacalone et al., 2021, Giacalone and Dressing, 2020]. `TRICERATOPS` recommends running it 20 times but I only did 10 because of a lack of time. For light curves with TTVs, `TRICERATOPS` is used similarly but with some alterations. These alterations will be explained in Section 2.3.3.

Running `TRICERATOPS` on the transit detected around TOI-682 gave $FPP = 0.0095 \pm 0.0061$. This means the transit signal can be considered a validated planet. Figure 2.5 shows that the most likely scenario for the transit is a transiting planet with no unresolved companions at $\sim 76\%$ chance of that scenario being true. The figure also shows the estimated parameters for the planet or eclipsing binary for each scenario. The radius of the planet estimated by `TRICERATOPS` is $3.633787 R_{\oplus}$ which is close to the tabulated value on ExoFOP at $3.74186 \pm 0.251835 R_{\oplus}$. With these results, the planet can be considered validated. `TRICERATOPS`makes a fit for each scenario which can be seen in Fig. 2.6.

2.3.2 Mock data

For the TTV systems, I decided to use mock data. The mock data is generated through transit injections to explore the limits of `TRICERATOPS`. The transit injections utilized the

---

[3]`https://triceratops.readthedocs.io/en/latest/tutorials/example.html` Accessed: 3 March 2024

	ID	scenario	M_s	R_s	P_orb	inc	b	ecc	w	R_p	M_EB	R_EB	prob
0	429304876	TP	0.850000	1.084330	6.839239	87.980103	0.593554	0.577973	235.474601	3.633787	0.000000	0.000000	7.579295e-01
1	429304876	EB	0.850000	1.084330	6.839239	75.751514	1.394115	0.670712	63.381528	0.000000	0.783774	0.811763	6.389914e-192
2	429304876	EBx2P	0.850000	1.084330	13.678478	87.730231	0.614106	0.668385	185.226119	0.000000	0.817094	0.843523	8.482699e-44
3	429304876	PTP	0.850000	1.084330	6.839239	88.109004	0.544413	0.399329	234.513355	3.705607	0.000000	0.000000	1.796638e-01
4	429304876	PEB	0.850000	1.084330	6.839239	81.881005	1.514671	0.369294	53.923992	0.000000	0.683437	0.712706	1.270241e-186
5	429304876	PEBx2P	0.850000	1.084330	13.678478	87.786531	0.531576	0.701878	182.216652	0.000000	0.830489	0.856911	1.182280e-22
6	429304876	STP	0.646706	0.669427	6.839239	89.988079	0.003525	0.407207	185.009028	5.014755	0.000000	0.000000	1.307255e-02
7	429304876	SEB	0.155155	0.186992	6.839239	89.594051	0.093584	0.829828	170.813565	0.000000	0.052768	0.100000	1.410987e-23
8	429304876	SEBx2P	0.162051	0.194031	13.678478	89.469740	0.335729	0.834034	200.235213	0.000000	0.157401	0.189308	2.670071e-17
9	429304876	DTP	0.850000	1.084330	6.839239	87.888369	0.602267	0.681419	236.147014	3.508042	0.000000	0.000000	4.122017e-02
10	429304876	DEB	0.850000	1.084330	6.839239	81.843137	1.582582	0.355248	55.938622	0.000000	0.799409	0.826464	4.900090e-183
11	429304876	DEBx2P	0.850000	1.084330	13.678478	87.853329	0.329512	0.838154	187.427992	0.000000	0.825734	0.852103	1.383340e-13
12	429304876	BTP	1.100000	1.321996	6.839239	87.136796	0.602593	0.339127	202.801208	9.846427	0.000000	0.000000	8.113786e-03
13	429304876	BEB	1.077000	1.366601	6.839239	87.916046	0.520984	0.229005	291.299867	0.000000	0.154884	0.186711	2.898272e-08
14	429304876	BEBx2P	0.616000	0.551854	13.678478	89.219011	0.482649	0.368264	158.882575	0.000000	0.588072	0.551854	1.733612e-07

Figure 2.5: Table generated by `TRICERATOPS` listing the scenarios and the probabilities of the likelihood of these scenarios. This only shows the results for one `TRICERATOPS` run. The final result is the average FPP of 10 runs [Giacalone et al., 2021, Giacalone and Dressing, 2020].
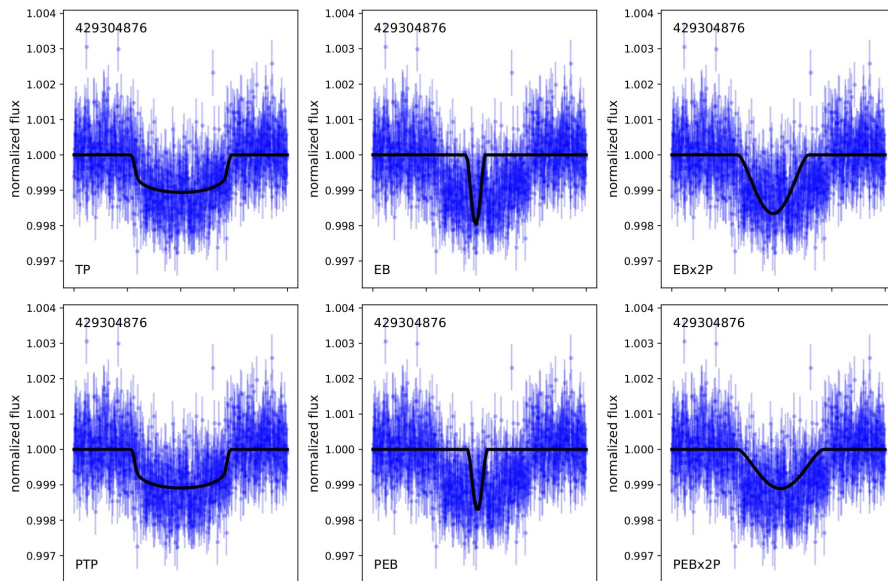

Figure 2.6: Figure showing the `TRICERATOPS` fit for six of the scenarios compared to the light curve for TOI-682 where the scenario TP was the best fit. The x-axis is the days from the transit center with the middle tick being 0 with the distance to each tick being 0.1 days [Giacalone et al., 2021, Giacalone and Dressing, 2020].
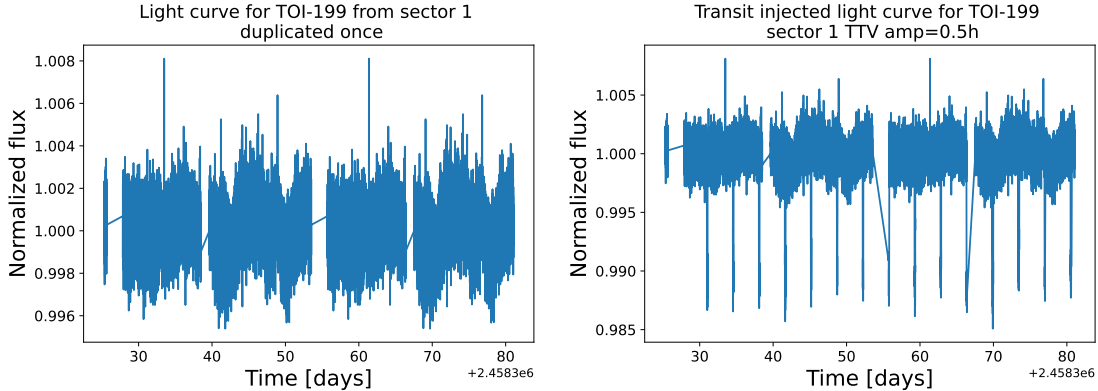
Figure 2.7: On the left, light curve from TOI-199 at sector 1. The light curve was duplicated once to increase the time for more transits. On the right, Transits with a TTV amplitude of 30 minutes injected into the light curve from TOI-199.

`PyTransit` package for the quadratic transit model [Parviainen, 2015]. To inject a transit, I needed a raw light curve with no obvious transits. I chose TOI-199 in Sector 1, seen in Fig. 2.7. TOI-199 is a G-type star with two long-period planets [Hobson et al., 2023]. I take a sector with no apparent transit. I duplicated the sector so that I could fit more injected transits. More transits increase the signal strength when I phase-fold them later and it will potentially have more TTVs. Each sector is roughly 30 days so the light curve spans about 60 days. I could then inject the transits with TTVs using a code provided to me by Judith Korth. It reads the normal light curve and then injects the transit which is altered by different parameters. These parameters are the TTV amplitude, TTV period, orbital period, impact parameter, stellar density, and transit depth. The TTV model is created according to the sinusoidal TTV model from the equation

$$TTV_{amplitude} \cdot sin((2 \cdot \pi \cdot tcs_{linear})/TTV_{period} + TTV_{phase}), \tag{2.7}$$

where tcs_{lin} is the transit center time following the linear ephemeris. The idea was to individually vary the TTV amplitude, TTV period, and impact parameter. While one of these parameters is being varied, the rest are constant. This is done to see how much each parameter affects `TRICERATOPS` and to see at what point it can no longer validate a planet. The other parameters (the orbital period, transit depth, and stellar density) are kept constant throughout the whole process. Once the injected light curve is done it can be saved and used on `TRICERATOPS`.

I decided that the injected transit should come from a Hot Jupiter because of their large transit depths because they can resemble eclipsing binaries in their transit signals at higher impact parameters. For the simulation to be as representative as possible, I wanted to use the average orbital period for Hot Jupiters as the mean period for the TTV transit in-
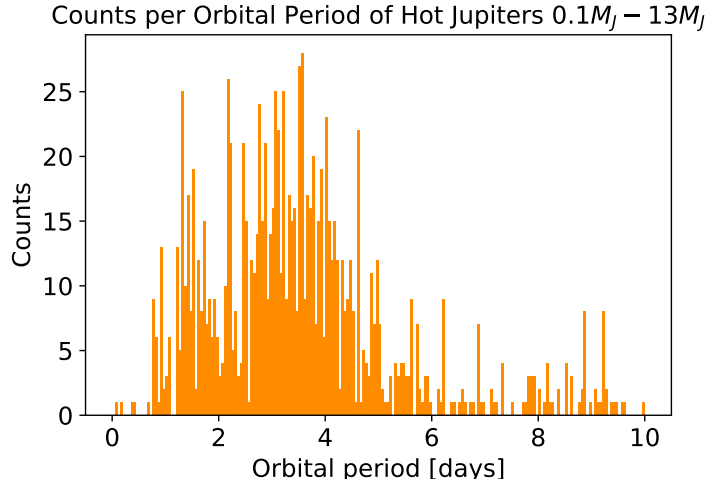
Figure 2.8: Histogram showing the amount of Hot Jupiters for each period interval. The x-axis has been divided into 200 bins.

jections. From NASA's exoplanet archive[4], I downloaded the data of all known transiting planets with masses $0.1M_J - 13M_J$ and with $P \leq 10$ days and I put them into a histogram. The histogram is in Fig. 2.8. I determined the mean orbital period to be $<P> \approx 3.5 \pm 1.8$ days.

I set the stellar density to be equal to the stellar density of TOI-199. The stellar density is ~ 1.83 g/cm^3, taken from ExoFOP. The radius ratio between the planet and the star is 0.1. This will give a transit depth of 0.01, from Eq. 1.3. The light curve with the injected transits can be seen in Fig. 2.7.

2.3.3 TTV system

The process of using `TRICERATOPS` for a system with TTVs is the same as in Section 2.3.1 with a couple of exceptions. The step with `lightkurve` is skipped because I created the used light curve in Section 2.3.2. Additionally, a mask was included to truncate the light curve at the probability calculation step. The mask considers only the transit data between -0.3 and 0.3 days from the transit center. This is done so that there is no excess of data that will prolong the calculation time for the probabilities. The transit curve is still fully visible so no important information is lost.

Varying TTV amplitude

I set the TTV period constant at 60 days and the impact parameter constant at $b = 0$. I injected a light curve with different TTV amplitudes from 0 hours to 0.6 hours in 0.05-hour

---

[4]`https://exoplanetarchive.ipac.caltech.edu/` Accessed: March 22

increments, while also calculating the relative TTV amplitude of the transit duration so that the results can be used on a signal with different stellar and planetary parameters as well. Then I phase-folded the light curves while not accounting for TTVs, seen on the Fig. 2.1. This means that the phase folding follows the linear approximation of the planet's position following the mean orbital period I determined for a Hot Jupiter. I saved the transit model as a .fits file and used it on `TRICERATOPS` which calculated the FPP and likely scenarios for all the amplitudes.

Varying TTV period

When varying the TTV period, I decided to test many different periods. The TTV amplitude was set constant at the two cases, at the limits of where `TRICERATOPS` could validate planets. The impact parameter is still at $b = 0$. The TTV periods I tested are 20, 40, 50, 60, 70, 80, 100, 120, and 600 days. I also wanted to see what would happen if the TTV period was double that of the orbital period. Additionally, I had the idea to change the TTV phase to see if that would also affect the resulting FPP as I increased the TTV period. I changed the phase from 0 to 0.2 and to 0.4. I injected the light curve with transits with these properties and input them into `TRICERATOPS`.

Varying TTV impact parameter

When I wanted to test the impact parameter, I tested it at the same TTV amplitudes as the TTV period case. I tested `TRICERATOPS` to see if it could distinguish a planet with high impact parameter from an eclipsing binary and to see how the FPP changes with higher impact parameters. I kept the period constant throughout the process at 60 days. I started the impact parameter at $b = 0.1$ with the idea of increasing the impact parameter by 0.1 until I reached 1. I injected the light curves with these properties and input them into `TRICERATOPS`.

Chapter 3

Results and Discussion

Varying TTV amplitude

By checking the FPP for a different amount of TTV amplitudes ranging from 0 to 0.6 hours, I found that `TRICERATOPS` could no longer validate a planet with the stellar and planetary parameters that I have chosen at a TTV amplitude of between 0.1 and 0.15 hours when not correcting for TTVs. This is between 6 and 9 minutes. At the TTV amplitude at 0.15 hours, the FPP was in the likely planet section but with a substantial standard deviation. The FPP for the TTV amplitude at 0.1 hours which corresponds to a relative TTV amplitude of 3.58% of the transit duration, is $FPP \approx 0.007 \pm 0.021$ with a low standard deviation and for any lower TTV amplitudes, the FPP is within the same range for FPP and standard deviation. For the TTV amplitude of 0.15 hours, a relative TTV amplitude of 5.37% of the transit duration, the $FPP \approx 0.425 \pm 0.464$. The FPP first reaches the not-a-planet zone at 0.25 hours or 15 minutes, a relative TTV amplitude of 8.95% of the transit duration, $FPP \approx 0.6 \pm 0.490$, again with a substantial standard deviation. Once the FPP reaches the region around $FPP = 1$ the standard deviation becomes smaller. This is at a TTV amplitude of 0.5 hours, a relative TTV amplitude of 17.9% of the transit duration, in this case. The FPP is $FPP \approx 1$ and continues at higher TTV amplitudes. In Fig. 3.1, the FPPs of the measured TTV amplitudes can be seen both when the TTV has been corrected for and when it has not been corrected.

The results suggest that `TRICERATOPS` does not work well at relative TTV amplitudes larger than 3.58% of the transit duration. `TRICERATOPS` becomes very unreliable in the region between 3.58% and 17.9% while not validating anything above 17.9%. While the average from 10 runs is taken, the standard deviation is very large, and most of the time the standard deviation spans multiple regions of between TTV amplitudes of 0.15 hours and 0.45 hours. With this, I can not confidently say that what we have is a validated planet or even a likely planet that we can follow up on and it is a big problem with `TRICERATOPS`. If the light curve is not phase-folded properly, the results may not be reliable.
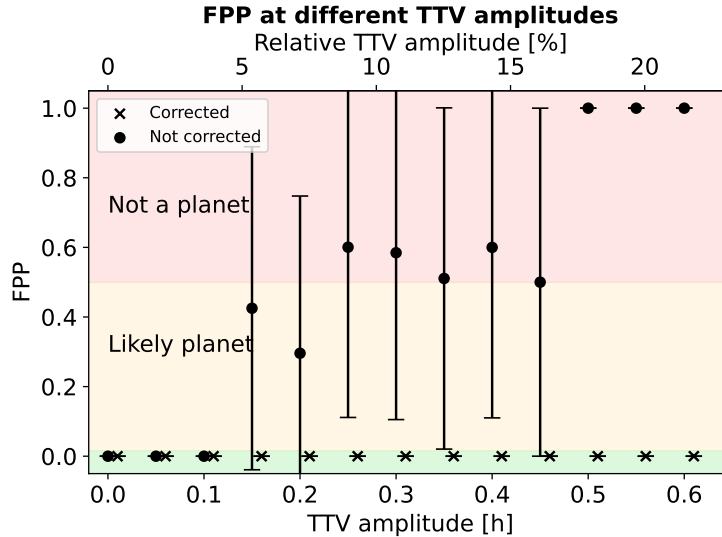
Figure 3.1: Figure showing the calculated FPPs for a range of TTV amplitudes with the TTV period set to 60 days and impact parameter set to 0. The points for the corrected and uncorrected TTVs that are next to each other are at the same TTV amplitude but are shifted slightly off each other to improve readability.

Varying TTV period

Changing the TTV period to the listed TTV periods in Section 2.3.3. The TTV amplitudes I used were 0.1 hours where `TRICERATOPS` could accurately validate the planet and 0.15 hours where `TRICERATOPS` predicted a likely planet but with a large uncertainty. For the 0.1 hours TTV amplitude case, I have plotted the measured FPP for each TTV period in Fig. 3.2, it can be seen that the FPP is mostly in the validated region with the occasional spikes, most notably at a TTV period of 120 days with a small one at 50 days. Not pictured in the plot is the FPP for the TTV periods at 600 days because this provided no result for multiple TTV amplitudes similar to the TTV amplitude at 0.25 hours not providing any result for the TTV period of 120 days. The 600-day TTV period had $FPP = 0.1 \pm 0.3$ which is in the likely planet region. This means that at low enough TTV amplitudes, the TTV period does not affect the FPP too much. These are still acceptable FPPs for follow-up observations. For the 0.15 hours TTV amplitude case, I have plotted the measured FPP for each TTV period in Fig. 3.2. From the plot, we can see that the FPP ranges from validated at low TTV periods like 20 days, to likely with a higher FPP leading to a higher standard deviation for the middle region with 50 days as the peak. The FPP seems to become smaller from there, reaching the validated region at 120 days again. However, the 600-day TTV period gave a NaN as the result for when the TTV was corrected. I am unsure why this happened and it did continue at the 0.25-hour TTV amplitude as well. At a TTV amplitude of 0.25 hours, most points have increased FPP with all but the TTV period at 70 days being validated which can be seen in Fig. B.3. Additionally, the FPP
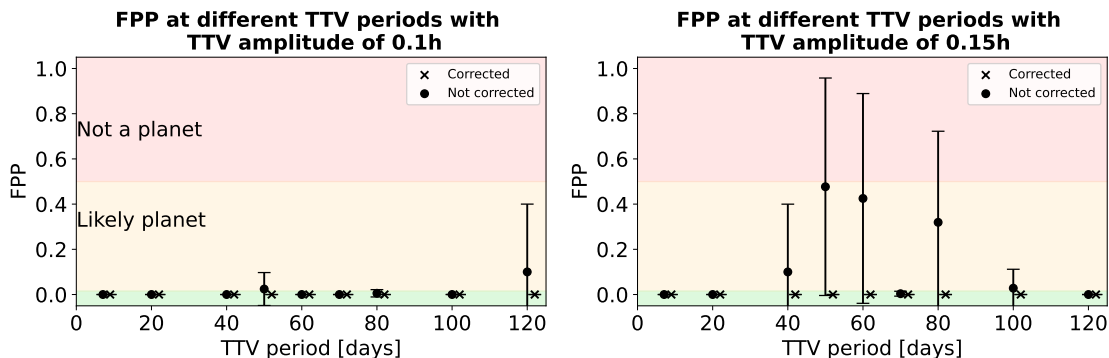
Figure 3.2: Figure showing the FPP for a range of TTV periods with the TTV amplitude set to 0.1 hours and impact parameter 0 on the left, and the FPP for the same range of TTV periods but with a TTV amplitude of 0.15 hours on the right. The corrected and uncorrected points that share the same TTV period have been shifted slightly away from each other to improve readability.

for the TTV periods of 50 and 60 days are not in the not-a-planet region. Additionally, the 600-day TTV period did not yield any results for the corrected case, it had a similar issue with the 120-day TTV period.

For the 0.1-hour TTV amplitude case, the spike at 120 seconds could attributed to how `TRICERATOPS` calculates the average FPP value, where a few of the 10 runs might have high FPPs, which throws off the average. Given that the FPP has consistently been low at this TTV amplitude for all periods, it appears that `TRICERATOPS` performs reasonably well. However, it is different for the 0.15-hour TTV amplitude case. The high FPPs at certain TTV periods, where `TRICERATOPS` struggles to confidently validate a planet and only classifies them as likely planets, clashes with the low FPPs at other TTV periods like 20 days and 120 days. The 0.25-hour TTV amplitude case also shows no clear pattern in the FPP values. One explanation for that random pattern could be the way I am sampling the TTV points at a constant grid with few points. In such a case a sampling feature problem called aliasing could occur. This can happen when there are not enough samples or the samples align in a way that causes the reconstructed sinusoidal wave to differ from the expected sine wave by the signal, as shown in Fig. 3.3.

Increasing the number of samples may overcome this problem as long as the TTV period is not a multiple of the orbital period. To take more samples, more transits need to be added to the mock data. When we observe real exoplanets, the observing time needs to be increased to gather more data. The TTV period affects the validation of TTVs in that the transit may align in such a way that not accounting for the TTV may make it impossible for `TRICERATOPS` to validate a planet. The inverse is also true, however. The transits may line up in such a way that completely negates the TTV if the TTV period is double the orbital
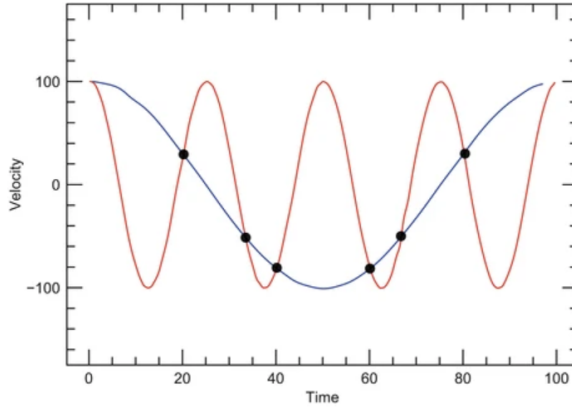
Figure 3.3: Figure showing how multiple waves can be fit to the Radial Velocity data because of lack of data points. The long-period wave is an alias of the short-period wave [Hatzes, 2016]. It works similarly for the TTV period where the data points are the observations of the transit. If the data points align with or are near the mean orbital period, `TRICERATOPS` will think that the amplitude is small or non-existent.

period or if every time the planet transits, its orbital period is close to the mean period. Due to the lack of a pattern when the TTV period is changed with the low amount of data points I have, the TTV period can not be used to determine the FPP of the system reliably.

To explore the effect of aliasing, I changed the TTV phase to get different samplings of the TTV. For this, I changed the phase from 0 to 0.2. This shift could change the part of the sine wave where the transit is detected, potentially altering the results. For the 0.1-hour TTV amplitude case, the change in phase has no influence (Fig. B.4 in appendix). This is also the case for when the phase is increased to 0.4 (Fig. B.6 in appendix). This means that at low TTV amplitudes, the signal will either be validated or deemed suitable for follow-up investigations, regardless of the TTV period or phase. Similarly to the case with a phase of 0, no discernible pattern emerged for the 0.15-hour and 0.25-hour TTV amplitude cases when the phase was increased to 0.2 and 0.4, further indicating that the TTV period is not reliable for FPP calculations as it increases. These plots can be seen in Fig. B.5 and Fig. B.7, respectively. To overcome aliasing and possibly achieve accurate FPP calculations, more data points will be needed.

Varying TTV impact parameter

First I started the 0.1-hour TTV amplitude case where the FPP increases as the impact parameter is increased with the FPP increasing at a similar time as the corrected case. The FPPs have large standard deviations that cross through all three regions in the graph. That means that we can not be certain if these are good candidates for follow-up investigations. This can be seen in Fig. 3.4. The same goes for the 0.15-hour TTV amplitude case where already at 0 impact parameter, we have a high FPP corresponding to a likely
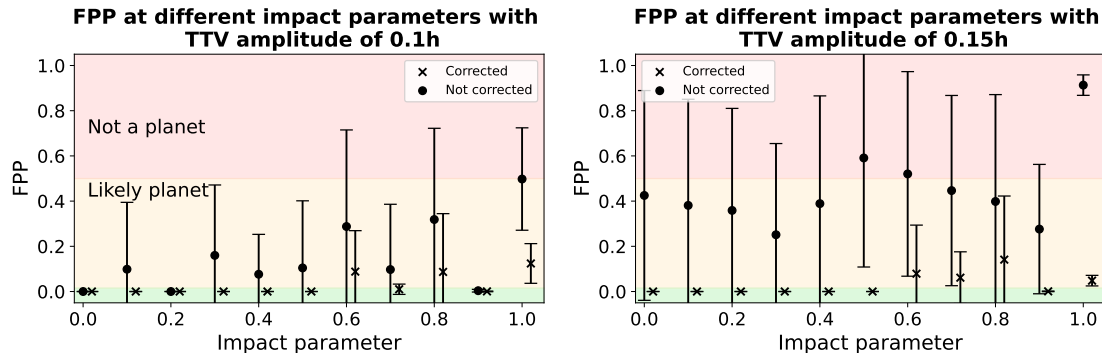
Figure 3.4: Figure showing the FPP as the impact parameter increases with a constant TTV amplitude at 0.1 hours and a constant TTV period at 60 days on the left, and for a TTV amplitude of 0.15 hours on the right. The corrected and uncorrected points that share the same impact parameter have been shifted slightly away from each other to improve readability.

planet and the FPP increases from there. This can be seen in Fig. 3.4. This is most likely due to the already high FPPs for these scenarios at $b = 0$ and from there, the FPP should increase as the impact parameter increases. It does so at $b = 0.5$ but starts dipping down at $b = 0.9$, possibly due how `TRICERATOPS` does the calculation for the FPP. Similarly, the 0.25-hour TTV amplitude stars in the not-a-planet region, as seen in Fig. B.8 (appendix). With the large standard deviations, it is still possible that it follows the trend of increasing the FPP with higher impact parameters but it is not explicitly visible. For the corrected TTV we can see that the FPP stays in the validated planet region until $b = 0.6$, which means that `TRICERATOPS` can be used to reliably classify transit signals when the impact parameter is 0.5 or less but remains unreliable for any uncorrected TTV.

`TRICERATOPS` can determine that a signal is a planet even with a high impact parameter when the TTV is corrected for but when it is not corrected for, `TRICERATOPS` does struggle with the validation of these planets, but it does not determine these signals to come from eclipsing binaries with a high probability until we reach an impact parameter of 0.9. This could be because the transit depth is still the same as for the planet we injected. If the transit depth had been deeper, `TRICERATOPS` may determine the signal to come from an eclipsing binary that does not have to graze the limb of the target star. So while `TRICERATOPS` does not get fooled by the high impact parameter with the shallow transit depth to think there is an eclipsing binary, it does still struggle with validation at the region of 0.1 hour TTV amplitude. With the large standard deviations, it is hard to determine if the candidate is good for follow-up investigations or not.

Chapter 4

Summary and conclusion

Because of the large number of exoplanet candidates waiting for confirmation, statistical validation tools were created to validate exoplanets and determine if they are worth doing follow-up investigations on. One of these tools is the Bayesian statistics tool `TRICERATOPS`. `TRICERATOPS` statistically determines the relative probability for a range of scenarios like if the signal is a transiting planet or an eclipsing binary for example and the FPP of the signal. `TRICERATOPS` works well when the orbital period is constant so I have tested `TRICERATOPS` to see how TTVs affect the validation of exoplanets. To do this I injected a transit into an empty light curve. These transits have TTVs and the strength of the TTV is determined by the TTV amplitude and TTV period. I also tested how `TRICERATOPS` would do with increasing impact parameters where the light curve starts to resemble the transit curve of an eclipsing binary.

By testing the viability of `TRICERATOPS` it can be seen that `TRICERATOPS` is not a useful tool for systems with a relative TTV amplitude of 5.37% of the transit duration and above. The FPPs go from a validated planet to a likely planet but with a large standard deviation showing that the FPP could possibly be in the validated planet and the not a planet regions. `TRICERATOPS` is very unreliable since the FPP result I have is an average of 10 FPPs because it is inconsistent between the 10 runs. It becomes consistent again at a relative TTV amplitude of 17.9% of the transit duration and above where $FPP \approx 1$. So `TRICERATOPS` is not reliable for use for systems with a relative TTV amplitude above 5.37% of the transit duration because the results are unreliable.

The TTV period is not a parameter that can be used for accurately determining the FPP, especially for relative TTV amplitudes above 5.38% of the transit duration. According to the plots I generated when changing the TTV period, the resulting FPP did not show a pattern for higher TTV amplitudes. This is possibly caused by aliasing. The same problem persisted when increasing the phase of the sinusoidal wave. The only reliable results were when the relative TTV amplitude was 5.38% of the transit duration and less where the FPP remained low.

For the impact parameter, it behaves similarly to what I previously expected with the FPP increasing when increasing the impact parameter. As the shape of the transit starts looking more like the shape made from eclipsing binaries but with no change to the transit depth, the FPP increases when we correct for TTVs and when we do not correct them. The high FPP values come from `TRICERATOPS` determining that the signal comes from either an eclipsing binary scenario of some kind or that the transiting planet is not transiting the target star but a companion. The plots show that if the TTV is corrected for, a signal with an impact parameter of 0.5 or less works well with `TRICERATOPS` while it remains unreliable for systems where the TTV is not corrected for. When the TTV is corrected for, they all have a sharper increase in FPP at around $b = 0.5 - 0.6$.

In conclusion, `TRICERATOPS` works well for systems with strict periodicity. When that periodicity is broken, it starts to struggle to validate planets with relative TTV amplitudes of 5.38% of the transit duration. This is a low percentage so when a system with TTV is being used on `TRICERATOPS`, the results will be unreliable at best. For example, the only known case where a HJ has TTVs, TOI-1130c, has a TTV amplitude of 12% relative to its transit duration and would fall in the problematic region Korth et al. [2023]. The TTVs have to be corrected for and even so, if the correction is not made carefully, the system being investigated could falsely be classified with high FPP due to how `TRICERATOPS` calculates the FPP. It is recommended to run the calculations many times and take the average FPP but when 9 out of 10 runs result in an FPP near 0 and the last run is 1, throws off the average. This can be enough to take a system from being a validated system to being a likely planet with a large standard deviation. So if you are using `TRICERATOPS`, be extra careful when correcting the TTVs.

Acknowledgements

---

[1] http://www.astropy.org
[2] https://matplotlib.org/
[3] https://numpy.org/

Bibliography

E. Agol and D. C. Fabrycky. Transit-Timing and Duration Variations for the Discovery and Characterization of Exoplanets. In H. J. Deeg and J. A. Belmonte, editors, *Handbook of Exoplanets*, page 7. 2018. doi: 10.1007/978-3-319-55333-7_7.

E. Agol, J. Steffen, R. Sari, and W. Clarkson. On detecting terrestrial planets with timing of giant planet transits. *MNRAS*, 359(2):567–579, May 2005. doi: 10.1111/j.1365-2966.2005.08922.x.

Astropy Collaboration, T. P. Robitaille, E. J. Tollerud, P. Greenfield, M. Droettboom, E. Bray, T. Aldcroft, M. Davis, A. Ginsburg, A. M. Price-Whelan, W. E. Kerzendorf, A. Conley, N. Crighton, K. Barbary, D. Muna, H. Ferguson, F. Grollier, M. M. Parikh, P. H. Nair, H. M. Unther, C. Deil, J. Woillez, S. Conseil, R. Kramer, J. E. H. Turner, L. Singer, R. Fox, B. A. Weaver, V. Zabalza, Z. I. Edwards, K. Azalee Bostroem, D. J. Burke, A. R. Casey, S. M. Crawford, N. Dencheva, J. Ely, T. Jenness, K. Labrie, P. L. Lim, F. Pierfederici, A. Pontzen, A. Ptak, B. Refsdal, M. Servillat, and O. Streicher. Astropy: A community Python package for astronomy. *A&A*, 558:A33, Oct. 2013. doi: 10.1051/0004-6361/201322068.

Astropy Collaboration, A. M. Price-Whelan, B. M. Sipőcz, H. M. Günther, P. L. Lim, S. M. Crawford, S. Conseil, D. L. Shupe, M. W. Craig, N. Dencheva, A. Ginsburg, J. T. Vand erPlas, L. D. Bradley, D. Pérez-Suárez, M. de Val-Borro, T. L. Aldcroft, K. L. Cruz, T. P. Robitaille, E. J. Tollerud, C. Ardelean, T. Babej, Y. P. Bach, M. Bachetti, A. V. Bakanov, S. P. Bamford, G. Barentsen, P. Barmby, A. Baumbach, K. L. Berry, F. Biscani, M. Boquien, K. A. Bostroem, L. G. Bouma, G. B. Brammer, E. M. Bray, H. Breytenbach, H. Buddelmeijer, D. J. Burke, G. Calderone, J. L. Cano Rodríguez, M. Cara, J. V. M. Cardoso, S. Cheedella, Y. Copin, L. Corrales, D. Crichton, D. D'Avella, C. Deil, É. Depagne, J. P. Dietrich, A. Donath, M. Droettboom, N. Earl, T. Erben, S. Fabbro, L. A. Ferreira, T. Finethy, R. T. Fox, L. H. Garrison, S. L. J. Gibbons, D. A. Goldstein, R. Gommers, J. P. Greco, P. Greenfield, A. M. Groener, F. Grollier, A. Hagen, P. Hirst, D. Homeier, A. J. Horton, G. Hosseinzadeh, L. Hu, J. S. Hunkeler, Ž. Ivezić, A. Jain, T. Jenness, G. Kanarek, S. Kendrew, N. S. Kern, W. E. Kerzendorf, A. Khvalko, J. King, D. Kirkby, A. M. Kulkarni, A. Kumar, A. Lee, D. Lenz, S. P. Littlefair, Z. Ma, D. M. Macleod, M. Mastropietro, C. McCully, S. Montagnac, B. M. Morris, M. Mueller, S. J. Mumford, D. Muna, N. A. Murphy, S. Nelson, G. H. Nguyen, J. P. Ninan, M. Nöthe,

S. Ogaz, S. Oh, J. K. Parejko, N. Parley, S. Pascual, R. Patil, A. A. Patil, A. L. Plunkett, J. X. Prochaska, T. Rastogi, V. Reddy Janga, J. Sabater, P. Sakurikar, M. Seifert, L. E. Sherbert, H. Sherwood-Taylor, A. Y. Shih, J. Sick, M. T. Silbiger, S. Singanamalla, L. P. Singer, P. H. Sladen, K. A. Sooley, S. Sornarajah, O. Streicher, P. Teuben, S. W. Thomas, G. R. Tremblay, J. E. H. Turner, V. Terrón, M. H. van Kerkwijk, A. de la Vega, L. L. Watkins, B. A. Weaver, J. B. Whitmore, J. Woillez, V. Zabalza, and Astropy Contributors. The Astropy Project: Building an Open-science Project and Status of the v2.0 Core Package. *AJ*, 156(3):123, Sept. 2018. doi: 10.3847/1538-3881/aabc4f.

Astropy Collaboration, A. M. Price-Whelan, P. L. Lim, N. Earl, N. Starkman, L. Bradley, D. L. Shupe, A. A. Patil, L. Corrales, C. E. Brasseur, M. N"othe, A. Donath, E. Tollerud, B. M. Morris, A. Ginsburg, E. Vaher, B. A. Weaver, J. Tocknell, W. Jamieson, M. H. van Kerkwijk, T. P. Robitaille, B. Merry, M. Bachetti, H. M. G"unther, T. L. Aldcroft, J. A. Alvarado-Montes, A. M. Archibald, A. B'odi, S. Bapat, G. Barentsen, J. Baz'an, M. Biswas, M. Boquien, D. J. Burke, D. Cara, M. Cara, K. E. Conroy, S. Conseil, M. W. Craig, R. M. Cross, K. L. Cruz, F. D'Eugenio, N. Dencheva, H. A. R. Devillepoix, J. P. Dietrich, A. D. Eigenbrot, T. Erben, L. Ferreira, D. Foreman-Mackey, R. Fox, N. Freij, S. Garg, R. Geda, L. Glattly, Y. Gondhalekar, K. D. Gordon, D. Grant, P. Greenfield, A. M. Groener, S. Guest, S. Gurovich, R. Handberg, A. Hart, Z. Hatfield-Dodds, D. Homeier, G. Hosseinzadeh, T. Jenness, C. K. Jones, P. Joseph, J. B. Kalmbach, E. Karamehmetoglu, M. Kaluszy'nski, M. S. P. Kelley, N. Kern, W. E. Kerzendorf, E. W. Koch, S. Kulumani, A. Lee, C. Ly, Z. Ma, C. MacBride, J. M. Maljaars, D. Muna, N. A. Murphy, H. Norman, R. O'Steen, K. A. Oman, C. Pacifici, S. Pascual, J. Pascual-Granado, R. R. Patil, G. I. Perren, T. E. Pickering, T. Rastogi, B. R. Roulston, D. F. Ryan, E. S. Rykoff, J. Sabater, P. Sakurikar, J. Salgado, A. Sanghi, N. Saunders, V. Savchenko, L. Schwardt, M. Seifert-Eckert, A. Y. Shih, A. S. Jain, G. Shukla, J. Sick, C. Simpson, S. Singanamalla, L. P. Singer, J. Singhal, M. Sinha, B. M. SipHocz, L. R. Spitler, D. Stansby, O. Streicher, J. Sumak, J. D. Swinbank, D. S. Taranu, N. Tewary, G. R. Tremblay, M. d. Val-Borro, S. J. Van Kooten, Z. Vasovi'c, S. Verma, J. V. de Miranda Cardoso, P. K. G. Williams, T. J. Wilson, B. Winkel, W. M. Wood-Vasey, R. Xue, P. Yoachim, C. Zhang, A. Zonca, and Astropy Project Contributors. The Astropy Project: Sustaining and Growing a Community-oriented Open-source Project and the Latest Major Release (v5.0) of the Core Package. *ApJ*, 935(2):167, Aug. 2022. doi: 10.3847/1538-4357/ac7c74.

A. C. Cameron. Extrasolar planets: Astrophysical false positives. *Nature*, 492(7427):48–50, Dec. 2012. doi: 10.1038/492048a.

D. Charbonneau, T. M. Brown, D. W. Latham, and M. Mayor. Detection of Planetary Transits Across a Sun-like Star. *ApJ*, 529(1):L45–L48, Jan. 2000. doi: 10.1086/312457.

N. J. Cornish. The lagrange points. 1998. URL https://wmap.gsfc.nasa.gov/media/ContentMedia/lagrange.pdf.

H. J. Deeg and R. Alonso. Transit Photometry as an Exoplanet Discovery Method. In H. J. Deeg and J. A. Belmonte, editors, *Handbook of Exoplanets*, page 117. 2018. doi: 10.1007/978-3-319-55333-7_117.

R. F. Díaz, J. M. Almenara, A. Santerne, C. Moutou, A. Lethuillier, and M. Deleuil. PASTIS: Bayesian extrasolar planet validation - I. General framework, models, and performance. *MNRAS*, 441(2):983–1004, June 2014. doi: 10.1093/mnras/stu601.

S. Giacalone and C. D. Dressing. triceratops: Candidate exoplanet rating tool, Feb. 2020.

S. Giacalone, C. D. Dressing, E. L. N. Jensen, K. A. Collins, G. R. Ricker, R. Vanderspek, S. Seager, J. N. Winn, J. M. Jenkins, T. Barclay, K. Barkaoui, C. Cadieux, D. Charbonneau, K. I. Collins, D. M. Conti, R. Doyon, P. Evans, M. Ghachoui, M. Gillon, N. M. Guerrero, R. Hart, E. Jehin, J. F. Kielkopf, B. McLean, F. Murgas, E. Palle, H. Parviainen, F. J. Pozuelos, H. M. Relles, A. Shporer, Q. Socia, C. Stockdale, T.-G. Tan, G. Torres, J. D. Twicken, W. C. Waalkes, and I. A. Waite. Vetting of 384 TESS Objects of Interest with TRICERATOPS and Statistical Validation of 12 Planet Candidates. *AJ*, 161(1):24, Jan. 2021. doi: 10.3847/1538-3881/abc6af.

G. J. Gilbert. Accurate modeling of grazing transits using umbrella sampling. *The Astronomical Journal*, 163(3):111, Feb. 2022. ISSN 1538-3881. doi: 10.3847/1538-3881/ac45f4. URL http://dx.doi.org/10.3847/1538-3881/ac45f4.

C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, Sept. 2020. doi: 10.1038/s41586-020-2649-2. URL https://doi.org/10.1038/s41586-020-2649-2.

A. P. Hatzes. The Radial Velocity Method for the Detection of Exoplanets. In V. Bozza, L. Mancini, and A. Sozzetti, editors, *Methods of Detecting Exoplanets: 1st Advanced School on Exoplanetary Science*, volume 428 of *Astrophysics and Space Science Library*, page 44, Jan. 2016. doi: 10.1007/978-3-319-27458-4_1.

A. P. Hatzes, W. D. Cochran, M. Endl, B. McArthur, D. B. Paulson, G. A. H. Walker, B. Campbell, and S. Yang. A planetary companion to cephei a. *The Astrophysical Journal*, 599(2):1383–1394, Dec. 2003. ISSN 1538-4357. doi: 10.1086/379281. URL http://dx.doi.org/10.1086/379281.

M. J. Hobson, T. Trifonov, T. Henning, A. Jordán, F. Rojas, N. Espinoza, R. Brahm, J. Eberhardt, M. I. Jones, D. Mekarnia, D. Kossakowski, M. Schlecker, M. Tala Pinto, P. J. Torres Miranda, L. Abe, K. Barkaoui, P. Bendjoya, F. Bouchy, M. Buttu, I. Carleo, K. A. Collins, K. D. Colón, N. Crouzet, D. Dragomir, G. Dransfield, T. Gasparetto, R. F. Goeke, T. Guillot, M. N. Günther, S. Howard, J. M. Jenkins, J. Korth, D. W. Latham,

M. Lendl, J. J. Lissauer, C. R. Mann, I. Mireles, G. R. Ricker, S. Saesen, R. P. Schwarz, S. Seager, R. Sefako, A. Shporer, C. Stockdale, O. Suarez, T.-G. Tan, A. H. M. J. Triaud, S. Ulmer-Moll, R. Vanderspek, J. N. Winn, B. Wohler, and G. Zhou. TOI-199 b: A Well-characterized 100 day Transiting Warm Giant Planet with TTVs Seen from Antarctica. *AJ*, 166(5):201, Nov. 2023. doi: 10.3847/1538-3881/acfc1d.

J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007. doi: 10.1109/MCSE.2007.55.

B. Kirk, K. Conroy, A. Prša, M. Abdul-Masih, A. Kochoska, G. Matijevič, K. Hambleton, T. Barclay, S. Bloemen, T. Boyajian, L. R. Doyle, B. J. Fulton, A. J. Hoekstra, K. Jek, S. R. Kane, V. Kostov, D. Latham, T. Mazeh, J. A. Orosz, J. Pepper, B. Quarles, D. Ragozzine, A. Shporer, J. Southworth, K. Stassun, S. E. Thompson, W. F. Welsh, E. Agol, A. Derekas, J. Devor, D. Fischer, G. Green, J. Gropp, T. Jacobs, C. Johnston, D. M. LaCourse, K. Saetre, H. Schwengeler, J. Toczyski, G. Werner, M. Garrett, J. Gore, A. O. Martinez, I. Spitzer, J. Stevick, P. C. Thomadis, E. H. Vrijmoet, M. Yenawine, N. Batalha, and W. Borucki. Kepler Eclipsing Binary Stars. VII. The Catalog of Eclipsing Binaries Found in the Entire Kepler Data Set. *AJ*, 151(3):68, Mar. 2016. doi: 10.3847/0004-6256/151/3/68.

J. Korth, D. Gandolfi, J. Šubjak, S. Howard, S. Ataiee, K. A. Collins, S. N. Quinn, A. J. Mustill, T. Guillot, N. Lodieu, A. M. S. Smith, M. Esposito, F. Rodler, A. Muresan, L. Abe, S. H. Albrecht, A. Alqasim, K. Barkaoui, P. G. Beck, C. J. Burke, R. P. Butler, D. M. Conti, K. I. Collins, J. D. Crane, F. Dai, H. J. Deeg, P. Evans, S. Grziwa, A. P. Hatzes, T. Hirano, K. Horne, C. X. Huang, J. M. Jenkins, P. Kabáth, J. F. Kielkopf, E. Knudstrup, D. W. Latham, J. Livingston, R. Luque, S. Mathur, F. Murgas, H. L. M. Osborne, E. Palle, C. M. Persson, J. E. Rodriguez, M. Rose, P. Rowden, R. P. Schwarz, S. Seager, L. M. Serrano, L. Sha, S. A. Shectman, A. Shporer, G. Srdoc, C. Stockdale, T. G. Tan, J. K. Teske, V. Van Eylen, A. Vanderburg, R. Vanderspek, S. X. Wang, and J. N. Winn. TOI-1130: A photodynamical analysis of a hot Jupiter in resonance with an inner low-mass planet. *A&A*, 675:A115, July 2023. doi: 10.1051/0004-6361/202244617.

E. Landau. How do you find – and confirm – a planet? 10 things about the search for exoplanets, 2018. URL https://exoplanets.nasa.gov/news/1524/how-do-you-find-and-confirm-a-planet-10-things-about-the-search-for-exoplanets/.

A. Lecavelier des Etangs and J. J. Lissauer. The iau working definition of an exoplanet. *New Astronomy Reviews*, 94:101641, June 2022. ISSN 1387-6473. doi: 10.1016/j.newar.2022.101641. URL http://dx.doi.org/10.1016/j.newar.2022.101641.

Lightkurve Collaboration, J. V. d. M. Cardoso, C. Hedges, M. Gully-Santiago, N. Saunders, A. M. Cody, T. Barclay, O. Hall, S. Sagear, E. Turtelboom, J. Zhang, A. Tzanidakis, K. Mighell, J. Coughlin, K. Bell, Z. Berta-Thompson, P. Williams, J. Dotson, and G. Barentsen. Lightkurve: Kepler and TESS time series analysis in Python. Astrophysics Source Code Library, Dec. 2018.

K. Mandel and E. Agol. Analytic Light Curves for Planetary Transit Searches. *ApJ*, 580 (2):L171–L175, Dec. 2002. doi: 10.1086/345520.

M. Mayor and D. Queloz. A Jupiter-mass companion to a solar-type star. *Nature*, 378 (6555):355–359, Nov. 1995. doi: 10.1038/378355a0.

M. Moe and R. Di Stefano. Mind Your Ps and Qs: The Interrelation between Period (P) and Mass-ratio (Q) Distributions of Binary Stars. , 230(2):15, June 2017. doi: 10.3847/1538-4365/aa6fb6.

T. D. Morton. An Efficient Automated Validation Procedure for Exoplanet Transit Candidates. *ApJ*, 761(1):6, Dec. 2012. doi: 10.1088/0004-637X/761/1/6.

H. Parviainen. PYTRANSIT: fast and easy exoplanet transit modelling in PYTHON. *MNRAS*, 450(April):3233–3238, 2015. doi: 10.1093/mnras/stv894. URL `http://mnras.oxfordjournals.org/cgi/doi/10.1093/mnras/stv894`.

S. N. Quinn, J. Horner, K. A. Collins, J. D. Eastman, L. Kreidberg, E. Rodriguez, Joseph, A. Vanderburg, R. A. Wittenmyer, G. Zhou, R. P. Butler, J. D. Crane, S. A. Shectman, J. K. Teske, I. B. Thompson, and S. X. Wang. TOI-682: two mini-Neptunes, one transiting. In *Posters from the TESS Science Conference II (TSC2)*, page 162, July 2021. doi: 10.5281/zenodo.5131353.

G. R. Ricker, J. N. Winn, R. Vanderspek, D. W. Latham, G. Á. Bakos, J. L. Bean, Z. K. Berta-Thompson, T. M. Brown, L. Buchhave, N. R. Butler, R. P. Butler, W. J. Chaplin, D. Charbonneau, J. Christensen-Dalsgaard, M. Clampin, D. Deming, J. Doty, N. De Lee, C. Dressing, E. W. Dunham, M. Endl, F. Fressin, J. Ge, T. Henning, M. J. Holman, A. W. Howard, S. Ida, J. M. Jenkins, G. Jernigan, J. A. Johnson, L. Kaltenegger, N. Kawai, H. Kjeldsen, G. Laughlin, A. M. Levine, D. Lin, J. J. Lissauer, P. MacQueen, G. Marcy, P. R. McCullough, T. D. Morton, N. Narita, M. Paegert, E. Palle, F. Pepe, J. Pepper, A. Quirrenbach, S. A. Rinehart, D. Sasselov, B. Sato, S. Seager, A. Sozzetti, K. G. Stassun, P. Sullivan, A. Szentgyorgyi, G. Torres, S. Udry, and J. Villasenor. Transiting Exoplanet Survey Satellite (TESS). *Journal of Astronomical Telescopes, Instruments, and Systems*, 1:014003, Jan. 2015. doi: 10.1117/1.JATIS.1.1.014003.

A. Santerne, R. F. Díaz, J. M. Almenara, F. Bouchy, M. Deleuil, P. Figueira, G. Hébrard, C. Moutou, S. Rodionov, and N. C. Santos. PASTIS: Bayesian extrasolar planet validation - II. Constraining exoplanet blend scenarios using spectroscopic diagnoses. *MNRAS*, 451(3):2337–2351, Aug. 2015. doi: 10.1093/mnras/stv1080.

P. W. Sullivan, J. N. Winn, Z. K. Berta-Thompson, D. Charbonneau, D. Deming, C. D. Dressing, D. W. Latham, A. M. Levine, P. R. McCullough, T. Morton, G. R. Ricker, R. Vanderspek, and D. Woods. The Transiting Exoplanet Survey Satellite: Simulations of Planet Detections and Astrophysical False Positives. *ApJ*, 809(1):77, Aug. 2015. doi: 10.1088/0004-637X/809/1/77.

G. Torres, M. Konacki, D. D. Sasselov, and S. Jha. Testing Blend Scenarios for Extrasolar Transiting Planet Candidates. I. OGLE-TR-33: A False Positive. *ApJ*, 614(2):979–989, Oct. 2004. doi: 10.1086/423734.

G. Torres, M. Konacki, D. D. Sasselov, and S. Jha. Testing Blend Scenarios for Extrasolar Transiting Planet Candidates. II. OGLE-TR-56. *ApJ*, 619(1):558–569, Jan. 2005. doi: 10.1086/426496.

J. Wang, D. A. Fischer, E. P. Horch, and X. Huang. On the occurrence rate of hot jupiters in different stellar environments. *The Astrophysical Journal*, 799(2):229, Jan. 2015. ISSN 1538-4357. doi: 10.1088/0004-637x/799/2/229. URL `http://dx.doi.org/10.1088/0004-637X/799/2/229`.

Appendix A

Additional information

Scenario priors

The prior probability of the orbital period $p(P'_{orb})$ is calculated by integrating the probability distribution $p(P_{orb})$ between $P'_{orb} - 0.1$ and $P'_{orb} + 0.1$.

$$p(P'_{orb}) = \int_{P'_{orb}-0.1}^{P'_{orb}+0.1} p(P_{orb})dP_{orb}. \tag{A.1}$$

The distribution for transiting planets comes from studies of planetary occurrence rates as a function of orbital period [Giacalone et al., 2021] and looks like,

$$p(P_{orb}) \sim \begin{cases} P_{orb}^{1.5} & 0.1 \text{ days} \leq P_{orb} \leq 10 \text{ days} \\ P_{orb}^{0.0} & 10 \text{ days} < P_{orb} \leq 50 \text{ days} \end{cases}, \tag{A.2}$$

The distribution for eclipsing binaries is based on data from the Kepler Eclipsing Binary Catalog [Kirk et al., 2016, Giacalone et al., 2021]. The distribution looks like this

$$p(P_{orb}) \sim \begin{cases} P_{orb}^{5.0} & 0.1 \text{ days} \leq P_{orb} \leq 0.3 \text{ days} \\ P_{orb}^{0.5} & 0.3 \text{ days} < P_{orb} \leq 50 \text{ days} \end{cases}, \tag{A.3}$$

Each scenario has a different set of parameter vectors θ_j. Parameter i is the inclination of the orbit and assuming an isotropic distribution of the orbits, the distribution of the inclinations will be given as,

$$p(i) \sim \sin i. \tag{A.4}$$

The inclination goes from $0°$ to $90°$ where at $90°$ the planet is edge-on. The frequency of discovered exoplanets increases with a higher inclination. The parameter R_p is the radius of the planet and depends on the host star. Giant planets, around $R_p > 6R_\oplus$, are 10 times as rare around M-type stars compared to the FGK stars. So the distributions of the radii around M-type and FGK-type stars are given as separate broken power laws

with the breaks at $R_p = 3R_\oplus$ where the rate stays constant below it and $R_p = 6R_\oplus$ to account for giants. The range is $0.5R_\oplus - 20R_\oplus$[Giacalone et al., 2021]. For M-type stars the distribution looks like,

$$p(R_p) \sim \begin{cases} R_p^{0.0} & 0.5\text{R}_\oplus \leq R_p \leq 3R_\oplus \\ R_p^{-7.0} & 3\text{R}_\oplus < R_p \leq 6R_\oplus \\ R_p^{-0.5} & 6\text{R}_\oplus < R_p \leq 20R_\oplus \end{cases} . \tag{A.5}$$

For FGK-types the distribution looks like,

$$p(R_p) \sim \begin{cases} R_p^{0.0} & 0.5\text{R}_\oplus \leq R_p \leq 3R_\oplus \\ R_p^{-4.0} & 3\text{R}_\oplus < R_p \leq 6R_\oplus \\ R_p^{-0.5} & 6\text{R}_\oplus < R_p \leq 20R_\oplus \end{cases} . \tag{A.6}$$

The parameters q_{short} and q_{long} are the mass ratios for stellar companions. q_{short} is between a star and its short-range stellar companion like a binary while q_{long} is for long-range stellar companions like unresolved bound companions. This is given as a broken power law with the break at 0.3 because the densities of binaries at $q > 0.3$ are well known. The range is 0.1-1[Giacalone et al., 2021]. For q_{short} the distribution looks like,

$$p(q_{short}) \sim \begin{cases} q_{short}^{0.3} & 0.1 \leq q \leq 0.3 \\ q_{short}^{-5.0} & 0.3 < q \leq 1.0 \end{cases} , \tag{A.7}$$

and for q_{long} the distribution looks like,

$$p(q_{long}) \sim \begin{cases} q_{long}^{0.3} & 0.1 \leq q \leq 0.3 \\ q_{long}^{-0.95} & 0.3 < q \leq 1.0 \end{cases} . \tag{A.8}$$

The excess of stellar twins is also considered. Stellar twins are stars that have a similar mass where the mass ratio is $q > 0.95$. $\mathcal{F}_{twin}$ is determined by taking the number of stars with $q > 0.95$ over the number of stars with $q > 0.3$. $\mathcal{F}_{twin}$ for q_{short} is $\mathcal{F}_{twin} = 0.3$ and for q_{long} it is $\mathcal{F}_{twin} = 0.05$.

The last parameter is the "simulated star" and it is the properties of a star simulated by TRILEGAL. These are used for scenarios with unresolved background stars where the lights from the target and the background star blend. To determine the properties of the blended stars, stars in a 0.1 deg^2 area around the target are simulated. By removing the simulated stars that are brighter than the target and dimmer than TESS magnitude 21, a distribution of possible background and foreground stars is created. For scenarios where a simulated star is needed, one of the possible background or foreground stars is selected[Giacalone et al., 2021].

Marginal likelihood continued

Equation 2.5 can not always be solved analytically so the marginal likelihood is determined by using arithmetic mean estimation and Monte Carlo sampling. It can then be approximated to

$$p(D|S_j) \sim \frac{1}{N} \sum_{n=1}^{N} p(D|\theta_j^{(n)}, S_j), \tag{A.9}$$

with $\theta_j^{(n)}$ being the nth sample from the parameter prior distribution and N is the total number of samples taken. A good amount of samples is $N = 10^6$. More samples lead to a more accurate approximation but will take longer to compute. Fewer samples and the variance in the marginal likelihood will be large. The number of parameters to marginalize over is reduced to reduce the variance. The period is fixed, the eccentricity is 0, mass, radius, and effective temperature are exact and can be taken from the TESS Input Catalogue (TIC) database if no parameters are given. If the star is not in the TIC database, stellar estimations can be made for the radius and temperature if the mass is given [Giacalone et al., 2021]. 84% of confirmed exoplanets with $P_{orb} < 30$ days have $e < 0.2$ and 72% of the confirmed eclipsing binaries with $P_{orb} < 10$ days have $e < 0.2$. The majority of TOIs have $P_{orb} < 30$ days so the assumption is valid. Unresolved background stars with no known estimates are given an assumption based on their mass [Giacalone et al., 2021].

$p(D|\theta_j^{(n)}, S_j)$ is calculated from

$$p(D|\theta_j^{(n)}, S_j) = p(D_{tra}|\theta_j^{(n)}, S_j) \cdot w^{(n)}, \tag{A.10}$$

with $p(D_{tra}|\theta_j^{(n)}, S_j)$ being the likelihood of transit data and $w^{(n)}$ being a weight for the ability to rule out unresolved companions by using high-resolution imaging to decrease the likelihood of the scenarios with unresolved companions. For scenarios with no companion stars $w^{(n)} = 1$. For scenarios with unresolved bound companions, the weight is calculated by first finding the magnitude difference between the primary and secondary stars. This is done using the mass of the target and the nth value of q_{long}. The contrast curve from high-resolution imaging is used to determine the angular separation where both stars would be distinguishable. With no high-resolution data, the angular separation is assumed to be 2.2"[Giacalone et al., 2021]. The angular separation can be converted to an orbital period using the parallax of the target star and the mass of both stars. This period with Equation (23) in [Moe and Di Stefano, 2017] calculates the frequency of bound stellar companions. A similar process is used for unresolved background and foreground stars aligned with the target star. These calculations are made from the results of the TRILEGAL simulations for the unresolved background star scenarios. The separation and the total number of simulated stars are used to estimate the frequency of unresolved stars aligned with the target[Giacalone et al., 2021].

The likelihood of transit data $p(D_{tra}|\theta_j^{(n)}, S_j)$ is calculated from

$$p(D_{tra}|\theta_j^{(n)}, S_j) \propto \prod exp\left[-\frac{1}{2}\left(\frac{y_l - f(t_l|\theta_j^{(n)})}{\sigma} \right)^2 \right],$$

(A.11)

with y_l being the flux of the lth data point, $f(t_l|\theta_j^{(n)})$ represents the flux given by the model for parameter vector $\theta_j^{(n)}$ at the time of the lth data point, and σ is the uncertainty of the flux. Equation A.11 is solved by modeling light curves according to Section **??**.

Appendix B

Plots and Figures

	ID	Tmag	Jmag	Hmag	Kmag	ra	dec	mass	rad	Teff	plx	sep (arcsec)	PA (E of N)	fluxratio	tdepth
0	429304876	9.2231	8.521	8.201	8.100	167.885855	-22.072551	0.85	1.084330	5085.0	11.997100	0.000	0.000	9.997757e-01	0.001213
1	901759724	19.2186	NaN	NaN	NaN	167.888936	-22.080197	NaN	NaN	NaN	-0.247224	29.385	159.524	2.459492e-05	0.000000
2	901759723	17.7358	NaN	NaN	NaN	167.890521	-22.080672	NaN	NaN	4895.0	0.739380	33.123	151.965	6.706475e-05	0.000000
3	901759725	20.8592	NaN	NaN	NaN	167.891346	-22.080533	NaN	NaN	NaN	NaN	34.078	147.481	3.460177e-06	0.000000
4	901759727	19.3954	NaN	NaN	NaN	167.894077	-22.062400	NaN	NaN	NaN	-0.914596	45.692	36.896	1.382913e-05	0.000000
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
64	901759748	19.8125	NaN	NaN	NaN	167.940924	-22.059848	NaN	NaN	NaN	1.131480	189.332	76.034	2.048527e-28	0.000000
65	429304869	18.7057	16.703	16.182	15.426	167.939879	-22.089771	NaN	NaN	3484.0	2.008170	190.584	108.992	1.778547e-31	0.000000
66	429303707	12.1184	11.315	10.888	10.802	167.877917	-22.018643	NaN	9.302920	5369.0	0.287458	195.867	352.227	9.194162e-26	0.000000
67	901759700	19.1763	NaN	NaN	NaN	167.898102	-22.126928	NaN	NaN	5640.0	-0.590572	199.976	168.215	1.696303e-35	0.000000
68	429303725	14.0672	13.348	12.937	12.865	167.825264	-22.075344	0.93	0.992926	5362.0	1.285620	202.387	267.141	1.029541e-25	0.000000

Figure B.1: Table generated by `TRICERATOPS` with information of each star within a 10-pixel radius of TOI-682. The important columns are the flux ratio, and transit depth, showing which stars can host the transit signal [Giacalone et al., 2021, Giacalone and Dressing, 2020].

	ID	Tmag	Jmag	Hmag	Kmag	ra	dec	mass	rad	Teff	plx	sep (arcsec)	PA (E of N)	fluxratio	tdepth
0	309792357	10.0391	9.329	8.952	8.814	80.105481	-59.895684	0.880000	0.878413	5180.0	9.749070	0.000	0.000	9.897351e-01	0.010104
1	309792355	16.8682	14.742	14.115	13.778	80.102390	-59.897981	0.129797	0.158376	3140.0	9.798490	9.976	214.024	1.849443e-03	0.000000
2	734471327	20.5255	NaN	NaN	NaN	80.115948	-59.896631	NaN	NaN	NaN	2.127010	19.204	100.228	5.703657e-05	0.000000
3	734471326	19.5896	NaN	NaN	NaN	80.094738	-59.897970	NaN	NaN	5598.0	-0.777054	21.072	247.004	1.480928e-04	0.000000
4	734471324	19.8114	NaN	NaN	NaN	80.094624	-59.893330	NaN	NaN	NaN	-0.035557	21.359	293.377	1.166708e-04	0.000000
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
97	734471305	19.4870	NaN	NaN	NaN	80.096016	-59.950465	NaN	NaN	4650.0	0.424476	197.950	184.945	2.730028e-22	0.000000
98	734472776	19.4038	NaN	NaN	NaN	80.005846	-59.919134	NaN	NaN	NaN	0.095399	198.673	244.811	4.320479e-23	0.000000
99	734471304	19.8245	NaN	NaN	NaN	80.044818	-59.941824	NaN	NaN	NaN	0.459465	198.927	213.359	2.615038e-22	0.000000
100	734471500	19.7264	NaN	NaN	NaN	80.205274	-59.871487	NaN	NaN	NaN	-0.536860	200.204	64.250	4.264864e-27	0.000000
101	382144212	17.3942	15.830	15.164	15.255	80.206040	-59.871505	NaN	NaN	NaN	NaN	201.422	64.439	1.691413e-26	0.000000

Figure B.2: Table generated by `TRICERATOPS` with information of each star within a 10-pixel radius of TOI-199. The important columns are the flux ratio and transit depth, showing which stars can host the transit signal [Giacalone et al., 2021, Giacalone and Dressing, 2020].
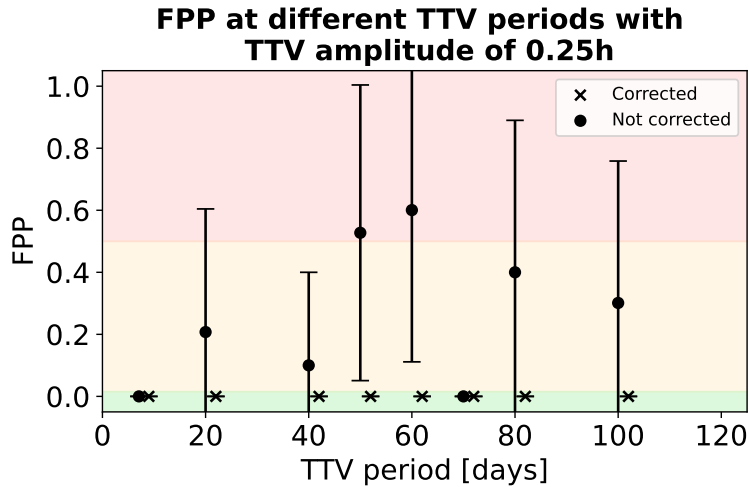

Figure B.3: Figure showing the FPP for a range of TTV periods with the TTV amplitude set to 0.25 hours and impact parameter 0. The corrected and uncorrected points that share the same TTV period have been shifted slightly away from each other to improve readability.

Figure B.4: Figure showing the FPP for a range of TTV periods when the TTV amplitude is constant at 0.1 hours, the impact parameter is constant at 0 and the phase is 0.2. The corrected and uncorrected points that share the same TTV period have been shifted slightly away from each other to improve readability.

Figure B.5: Figure showing the FPP for a range of TTV periods with the TTV amplitude set to 0.15 hours, impact parameter 0 and TTV phase 0.2 on the left, and the FPP for the same range of TTV periods but with a TTV amplitude of 0.25 hours on the right. The corrected and uncorrected points that share the same TTV period have been shifted slightly away from each other to improve readability.
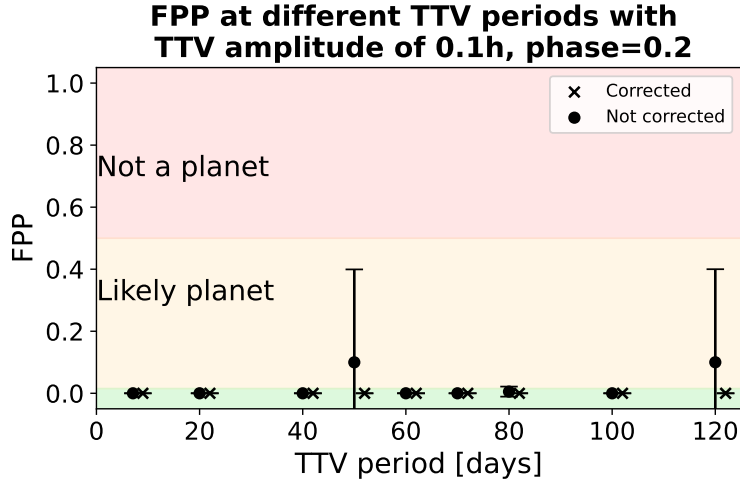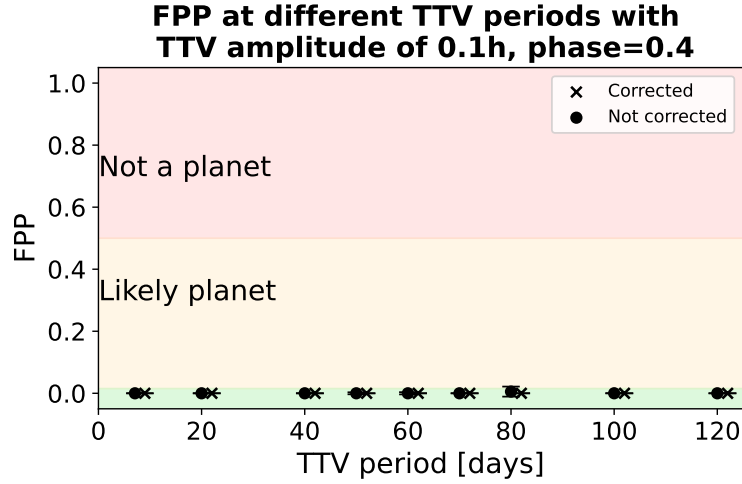
Figure B.6: Figure showing the FPP for a range of TTV periods when the TTV amplitude is constant at 0.1 hours, the impact parameter is constant at 0 and the phase is 0.4. The corrected and uncorrected points that share the same TTV period have been shifted slightly away from each other to improve readability.
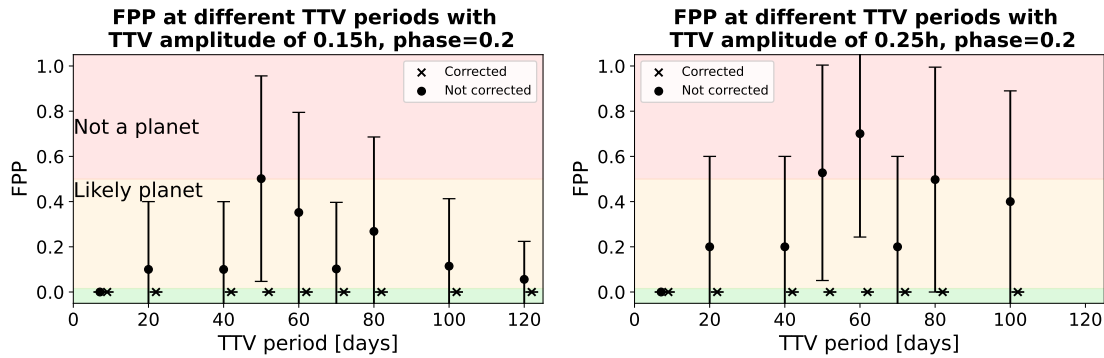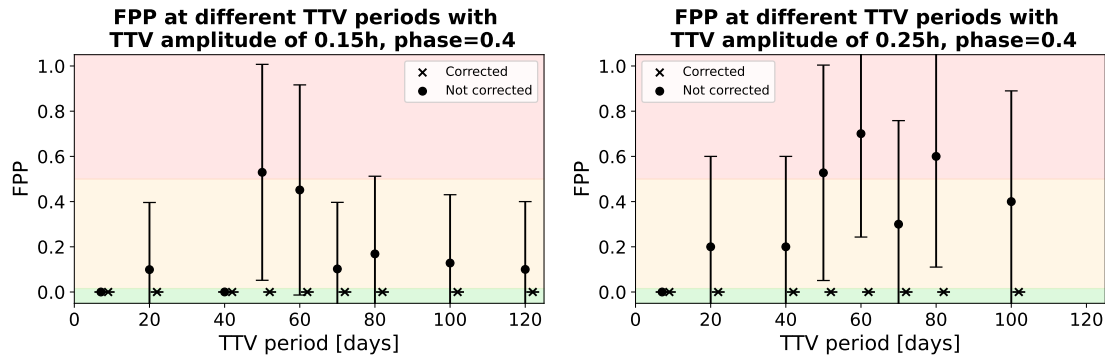

Figure B.7: Figure showing the FPP for a range of TTV periods with the TTV amplitude set to 0.15 hours, impact parameter 0 and TTV phase 0.4 on the left, and the FPP for the same range of TTV periods but with a TTV amplitude of 0.25 hours on the right. The corrected and uncorrected points that share the same TTV period have been shifted slightly away from each other to improve readability.
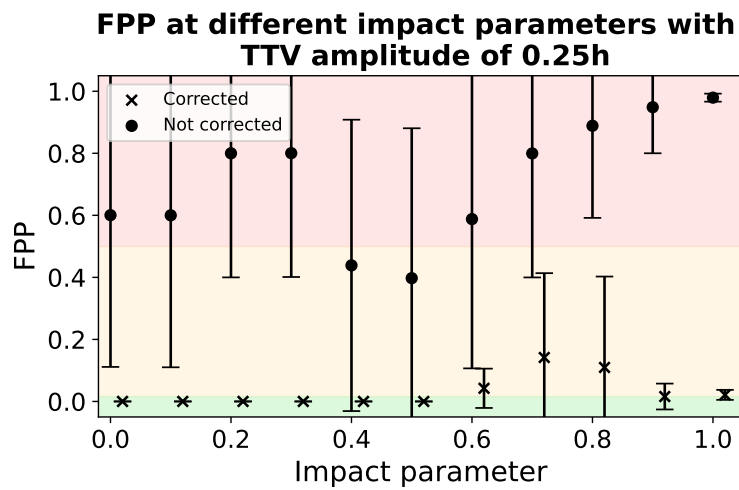
Figure B.8: Figure showing the FPP as the impact parameter increases with a constant TTV amplitude at 0.25 hours and a constant TTV period at 60 days. The corrected and uncorrected points that share the same impact parameter have been shifted slightly away from each other to improve readability.

Appendix C

Codes

Average orbital period for Hot Jupiters

```python
# -*- coding: utf-8 -*-
"""
Created on Fri Mar 22 16:07:28 2024

@author: warsa
"""

from astropy.io import ascii
import numpy as np
import matplotlib.pyplot as plt

file = ascii.read("C:/Users/warsa/Downloads/Hot_Jupiter_List_excluding_13
    +M.csv") #read csv file of all the periods
period = file["pl_orbper"] #only take the columns that are about the
    periods
print(period)

n = 10 #how many days we will show on the plot

std = np.std(period)
print(f'standard deviation = {std}')

mean = np.mean(period)
print(f'mean = {mean}')

median = np.median(period)
print(f'median = {median}')


plt.hist(period, bins=200, range=[0,n], color="darkorange")
plt.title("Counts per Orbital Period of Hot Jupiters \n $0.1M_J - 13M_J$"
    , fontsize=14)
plt.xlabel("Orbital period [days]", fontsize=14)
plt.ylabel("Counts", fontsize=14)
```

```
32  plt.tick_params(axis='both', which='major', labelsize=14)
33  plt.savefig("HotJupiterOrbitalPeriod.pdf")
34  plt.show()
```

Transit Injection

```
 1  #!/usr/bin/env python
 2  # coding: utf-8
 3
 4  # This notebook is an example on how to create a mock light curve and
        inject a transiting planet with an orbit that shows TTVs
 5
 6  # In[1]:
 7
 8
 9  get_ipython().run_line_magic('pylab', 'inline')
10
11
12  # In[2]:
13
14
15  from astropy.table import Table
16  from pytransit import QuadraticModel
17  from pytransit.orbits import as_from_rhop, i_from_ba, fold, epoch
18  from pytransit.orbits import d_from_pkaiews
19  import matplotlib.pyplot as plt
20
21
22  # First, we read a light curve to get time, flux and flux errors to
        create the mock light curve and where we inject the transiting planet
        . Here, we read a light curve of sector 1 of TIC 309792357
23
24  # In[3]:
25
26
27  tb = Table.read("tess2018206045859-s0001-0000000309792357-0120-s_lc.fits"
        )
28
29
30  # In[4]:
31
32
33  time = tb['TIME'].value.data+ tb.meta['BJDREFI']
34  flux = tb['PDCSAP_FLUX'].value.data
35  flux_err =  tb['PDCSAP_FLUX_ERR'].value.data
36  m = isfinite(time) & isfinite(flux)
37  time = time[m]
38  flux_err = flux_err[m]/median(flux[m])
39  flux = flux[m] / median(flux[m])
40
41
42  # Next, we create a long array based on the light curve data that we read
```

```
       . We create a longer light curve by duplicating the existing real
       light curve data. The light curve has a length of a typical TESS
       sector (~30 days), thus, we duplicate the data in terms of a TESS
       sector (nsectors)
43
44 # In[5]:
45
46
47 def create_final_time_and_flux_arrays(time, flux,flux_err, nsectors):
48     tt = zeros(time.size)
49     tt[1:] = diff(time)
50     final_time = time[0] + tile(tt, nsectors).cumsum()
51     final_flux = tile(flux, nsectors)
52     final_flux_err = tile(flux_err,nsectors)
53     return final_time, final_flux,final_flux_err
54
55 f_time, f_flux, f_flux_err =create_final_time_and_flux_arrays(time, flux,
       flux_err, 2)
56
57
58 # We plot the created light curve array
59
60 # In[6]:
61
62
63 xlabel('Time [days]', fontsize=18)
64 ylabel('Normalized flux', fontsize=18)
65 title('Light curve for TOI-199 from sector 1 \n duplicated once',
       fontsize=15)
66 plt.tick_params(axis='both', which='major', labelsize=12)
67 plot(f_time, f_flux)
68 plt.tight_layout()
69 #savefig('TOI-199_sector_1_light_curve.pdf')
70
71
72 # Now, we define some parameters needed for the transit injection and the
        TTV model.
73
74 # In[7]:
75
76
77 p = 3.539183488226514       # orbital period of the transiting planet in
       days
78 rho = 1.830659       # stellar density in unit of solar density
79 b = 0          # impact parameter
80 t0 = 0          # time of first transit
81 k = 0.1         # planet to star radius ratio
82
83 depth = k**2 # transit depth
84
85 ttv_ampl = 0.1/24 # ttv amplitude in days (h/24=days)
```

```python
86  ttv_per = 60 # ttv period in days
87
88
89  # We create a ttv_model according to a sinusoidal TTV with TTV amplitude,
         TTV period, and TTV phase. It needs the linear (unperturbed) transit
         center times
90
91  # In[8]:
92
93
94  def ttv_model(tcs_lin, ttv_ampl, ttv_per, ttv_phase = 0):
95      return ttv_ampl*sin((2*pi*tcs_lin)/ttv_per+ttv_phase)
96
97
98  # We create the transit signal that will be injected to the TESS light
        curve.
99
100 # In[9]:
101
102
103 def create_ttv_lc(time, p, rho, b, k, t0, ttv_ampl, ttv_per, ttv_phase =
        0):
104
105     a = as_from_rhop(rho,p)                       # scaled semi-major axis
106     inc = i_from_ba(b,a)                          # orbital inclination in
        radians
107     tm = QuadraticModel(interpolate=False)        # transit model assuming
        quadratic limb darkening
108
109     epochs = epoch(time, t0, p)                   # transit epochs (transit
         number)
110     phase_lin = fold(time, p, t0)                 # phase calculated based
        on the linear ephemeris
111     lcids = zeros(time.size,int)                  # light curve indices (
        see PyTransit documentation)
112     lcids[1:] = cumsum(clip(diff(epochs),0,1))
113     epids = unique(lcids)                         # epoch indices
114     tm.set_data(time, lcids, epids = epids)
115
116     tcs_lin = t0 + unique(epochs)*p               # transit centers based
        on linear ephemeris
117     ttv = ttv_model(tcs_lin, ttv_ampl, ttv_per)  # transit timing
        variations calcuated using the ttv_model
118     tcs_ttv = tcs_lin + ttv                       # transit centers based
        on the TTV model
119
120     phase_ttv = phase_lin - ttv[epids[lcids]]     # phase based on TTV
        model
121
122     return tcs_ttv,phase_lin, phase_ttv, tm.evaluate(k,[0.2,0.3],tcs_ttv,
        p,a,inc)
```

```
123
124
125 # We call the function create_ttv_lc
126
127 # In [10]:
128
129
130 tcs, ph_lin, ph_ttv, flux_model = create_ttv_lc(f_time, p, rho, b, k, t0,
        ttv_ampl, ttv_per)
131
132
133 # We plot the transit with a phase based on the linear and TTV model
134
135 # In [11]:
136
137
138 xlabel('Phase', fontsize=18)
139 ylabel('Normalized flux', fontsize=18)
140 title('Phase folded TTV and Linear Ephemeris \n TTV amp=0.1h', fontsize
        =15)
141 plt.tick_params(axis='both', which='major', labelsize=16)
142 xlim(-0.3, 0.3)
143 plot(ph_lin, flux_model * f_flux ,'.',label="Not Corrected")
144 plot(ph_ttv, flux_model,'.',label="Corrected", color='black')
145 legend()
146 plt.tight_layout()
147 #savefig('TTV_vs_Linear_0.1hr_amp_60days_per_b=0.pdf')
148
149
150 # We also plot the created light curve with the TTV model
151
152 # In [12]:
153
154
155 xlabel('Time [days]', fontsize=18)
156 ylabel('Normalized flux', fontsize=18)
157 title('Transit injected light curve for TOI-199 \n sector 1 TTV amp=0.1h'
        , fontsize=15)
158 plt.tick_params(axis='both', which='major', labelsize=12)
159 plot(f_time, f_flux * flux_model)
160 plt.tight_layout()
161 #savefig('TOI-199_sector_1_TTV_light_curve_0.1h_60days_TTVper_b=0.pdf')
162
163
164 # We save the injected transit model as a fits file. The Model parameters
        are saved in the fits header.
165
166 # In [13]:
167
168
169 tb = Table([f_time, ph_lin, ph_ttv, flux_model, f_flux*flux_model,
```

```
        f_flux_err],
170            names='time phase_lin phase_ttv model_flux flux flux_err'.split
        ())
171
172 tb.meta['t0'] = t0
173 tb.meta['p'] = p
174 tb.meta['k'] = k
175 tb.meta['b'] = b
176 tb.meta['rho'] = rho
177 tb.meta['ttv_ampl'] = ttv_ampl
178 tb.meta['ttv_per'] = ttv_per
179 tb.meta['ttv_pha'] = 0
180
181 for i,tc in enumerate(tcs):
182     tb.meta[f'tc_{i}'] = tc
183 #tb.write('test_ampl_0.1h_per_60_TOI-199_sector1_b=0.0_stellardensity
        =1.83_phase=0.fits', overwrite=True)
184
185
186 # In[ ]:
```

TRICERATOPS used on TOI-682

```
1 #!/usr/bin/env python
2 # coding: utf-8
3
4 # In[2]:
5
6
7 import numpy as np
8 import pandas as pd
9 import time
10 import lightkurve as lk
11 from lightkurve import TessLightCurve
12 import matplotlib.pyplot as plt
13 get_ipython().run_line_magic('matplotlib', 'inline')
14 import triceratops.triceratops as tr
15
16 #finding the transit curve
17 search_result = lk.search_lightcurve("TIC 429304876", author = "SPOC",
        cadence="short")
18 print(search_result)
19 lc = search_result.download()
20 lc = lc.normalize()
21 lc.plot()
22 print(lc)
23
24
25 # In[24]:
26
27
28 #phase fold
```

```
29 period = lc.to_periodogram("bls").period_at_max_power
30 folded_lc=lc.fold(period=period, epoch_time=1546.29)
31 folded_lc.plot(marker='.', linestyle='none', markersize=4)
32 print(period)
33
34 #truncate the plot/data
35 folded_truncated_lc = folded_lc.truncate(-0.2,0.2)
36 folded_truncated_lc.plot(marker='.', linestyle='none', markersize=4)
37
38 ###For saving data. If data is already saved, comment out this stuff
39
40 # Creating a table with time, flux, and flux error
41 folded_table = folded_truncated_lc.to_table()
42 #folded_table = folded_lc.to_table()
43 print(folded_table)
44
45 # Saving the Astropy table to a CSV file
46 # Saving the specified columns from Astropy table to a CSV file with
       headers
47 selected_columns = ["time", "flux", "flux_err"]
48 folded_table[selected_columns].write("TOI-682.csv", format="csv", names=
       selected_columns, overwrite=True)
49
50
51 # In[4]:
52
53
54 get_ipython().run_cell_magic('time', '', '\nID = 429304876\nsectors = np.
       array([9])\ntarget = tr.target(ID=ID, sectors=sectors)\n\nap = np.
       array([[913,881],[914,881],[915,881],\n
       [913,880],[914,880],[915,880],\n                    [913,879],[914,879]])\
       n\n#plot\ntarget.plot_field(sector=9, ap_pixels=ap)\n\n#table\ntarget
       .stars\n')
55
56
57 # In[26]:
58
59
60 apertures = np.array([ap])
61 target.calc_depths(tdepth=0.001213, all_ap_pixels=apertures) #tdepth is
       the transit depth in ppm/1,000,000
62
63 #table
64 target.stars
65
66
67 # In[29]:
68
69
70 get_ipython().run_cell_magic('time', '', '\n# read in the light curve\
       nlightcurve = pd.read_csv("TOI-682.csv", header=None)\n\n# Convert
```

```
       all columns to numeric values\nlightcurve = lightcurve.apply(pd.
       to_numeric, errors=\'coerce\')\n\n# Remove rows with NaN or infinite
        values\nlightcurve = lightcurve.dropna()\n\n# Extract time, flux, and
        flux_err\ntime, flux, flux_err = lightcurve[0].values, lightcurve
       [1].values, lightcurve[2].values\nP_orb = 6.8392388\n\
       nlightcurve_binsize = (time.max() - time.min()) / 100\nlightcurve =
       TessLightCurve(time=time, flux=flux, flux_err=flux_err).bin(
       time_bin_size=lightcurve_binsize)\nprint(lightcurve)\n\ntarget.
       calc_probs(\n     time=lightcurve.time.value,\n     flux_0=lightcurve.
       flux.value,\n     flux_err_0=np.mean(lightcurve.flux_err.value),\n
       P_orb=P_orb\n)\n')
71
72
73 # In[30]:
74
75
76 df_results = target.probs
77 print("FPP =", np.round(target.FPP, 4))
78 print("NFPP =", np.round(target.NFPP, 4))
79 df_results
80
81
82 # In[ ]:
83
84
85 target.plot_fits(time=time, flux_0=flux, flux_err_0=np.mean(flux_err),
       save=True)
86 #plt.savefig("test.png")
87
88
89 # In[36]:
90
91
92 get_ipython().run_cell_magic('time', '', '\nFPPs = np.zeros(10)\nfor i in
       range(10):\n     target.calc_probs(time=lightcurve.time.value,\n
                        flux_0=lightcurve.flux.value,\n
        flux_err_0=np.mean(lightcurve.flux_err.value),\n
         P_orb=P_orb,\n                          parallel=True,\n
            verbose=0)\n     FPPs[i] = target.FPP\n\nmeanFPP = np.round(np.
    mean(FPPs), 4)\nstdvFPP = np.round(np.std(FPPs), 4)\nprint("FPP =",
    meanFPP, "+/-", stdvFPP)\n')
93
94
95 # In[33]:
96
97
98 #selected_columns = ["time", "flux", "flux_err"]
99 #lc[selected_columns].write("TOI-682_full_lightcurve.csv", format="csv",
       names=selected_columns, overwrite=True)
100
101
```

```
102 # In [34]:
103
104
105 #print(lc.meta)
106
107
108 # In [35]:
109
110
111 #print(lightcurve.meta)
112
113
114 # In[ ]:
```

Example of how `TRICERATOPS` was used on TOI-199

```
1 #!/usr/bin/env python
2 # coding: utf-8
3
4 # How to apply 'triceratops' on a star using our injected transit model
      with TTVs
5
6 # In[1]:
7
8
9 import numpy as np
10 import pandas as pd
11 from astropy.table import Table
12 import time
13 from lightkurve import TessLightCurve
14 from numpy.random import seed
15 import matplotlib.pyplot as plt
16 get_ipython().run_line_magic('matplotlib', 'inline')
17
18 import triceratops.triceratops as tr
19
20
21 # ## Example #1
22 #
23 # $\textbf{First, let's apply the tool on the example we just created
      using TIC 358248442 and sector 69}$
24 #
25 # Begin by defining the target star object with the 'target(ID, sectors)'
       class. The arguments for this class are 'ID' (the TIC ID of the star
      ) and 'sectors' (the sectors in which the star was observed). 'ID'
      should be an integer and 'sectors' should be a numpy array.
26 #
27 #
28
29 # In[2]:
30
31
```

```
32 get_ipython().run_cell_magic('time', '', 'ID = 309792357\nsectors = np.
      array([1])\ntarget = tr.target(ID=ID, sectors=sectors)\n')
33
34
35 # Next, let's define the aperture used to extract the light curve, plot
      an image of the field, and display a table of nearby stars.
36 #
37 # $\textbf{Note, the aperture we take from the DS9}$
38 #
39 # The aperture should be a 2D numpy array with the formal '[[col#, row#],
       [col#, row#], [col#, row#], ...]', where where each '[col#, row#]'
      is the column and row number of the pixel.
40 #
41 # An image of the field can be plotted with the '.plot_field(sector,
      ap_pixels)' method. This method takes as optional arguments 'sector'
       (the image sector you would like to plot) and 'ap_pixels' (the
      aperture we just defined).
42 #
43 # We can display a table of the stars in the image with the '.stars'
      attribute. This table contains the separation and position angle of
      each star from the target star, so that each can be identified in the
       plot. Note that the scale of a TESS pixel is about 20 arcseconds.
44
45 # In[3]:
46
47
48 ap = np.array([                    [1353,1972], [1354,1972], [1355,1972],
49                  [1352,1971], [1353,1971], [1354,1971], [1355,1971],
      [1356,1971],
50                  [1352,1970], [1353,1970], [1354,1970], [1355,1970],
      [1356,1970],
51                  [1352,1969], [1353,1969], [1354,1969], [1355,1969],
      [1356,1969],
52                                                 [1354,1968], [1355,1968]])
53
54 target.plot_field(sector=1, ap_pixels=ap)
55
56 target.stars
57
58
59 # We can now determine which stars in the aperture are bright enough to
      produce the observed transit. $\textbf{The transit for injected
      transiting planet has a depth of ~10000 ppm}$. We'll use the '.
      calc_depths(tdepth, all_ap_pixels)' method to do this. This method
      takes as arguments 'tdepth' (the transit depth of the candidate) and
      'all_ap_pixels' (a numpy array of all apertures). After doing this,
      the '.stars' table includes the flux ratio contributed by each star
      in the aperture and the transit depth each star would have if it were
       the host of the signal.
60
61 # In[4]:
```

```
62
63
64  apertures = np.array([ap])
65  target.calc_depths(tdepth=0.01, all_ap_pixels=apertures)
66
67  target.stars
68
69
70  # After doing this, we can calculate the probability of each scenario
        using the '.calc_prob(time, flux_0, flux_err_0, P_orb)' method, which
         requires as arguments 'time' (times from phase-folded light curve in
         units of days from transit center), 'flux_0' (normalized flux from
        phase-folded light curve), 'flux_err_0' (flux error values of the
        target's phase-folded light curve), and 'P_orb' (orbital period of
        the TOI in days).
71  #
72  # If you have a lot of data points, I recommend binning your light curve
        down to ~100 to save time. Binning your data is poor practice when
        fitting light curves for a planet's parameters, but it doesn't yield
        significantly different results for our purposes. This should only
        take a few minutes.
73
74  # $\textbf{We do this:}$
75  #
76  # $\textbf{1. for the phase folded data accouted for the TTVs}$
77  #
78  # $\textbf{2. for the phase folded data based on the linear ephemeris}$
79  #
80  # $\textbf{considering the same random number}$
81
82  # In[5]:
83
84
85  get_ipython().run_cell_magic('time', '', '\n# read in the light curve\
        nlc_ttv = Table.read("test_ampl_0.1h_per_60_TOI-199_sector1_b=0
        _stellardensity=1.83.fits")\ntime, flux, flux_err = lc_ttv[\'
        phase_ttv\'].value.astype(\'d\'), lc_ttv[\'flux\'].value.astype(\'d
        \'), lc_ttv[\'flux_err\'].value.astype(\'d\')\nP_orb =
        3.539183488226514\n\nm = abs(time)<0.3\ntime = time[m]\nflux = flux[m
        ]\nflux_err = flux_err[m]\n\nseed(0)\n\nlc_ttv_binsize = (time.max()-
        time.min())/100\nlc_ttv = TessLightCurve(time=time, flux=flux,
        flux_err=flux_err).bin(time_bin_size=lc_ttv_binsize)\n\ntarget.
        calc_probs(time=lc_ttv.time.value, flux_0=lc_ttv.flux.value,
        flux_err_0=np.mean(lc_ttv.flux_err.value), P_orb=P_orb)\n')
86
87
88  # Now that that's done, let's check out a table of our results with the
        '.probs' attribute and calculate the false positive probability and
        nearby false positive probability using the '.FPP' and '.NFPP'
        attributes. We expect a scatter of a few percent in our probabilities
        , so don't be alarmed if it's slightly different with each run. We
```

```
     can also plot the transit fits of each scenario using the '.plot_fits
     (time, flux_0, flux_err_0)' method.
89
90 # In[6]:
91
92
93 df_results = target.probs
94 print("FPP =", np.round(target.FPP, 4))
95 print("NFPP =", np.round(target.NFPP, 4))
96 df_results
97
98
99 # In[7]:
100
101
102 target.plot_fits(time=time, flux_0=flux, flux_err_0=np.mean(flux_err))
103
104
105 # In[8]:
106
107
108 get_ipython().run_cell_magic('time', '', '\n# read in the light curve\
     nlc_lin = Table.read("test_ampl_0.1h_per_60_TOI-199_sector1_b=0
     _stellardensity=1.83.fits")\ntime, flux, flux_err = lc_lin[\'
     phase_lin\'].value.astype(\'d\'), lc_lin[\'flux\'].value.astype(\'d
     \'), lc_lin[\'flux_err\'].value.astype(\'d\'))\nP_orb =
     3.539183488226514\n\nm = abs(time)<0.3\ntime = time[m]\nflux = flux[m
     ]\nflux_err = flux_err[m]\n\nseed(0)\n\nlc_lin_binsize = (time.max()-
     time.min())/100\nlc_lin = TessLightCurve(time=time, flux=flux,
     flux_err=flux_err).bin(time_bin_size=lc_lin_binsize)\n\ntarget.
     calc_probs(time=lc_lin.time.value, flux_0=lc_lin.flux.value,
     flux_err_0=np.mean(lc_lin.flux_err.value), P_orb=P_orb)\n')
109
110
111 # In[9]:
112
113
114 df_results = target.probs
115 print("FPP =", np.round(target.FPP, 4))
116 print("NFPP =", np.round(target.NFPP, 4))
117 df_results
118
119
120 # In[10]:
121
122
123 target.plot_fits(time=time, flux_0=flux, flux_err_0=np.mean(flux_err),
     save=True)
124
125
126 # If you plan to quote these results in a paper, you should run '.
```

```
         calc_probs()' several times and quote the mean and standard deviation
           of the FPP. This way, you can show that your result is not sensitive
           to the intrinsic scatter in the calculation.
127 #
128 # We illustrate this below by calculating the FPP 20 times. We use the
           optional argument 'parallel' to enable parallel light curve
           simulation, which makes the calculation ~5x faster. Enabling
           parallelization yields the same results as the non-parallel option,
           so you should feel free to set this argument to 'True' at all times.
129
130 # In[11]:
131
132
133 get_ipython().run_cell_magic('time', '', 'n = 10\nFPPs = np.zeros(n)\ns0
           = np.zeros(n)\ns1 = np.zeros(n)\ns2 = np.zeros(n)\ns3 = np.zeros(n)\
           ns4 = np.zeros(n)\ns5 = np.zeros(n)\ns6 = np.zeros(n)\ns7 = np.zeros(
           n)\ns8 = np.zeros(n)\ns9 = np.zeros(n)\ns10 = np.zeros(n)\ns11 = np.
           zeros(n)\ns12 = np.zeros(n)\ns13 = np.zeros(n)\ns14 = np.zeros(n)\
           nfor i in range(n):\n      seed(i)\n      target.calc_probs(time=lc_ttv.
           time.value, \n                                flux_0=lc_ttv.flux.value, \n
                            flux_err_0=np.mean(lc_ttv.flux_err.value), \n
                        P_orb=P_orb,\n                                parallel=True,\n
                            verbose=0)\n      FPPs[i] = target.FPP\n      s0[i]=
           target.probs["prob"][0]\n      s1[i]=target.probs["prob"][1]\n      s2[i
           ]=target.probs["prob"][2]\n      s3[i]=target.probs["prob"][3]\n      s4[
           i]=target.probs["prob"][4]\n      s5[i]=target.probs["prob"][5]\n      s6
           [i]=target.probs["prob"][6]\n      s7[i]=target.probs["prob"][7]\n
           s8[i]=target.probs["prob"][8]\n      s9[i]=target.probs["prob"][9]\n
            s10[i]=target.probs["prob"][10]\n      s11[i]=target.probs["prob
           "][11]\n      s12[i]=target.probs["prob"][12]\n      s13[i]=target.probs
           ["prob"][13]\n      s14[i]=target.probs["prob"][14]\n\n\nmeanFPP = np.
           mean(FPPs)\nstdvFPP = np.std(FPPs)\nprint("FPP =", meanFPP, "+/-",
           stdvFPP)\n\ntb = Table([s0, s1, s2, s3, s4, s5, s6, s7, s8 ,s9 ,s10,
           s11, s12 ,s13, s14, FPPs], \n               names=\'TP EB EBx2P PTP PEB
           PEBx2P STP SEB SEBx2P DTP DEB DEBx2P BTP BEB BEBx2P FPP\'.split())\n\
           ntb.write(\'TOI-199_0.1h_amp_60days_per_TTV.csv\', overwrite=True)\n'
           )
134
135
136 # The FPP is favorable, but is not low enough to validate the planet.
137
138 # In[12]:
139
140
141 get_ipython().run_cell_magic('time', '', 'n = 10\nFPPs = np.zeros(n)\ns0
           = np.zeros(n)\ns1 = np.zeros(n)\ns2 = np.zeros(n)\ns3 = np.zeros(n)\
           ns4 = np.zeros(n)\ns5 = np.zeros(n)\ns6 = np.zeros(n)\ns7 = np.zeros(
           n)\ns8 = np.zeros(n)\ns9 = np.zeros(n)\ns10 = np.zeros(n)\ns11 = np.
           zeros(n)\ns12 = np.zeros(n)\ns13 = np.zeros(n)\ns14 = np.zeros(n)\
           nfor i in range(n):\n      seed(i)\n      target.calc_probs(time=lc_lin.
           time.value, \n                                flux_0=lc_lin.flux.value, \n
```

```
                       flux_err_0=np.mean(lc_lin.flux_err.value), \n
                   P_orb=P_orb,\n                            parallel=True,\n
                        verbose=0)\n    FPPs[i] = target.FPP\n    s0[i]=
   target.probs["prob"][0]\n    s1[i]=target.probs["prob"][1]\n    s2[i
   ]=target.probs["prob"][2]\n    s3[i]=target.probs["prob"][3]\n    s4[
   i]=target.probs["prob"][4]\n    s5[i]=target.probs["prob"][5]\n    s6
   [i]=target.probs["prob"][6]\n    s7[i]=target.probs["prob"][7]\n
   s8[i]=target.probs["prob"][8]\n    s9[i]=target.probs["prob"][9]\n
    s10[i]=target.probs["prob"][10]\n    s11[i]=target.probs["prob
   "][11]\n    s12[i]=target.probs["prob"][12]\n    s13[i]=target.probs
   ["prob"][13]\n    s14[i]=target.probs["prob"][14]\n\n\nmeanFPP = np.
   mean(FPPs)\nstdvFPP = np.std(FPPs)\nprint("FPP =", meanFPP, "+/-",
   stdvFPP)\n\ntb = Table([s0, s1, s2, s3, s4, s5, s6, s7, s8 ,s9 ,s10,
   s11, s12 ,s13, s14, FPPs], \n           names=\'TP EB EBx2P PTP PEB
   PEBx2P STP SEB SEBx2P DTP DEB DEBx2P BTP BEB BEBx2P FPP\'.split())\n\
   ntb.write(\'TOI-199_0.1h_amp_60days_per_linear.csv\', overwrite=True)
   \n')
142

143
144 # In[ ]:
```

Results plots example, impact parameter

```python
1 # -*- coding: utf-8 -*-
2 """
3 Created on Fri May 10 10:58:47 2024
4
5 @author: warsa
6 """
7
8 import matplotlib.pyplot as plt
9
10 b = [0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0] #impact
     parameter
11 b_offset = [0.02, 0.12, 0.22, 0.32, 0.42, 0.52, 0.62, 0.72, 0.82, 0.92,
     1.02]
12
13 TTV = [1.1102230246251566e-17, 2.2204460492503132e-17, 2.2204460492503132
     e-17, -2.2204460492503132e-17, 5.551115123125783e-17,
     1.398629605464663e-08, 0.08802842217472101, 0.010044176667242355,
     0.08674510374034655, 0.0005548847214168151, 0.12409350481875725] #
     The FPPs for when TTV is corrected for
14 TTV_std = [3.33066907387547e-17, 4.440892098500627e-17, 4.440892098500627
     e-17, 1.0877919644084145e-16, 1.5895974606912447e-16,
     4.1958570604107525e-08, 0.18122829558830136, 0.02278103838381823097,
     0.2578138750606026, 0.0015065431067043252, 0.08780111345305526] # The
      standard deviation for the FPPs when the TTV has been corrected for
15
16 Linear = [3.890152059893292e-06, 0.09863420546049541, 1.3450607250220514e
     -07, 0.15986178876446838, 0.07637873049320434, 0.1044358886163935,
     0.28759819395941255, 0.09721906263524328, 0.3186319180142518,
     0.003634009222574504, 0.49770787380854975] # The FPPs for the the
```

```
     TTV is not corrected for
17 Linear_std = [1.1498507562608107e-05, 0.2959021732905084,
      3.8658756187941456e-07, 0.31154930802132386, 0.17648479453355548,
      0.2968950273219815, 0.4273048365527431 5, 0.28878040931285565,
      0.40353949230795677, 0.005937313921526824, 0.2265726941204414] # The
      standard deviation for the FPPs when the TTV is not corrected for
18
19 plt.scatter(b_offset, TTV, marker="x", color="k", label="Corrected")
20 plt.scatter(b, Linear, marker="o", color="k", label="Not corrected")
21 plt.title("FPP at different impact parameters with \n TTV amplitude of
      0.1h", fontsize=16, weight="bold")
22 plt.ylabel("FPP", fontsize=16)
23 plt.xlabel("Impact parameter", fontsize=16)
24 plt.ylim(-0.05, 1.05)
25 plt.xlim(-0.02, 1.05)
26 plt.errorbar(b, Linear, yerr=Linear_std, fmt="none", ecolor="black",
      capsize=5)
27 plt.errorbar(b_offset, TTV, yerr=TTV_std, fmt="none", ecolor="black",
      capsize=5)
28 plt.axhspan(0.5, 2.0, color="red", zorder=-10, alpha=0.1)
29 plt.axhspan(0.015, 0.5, color="orange", zorder=-10, alpha=0.1)
30 plt.axhspan(-1.0, 0.015, color="lightgreen", zorder=-10,alpha=0.3)
31 plt.tight_layout()
32 plt.legend()
33 plt.annotate("Likely planet", (0.01, 0.42), fontsize=16)
34 plt.annotate("Not a planet", (0.01, 0.7), fontsize=16)
35 plt.tick_params(axis='both', which='major', labelsize=14)
36 plt.savefig("impact parameter 0.1h.pdf")
```