

## VSPEC and Friends: A suite of utilities to model spectroscopic phase curves of 3D exoplanet atmospheres in the presence of stellar variability.

TED JOHNSON,<sup>1,2,3</sup> CAMERON KELAHAN,<sup>1</sup> AVI M. MANDELL,<sup>1</sup> TOM BARCLAY,<sup>1</sup> VESELIN B. KOSTOV,<sup>1</sup> AND GERONIMO L. VILLANUEVA<sup>1</sup>

<sup>1</sup>*NASA Goddard Space Flight Center  
8800 Greenbelt Rd  
Greenbelt, MD 20771, USA*

<sup>2</sup>*Nevada Center for Astrophysics, University of Nevada, Las Vegas, 4505 South Maryland Parkway, Las Vegas, NV 89154, USA*

<sup>3</sup>*Department of Physics and Astronomy, University of Nevada, Las Vegas, 4505 South Maryland Parkway, Las Vegas, NV 89154, USA*

### ABSTRACT

We present the Variable Star Phase Curve (**VSPEC**) code, a python package to simulate combined-light spectroscopic observations of exoplanets in the presence of stellar variability and inhomogeneity. VSPEC uses the Planetary Spectrum Generator’s Global Emission Spectra (PSG GlobES) application along with a custom-built stellar model based on an existing grid of stellar photosphere models to produce spectroscopic light curves of the planet-host system. VSPEC can be a useful tool for modeling observations of exoplanets in transiting geometries (primary transit, secondary eclipse) as well as orbital phase curve measurements, and is built in a modular and flexible configuration for easy adaptability to new stellar and planetary model inputs. We additionally present a set of codes developed alongside VSPEC, including the stellar surface model **vspec-vsm**, the stellar spectral grid interpolation code Gridpolator, and a Python interface for PSG **pypsg**.

*Keywords:* Exoplanet, M dwarf

### 1. INTRODUCTION

In the era of high-sensitivity transiting exoplanet characterization missions such as the Hubble Space Telescope (HST), the James Webb Space Telescope (JWST) and the future Atmospheric Remote-sensing Infrared Exoplanet Large-survey (ARIEL), spectral analysis of exoplanet atmospheres is increasingly sensitive to contamination due to stellar inhomogeneities (e.g. spots, granulation) and stochastic stellar variability (e.g. flares). As an example, recent analysis of the JWST/NIRSpec transit of GJ 486b by Moran et al. (2023) exposed a degeneracy between atmospheric absorption by water and water-rich spots on the stellar surface, and similar effects were seen in the transit spectrum of LHS 1140b Cadieux et al. (2024). This “transit light source effect” (TLS, Rackham et al. 2018, see also Apai et al. (2018); Barclay et al. (2021); Garcia et al. (2022); Barclay et al. (2023)) occurs when the region of the stellar surface occulted by the transiting planet is not representative of the disk-integrated spectrum.

Stellar contamination may also cause errors in the extraction of thermal phase curve spectroscopy if the stellar spectrum changes significantly over the period of

the planet’s orbit; if the variations are non-linear but only linear interpolation between subsequent secondary eclipses is used to remove the stellar contribution, the results planetary flux measurements will be contaminated. This will be particularly important for observations of planets around more variable stars such as active M-stars or later-type evolved host stars.

To adequately prepare for future observations and future missions it will be necessary to demonstrate a method to mitigate these effects. This task requires a flexible tool that combines models of exoplanet atmospheres and stellar variability in a robust way. In this paper we present VSPEC: Variable Star PhasE Curve<sup>1</sup>, an open-source Python 3 package to simulate observations of exoplanet systems with variable host stars. VSPEC itself is merely an interface that pulls together a variable star surface model **vspec-vsm**<sup>2</sup>, a stellar spectra grid interpolation code Gridpolator<sup>3</sup>, and a Python interface

<sup>1</sup> <https://github.com/VSPEC-collab/VSPEC>

<sup>2</sup> <https://github.com/VSPEC-collab/vspec-vsm>

<sup>3</sup> <https://github.com/VSPEC-collab/GridPolator>

for the Planetary Spectrum Generator (PSG, [Villanueva et al. 2018](#)), `pypsg`. All of these codes together allow VSPEC to account for 3D planetary atmospheres, time-resolved effects of both planet and star, many geometries including transit and eclipse, and realistic noise modeling.

**Change this later->** In this paper we will describe the science behind VSPEC and demonstrate examples of its use. Section 4 describes the stellar model and the available sources of variability. Section ?? discusses modeling the planetary atmosphere, including PSG/GlobES and the built-in GCM. In Section 7 we provide examples of VSPEC use cases and demonstrate its value to the exoplanet community. Finally, in Section 8 we will discuss the future of the code and issues it might have.

## 2. USING VSPEC

The flow of VSPEC’s code is broken into three parts: i) read in the configuration, ii) compute planetary spectra, and iii) compute stellar spectra.

### 2.1. Configuring VSPEC

VSPEC configurations are designed to minimize human errors by providing two equivalent formats. The first, a file written in YAML, is optimized for human readability. For example, the `system` section, which describes the relationship between the observed planetary system and the observer, could be written:

```
system:
  distance: 12.4 pc
  inclination: 89.7 deg
  phase_of_pariasteron: 0 deg
```

Note that the units of each of these parameters is included in a human-readable way. Internally, VSPEC casts each of these parameters to an Astropy `Quantity` instance, so that the user can input any desired value and unit combination so long as the physical type is correct – i.e. it would be equivalent to write `3.83e17 m` as the value for the `distance` parameter.

The second input method available is to directly initialize the Python object that VSPEC uses internally to store its model parameters. The top-level object, `InternalParameters`, is structured to mirror the YAML input file, with one argument per YAML section. The parameters in Listing 2.1 would be written:

```
from astropy import units as u
from VSPEC.params import *
params = InternalParameters(
    ... # Other arguments
    system=SystemParameters(
        distance=12.4*u.pc,
```

```
    inclination=89.7*u.deg,
    phase_of_pariasteron=0*u.deg
)
)
```

This is convenient for producing configurations programmatically, for example for producing a grid of model phase curves.

However the user decides to provide model parameters, they are read into the main VSPEC object: the `ObservationModel`. This is the class that runs the model and handles files. Upon initialization a local working directory (by convention `.vspec`) is created to store simulation outputs and intermediate files.

### 2.2. Planetary Phase Curve

VSPEC generates a phase curve through a series of API calls to the Planetary Spectrum Generator (PSG [Villanueva et al. 2018](#)), using `pypsg` (see Section 3) to interface with the API. Initially, configurations are sent that give PSG information about the system that does not change with time – that includes the 3D Global Circulation Model (GCM) and instrument parameters. VSPEC then enters the main loop, iterating through each observation epoch (e.g. integration or combination of integrations) and making a pair of API calls: one to get a combined light spectrum and one to get the thermal flux only. Results from PSG in each epoch are stored locally in the `.fits` format.

#### 2.2.1. Phase Sampling

Making API calls to PSG can be computationally expensive, especially when trying to resolve small structures in the climate model. In order to reduce unnecessary calls to PSG, the temporal sampling of the planetary spectrum is independent of the stellar model/output sampling. When computing the planetary flux in the output files, the raw PSG output is interpolated and averaged over each time step using the trapezoid integration rule. By default one spectrum is computed with PSG at each interface separating time steps; this eases the calculation of edge effects between steps.

### 2.3. Stellar Lightcurve

The last step is to replace the stellar spectra used by PSG with VSPEC’s stellar model. In this step VSPEC acts as a bridge between the stellar surface model `vspec-vsm` (which models spatial changes to the star with time, see Section 4) and a grid of pre-computed stellar spectra (from the GridPolator package; see Section 5). The star is initialized and allowed to evolve for a user-specified amount of time in order for spots and faculae to approach growth-decay equilibrium (this is also called the

“burn in” phase). The star is then evolved at the output cadence, at each epoch generating a set of key-value pairs that describe the surface temperature and coverage fractions of the portion of the stellar disk visible to the observer. These  $T_{\text{eff}}$ s and coverage fractions are fed into GridPolator to produce composite stellar spectra.

### 2.3.1. Reflected light

VSPEC also produces a composite spectrum of the portion of the star visible to the planet to accurately incorporate the variable star into the reflected light spectrum. To compute the total reflected light in the output files, VSPEC first divides the reflected light flux from PSG by the PSG stellar model to obtain the apparent albedo (I/F). This albedo is then multiplied by the composite spectrum (facing the planet) to obtain total reflected light flux.

### 2.3.2. Noise

Alongside the planetary radiance files, PSG returns a file giving the noise profile for each epoch broken down by source. We assume that the total noise is the quadrature sum of its constituent parts and that the **source** column (i.e. photon noise) is the only one affected by replacing one stellar spectrum with the other. This means the photon noise can be calculated:

$$N_{\text{photon}} = N_{\text{photon, PSG}} \sqrt{\frac{F_{*, \text{VSPEC}}}{F_{*, \text{PSG}}}} \quad (1)$$

where  $F_*$  is the stellar flux. The total noise associated with the integration is then

---


$$N_{\text{tot}} = \sqrt{N_{\text{photon}}^2 + N_{\text{detector}}^2 + N_{\text{telescope}}^2 + N_{\text{background}}^2} \quad (2)$$

### 2.3.3. Transit and Eclipse

Users must be careful when using VSPEC to simulate transit and eclipse measurements because of the vast timescale differences between those events and the typical phase variations of a planet – a sparsely sampled phase curve that happens to include one epoch of transit will not adequately sample the lightcurve accurately during a transit or eclipse because the interpolator knows no better than to assume a linear interpolation scheme. It is recommended that users use a high cadence for observations of transits and eclipses so that each event will be sampled by more than one point.

In the case of a transiting geometry, VSPEC computes the spectrum of the occulted portion of the star in order to properly simulate the transit light source effect (TLS [Rackham et al. 2018](#)). This spectrum is the flux blocked by the planet in the case that it is a solid occulting

circle, and therefore produces a flat transmission spectrum. The effect of atmospheric transmission is added by comparing the PSG-computed effective radius to a purely geometric calculation of such a “bare-rock” transit depth:

$$F_{\lambda} = F_{\text{rock}, \lambda} F_{\text{PSG}, \lambda} \left( \frac{R_*}{R_p} \right)^2 \quad (3)$$

where  $F_{\lambda}$  is the flux of the variable star surface occulted by the planet with the atmosphere considered.  $F_{\text{rock}, \lambda}$  is the flux of the surface occulted by a bare rock with the planet’s radius,  $F_{\text{PSG}, \lambda}$  is the flux computed by PSG to be lost due to occultation and transmission, and  $R_*$  and  $R_p$  are the stellar and planetary radii, respectively.

In the case of a total eclipse of the planet, the planetary thermal and reflected contributions are set to zero flux. However, in the case of a partial eclipse, VSPEC computes the fraction of the planet that is visible to the observer and reduces both thermal and reflected flux accordingly. In this case VSPEC treats the planet homogeneously; however, future work could utilize PSG/GlobES’s ability to return a hypercube of spatially resolved spectra in order to construct planetary spectra only from regions of the planet that are not eclipsed.

## 3. INTERFACING WITH THE PLANETARY SPECTRUM GENERATOR VIA PYPSG

PSG<sup>4</sup> ([Villanueva et al. 2018](#)) is a powerful radiative transfer tool that is ubiquitous in the exoplanet and solar system atmosphere fields. Either through its web graphical interface, its public API, or the local version available through Docker, PSG allows users to simulate observations of planets, comets, and moons with a variety of geometries and realistic noise models **probably someone like Geronimo should write this part**.

VSPEC uses a Python interface for the PSG API system called **pypsg**, which is a general stand-alone library built for any Python-based API call to PSG. **pypsg** draws inspiration from object-relational mapping in frameworks such as Django [cite](#) to encode a PSG configuration file as a native Python object. Fields of a **pypsg** data model represent one or more lines of a configuration file, and the two representations are interchangeable without loss of information. Upon creation of this configuration as a **PyConfig** object, a user can create an **APICall** instance which handles sending a request to PSG via Python’s **requests** [cite](#) package. This dedicated caller is important in that it reads a user’s API key from a known location on disk, removing the possibility that the key could be committed to a public repository.

<sup>4</sup> <https://psg.gsfc.nasa.gov/>

A powerful feature of `pypsg` is that it provides a universal interface for 3D atmosphere models and PSG’s Global Emission Spectra (GlobES) application. GlobES expects 3D data to be presented in a binary data format, and converting a model to this format is non-trivial. `pypsg`’s PyGCM class provides a native Python representation of GlobES input, and includes built-in methods for converting popular climate models (e.g. ExoCAM, WACCM [full names and ref?](#)) to the correct format.

#### 4. STELLAR VARIABILITY MODEL

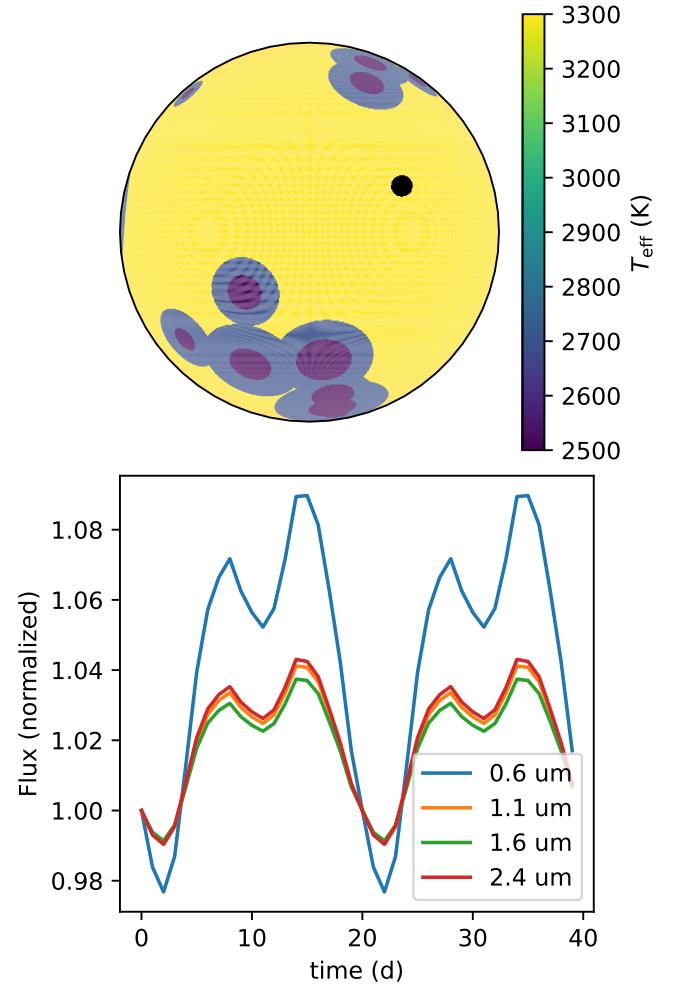
The VSPEC stellar model is designed [modularly in a modular fashion](#) to allow for both simple and complex behaviors. The model is available as the Python package `vspec-vsm`(VSPEC Variable Star Model). It is spectral-model agnostic, meaning that it relies on another package (GridPolator, see Section 5) to compute the actual variable spectra. Instead, the surface model takes a pair of latitude and longitude points as input and returns a dictionary  $\{T_{\text{eff}} : f_{T_{\text{eff}}}\}$  where the keys  $T_{\text{eff}}$  are the effective temperatures of each surface component and the values  $f_{T_{\text{eff}}}$  are the fraction of the visible stellar disk with that effective temperature.

The main model class is the `Star` object, which describes bulk properties [like radius, nominal \(e.g. radius, photospheric  \$T\_{\text{eff}}\$ , and limb darkening parameters, \)](#) in addition to acting as a container for objects representing sources of variability. [Additional classes are defined for additional stellar components, e.g. spots, faculae,..., which are discussed below.](#)

##### 4.1. Stellar Surface Grid

Sources of variability that exist as major structures on the stellar surface (i.e. spots and faculae) are encoded as deviations from the [nominal photospheric](#) effective temperature on a grid of points representing the stellar surface. Each time the fractions of each surface component are calculated, the grid is initialized with constant  $T_{\text{eff}}$ . The code then iterates through each spot and facula, replacing portions of the grid with the effective temperatures that make up each structure.

The simplest grid to build is one that is rectangular in latitude and longitude, and in which the area that each point represents decreases towards the poles. However, this can lead to unnecessary spatial oversampling at the poles, and may lead to poor resolution at the equator to keep the total number of points from being excessively high. To mitigate these problems `vspec-vsm` supports a “grid” generated using a Fibonacci spiral to create a near-isotropic distribution of points. This is especially useful when simulating transits, giving optimal sampling at the stellar equator. It is also useful for resolving small



**Figure 1.** **Top:** The surface map of a star with spots during a transit. The large ellipses are spots, while the transiting planet occults a portion of the disk represented by the black circle. **Bottom:** An example lightcurve of a spotted star. [Remove transit and put it in a LD figure](#)



structures on the surface, as the number of points needed for a given minimum resolution [are is](#) greatly reduced.

##### 4.2. Spots

Our star spot model is nearly entirely based on observations of the sun. Sunspots Sun. Sunspots can be resolved and are well-studied, whereas spots on other stars (e.g., M dwarfs especially non-solar-type stars) can only be observed indirectly. We therefore designed our spot model to mimic sun spots the behavior of sunspots but with parameterized values for spot temperature and lifetime that can be matched to observations of other stellar types and ages.

On the Sun, spots have two regions shown if in Figure 1: the dark, central umbra and the lighter, surrounding

penumbra . A (a detailed review of sunspot behavior, including sizes and lifetimes, is Solanki (2003), however the specific solar values are not relevant to this description. However, the described in Solanki (2003)). The spot area as a function of time is

$$A(t) = \begin{cases} A_0 e^{(t-t_0)/\tau}, & \text{if } t \leq t_0 \\ A_0 - W(t-t_0), & \text{if } t > t_0 \end{cases} \quad (4)$$

where  $A_0$  is the maximum area reached,  $t_0$  is the time of the maximum,  $\tau$  is the exponential growth rate, and  $W$  is the linear decay rate.

In addition to modeling the variability produced by a population of spots, `vspec-vsm` contains utilities to produce those populations given some description. These parameters include the average spot area (lognormally distributed, Bogdan et al. 1988),  $T_{\text{eff}}$ , growth and decay rates, and the method for distributing spots on the surface. These `SpotGenerator` objects are responsible for evolving the spot population as the star ages. The number of spots created in some interval  $\Delta t$  is:

$$N(\Delta t) = \frac{4\pi R_*^2 f_{\text{spot}}}{A_{\text{mean}}} \frac{\Delta t}{T} \quad (5)$$

where  $R_*$  is the stellar radius,  $f_{\text{spot}}$  is the fraction of the surface covered by spots in growth-decay equilibrium,  $A_{\text{mean}}$  is the lifetime-averaged mean spot area, and  $T$  is the spot lifetime, defined as:

$$T = \frac{A_0}{W} - \frac{1}{\tau} \ln \left( \frac{A_{\text{min}}}{A_0} \right) \quad (6)$$

where  $A_{\text{min}}$  is the area given to each spot when it is initialized.

There are also two modes to distribute spots on the star's surface. The first distributes them isotropically. The ; the second, modeled after analysis of spots on the Sun by Mandal et al. (2017), concentrates the spot latitude around  $\pm 15^\circ$ .

### 4.3. Faculae

Faculae are magnetically-generated regions of the solar surface that usually appear as bright points near the limb; we employ the “hot wall” model (Spruit 1976) where faculae are described as three-dimensional pores in the stellar surface with a hot, bright wall , and a cool, dark floor, as shown in Figure 2.

Their three-dimensional structure causes faculae’s faculae observational properties to change depending on their angle from disk-center. Close to the limb, the hot wall is visible to the observer, and faculae appear as bright points. Near ; near the center, however, the cool floor is exposed and faculae appear dark. To consider

this effect in the faculae lightcurve, we compute the fraction of the each facula’s normalized projected area – the area on the disk it would occupy as a flat spot seen by the observer – that is occupied by each the hot wall and versus the cool floor. This is done via numerical integral along the radius of the spot.

We calculate each portion using geometrical relationships between the spot dimensions and the viewing angle:

$$f_{\text{wall}} = \frac{\int_{-R}^R Z_{\text{eff}} dr}{\int_{-R}^R Z_{\text{eff}} dr + \int_{-R}^R R_{\text{eff}} dr} \quad (7)$$

where  $Z_{\text{eff}}$  is the apparent height of the wall

$$Z_{\text{eff}} = \begin{cases} Z_w \sin \alpha, & \text{if } \alpha \leq \alpha_{\text{crit}} \\ 2\sqrt{R^2 - r^2} \cos \alpha, & \text{if } \alpha > \alpha_{\text{crit}} \end{cases} \quad (8)$$

and  $R_{\text{eff}}$  is the apparent width of the floor

$$R_{\text{eff}} = \begin{cases} 2\sqrt{R^2 - r^2} - Z_w \sin \alpha, & \text{if } \alpha \leq \alpha_{\text{crit}} \\ 0, & \text{if } \alpha > \alpha_{\text{crit}} \end{cases} \quad (9)$$

for facula radius  $R$ , depth  $Z_w$ , and angle from disk-center  $\alpha$ .  $r$  in this numerical scheme is defined as the distance from the center of the facula along the radial line connecting the facula center to the disk center.  $\alpha_{\text{crit}}$  is the value of alpha at which the floor is no longer visible and is defined to be  $\arctan \frac{2\sqrt{R^2 - r^2}}{Z_w}$ .

According to Topka et al. (1997)studies of solar faculae (Topka et al. 1997), faculae temperatures (of both the floor and wall) are dependent on the facula radius, while depth appears to be constant. They also find that the smallest faculae have no visible floor, and that even at disk center they appear as bright points. We parameterize the floor temperature to be

$$\Delta T_{\text{eff,floor}} = \begin{cases} \text{Not visible,} & \text{if } r < r_{\text{min}} \\ m_{\text{floor}}(r - r_{\text{min}}) + \Delta T_{\text{eff,floor,0}}, & \text{if } r \geq r_{\text{min}} \end{cases} \quad (10)$$

Where Change variable names, make diagram where  $r$  is the radius of the facula,  $r_{\text{min}}$  is the minimum radius where the floor is visible,  $\Delta T_{\text{eff,floor,0}}$  is the difference between the floor and photosphere  $T_{\text{eff}}$  at  $r_{\text{min}}$ , and  $m_{\text{floor}}$  is the slope of the relationship with units of [temperature] [length] $^{-1}$ .

Similarly, the wall temperature is parameterized as

$$\Delta T_{\text{eff,wall}} = m_{\text{wall}}r + \Delta T_{\text{eff,wall,0}} \quad (11)$$

Where where  $\Delta T_{\text{eff,wall,0}}$  is the temperature of a zero-radius facula and and  $m_{\text{wall}}$  is the slope of the radius-temperature relationship with units of [temperature] [length] $^{-1}$ . These relationships can be defined by the

user, for example setting  $m = 0$  for constant temperatures.

Facula lifetimes are defined as the time it takes its radius to decay by  $e^{-2}$ . Because faculae grow and decay exponentially at the same rate, each facula spends one lifetime with a radius greater than  $e^{-1}$  of its maximum. Each facula is born and dies at a radius of  $e^{-2}$  of its maximum, effectively existing in the code for two lifetimes.

Hovis-Afflerbach & Pesnell (2022) suggest a typical facula lifetime to be on the order of 6 hours, with a distribution that resembles a Poisson function. However, we choose a lognormal distribution for both lifetime and maximum radius because it does not allow for these values to be 0. We choose to correlate lifetime and radius so that they are determined by the normally distributed random variable  $\mu$ ; a normalized random distribution - for each new facula,  $\mu$  a value is randomly drawn and determines both the facula lifetime and maximum radius.

#### 4.4. Flares

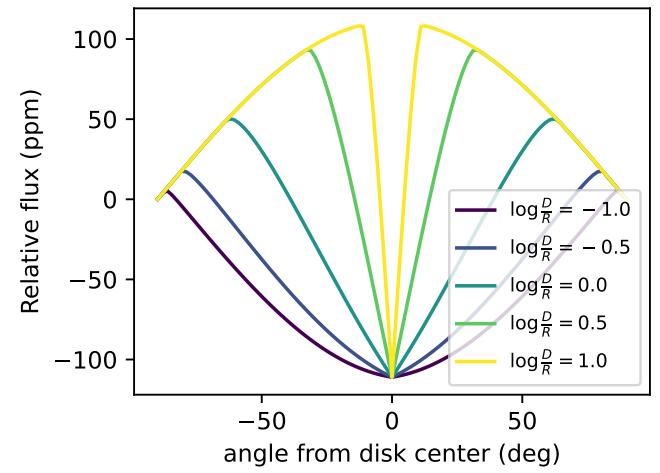
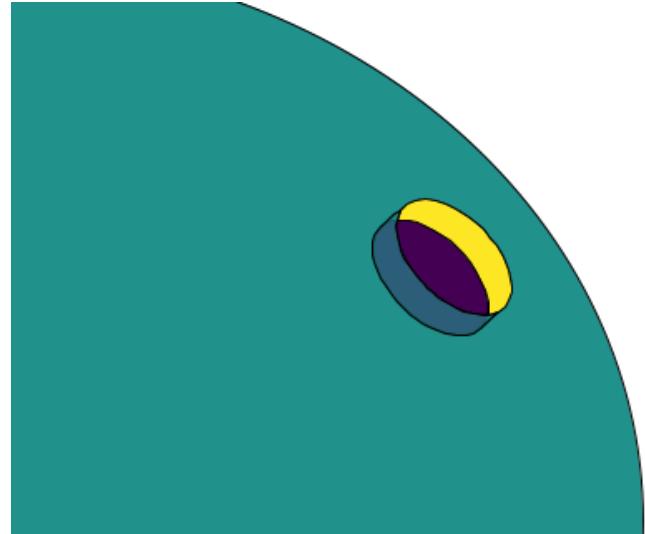
Flares are an important source of stellar variability on short timescales. vspec-vsm’s flare lightcurve model is based on the xoflares package (Barclay & Emily-gilbert 2020), which itself is based on empirically derived lightcurve shapes from Davenport (2016). However, xoflares is designed to fit lightcurves in a single spectral band, so we add additional parameters to extend it to a multiwavelength lightcurve. We completely describe a flare using its temperature, total energy, full width at half maximum (FWHM), and the time of its peak. We model a flare as a hot, optically thin region above the photosphere that produces a blackbody spectrum. In this model, the temperature is constant, and the sharp rise and fall seen in the lightcurve is caused by the region’s rapidly changing area. This simplification allows the total flare energy  $E$  to be used as a normalization factor:

$$\int_{-\infty}^{\infty} A dt = \frac{E}{\sigma T^4} \quad (12)$$

where  $A$  is the time-dependent area of the flare region,  $T$  is the constant flare temperature, and  $\sigma$  is the Stefan-Boltzmann constant. This allows flare lightcurves to be produced with a fixed bolometric luminosity.

Because the temperature is fixed, the relevant quantity for modeling the flux of a flare in a given integration is the integrated time-area  $\int A dt$ . These quantities are computed numerically and fed to VSPEC along with  $T$  to produce spectra with the appropriate absolute flux.

Before simulating an observation, VSPEC asks vspec-vsm to pre-compute a population of flares based

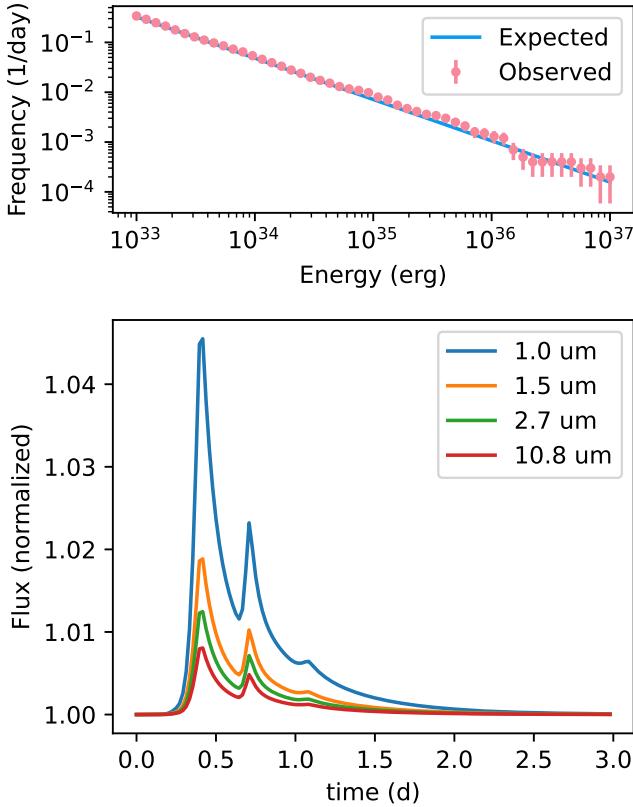


**Figure 2.** “Hot Wall” model of faculae. Faculae structure causes their contrast to be dependent on their distance from the center of the disk. **Top:** Depiction of a facula on the limb of a star. The hot wall is exposed to the observer causing the pore to appear bright. At disk center, the cool floor is most visible. **Bottom:** The effects of depression depth and viewing angle on facula brightness. The 3D structure of faculae is most apparent when radius  $\sim$  depth. A toy flux model was used to demonstrate the shape of these curves, but their magnitudes in practice their magnitudes depend on stellar spectral models.

on the a power-law frequency-energy relationship from Gao et al. (2022) similar to the results of what data? (Gao et al. 2022):

$$\log(f/\text{[day]}) = \beta + \alpha \log(E/\text{[erg]}) \quad (13)$$

Where where  $f$  is the frequency of flares with energies  $\geq E$ . We compute the number of expected flares over a time duration and determine the number to create  $N$



**Figure 3.** **Top:** Generated flare frequencies compared to those expected from Gao et al. (2022), generated using a 10,000 day simulation. The  $y$ -axis shows the frequency of flares with energies greater than or equal to the value of the  $x$ -axis. Error bars based on the square root of the number of observed flares as this is a Poisson process. **Bottom:** Lightcurve of a star flaring with the same powerlaw slope as the top panel, but the intercept ( $\beta$ ) has been increased by 0.3 for visual effect. The mean flare temperature is 9000 K and the mean FWHM is 3 hours.

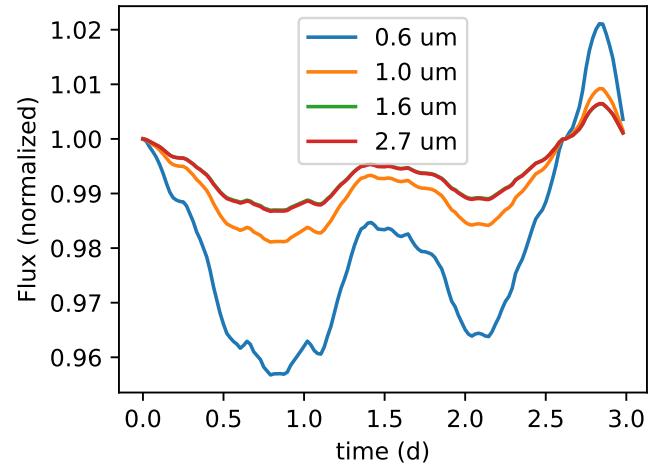
by a random Poisson draw. We then generate  $N$  flares with energies determined by the quantile function:

$$E = E_{\min}(1 - X)^{1/\alpha} \quad (14)$$

where  $E_{\min}$  is the minimum considered flare energy and  $X$  is a random number on the interval  $[0, 1]$ . We set the default values  $\alpha = -0.829$ ,  $\beta = 26.87$  from Gao et al. (2022), but these can be adjusted by the user. Figure 3 shows our simulation results compared to their powerlaw values.

#### 4.5. Granulation

Granulation is a source of stellar variability that arises from convection near the surface of the star. The result is a stellar surface that is not constant in temperature and that changes on very short timescales. Hydrodynamic simulations of stellar atmospheres (e.g. Magic &



**Figure 4.** Effect of granulation on the lightcurve of a star. This simulation assumes a mean 10% coverage by an inter-granule region whose temperature is 300 K 300K lower than the surrounding photosphere. It also assumes that the coverage value varies with a magnitude of 1% on a timescale of 6 hours find some constraints in Gordon paper.

Asplund 2014) describe hot granules of rising gas surrounded by cooler, sinking regions with a temperature a few percent lower than the nominal value.

We model granulation as a global process, and its effects are computed after the effects of spots and faculae. Of the remaining “quiet” photosphere (i.e. the regions not covered in spots or faculae), a fraction is computed to be part of the cool inter-granule surface. The ; the surface coverage of the cool region at any given time is computed by a Gaussian process (GP) using the TINYGP package (Foreman-Mackey et al. 2024) following the methodology of Gordon et al. (2020). The GP uses a custom kernel function based on a power spectrum (Anderson et al. 1990; Kallinger et al. 2014) to produce random changes in the granulation coverage.

#### 4.6. Transits

Change section to talk about limb darkening and other geometric considerations In the case of a transiting geometry, `vspec-vsm` is responsible for computing the properties of the portion of the stellar disk that is occulted. It does this by first projecting both the transiting planet and the visible part of the surface orthographically as they would be viewed by a distant observer. It then selects all the points that are near enough to the transit to be occulted. The code then iterates through each of those points and computes the fraction of it covered by the occulting circle using a 2D numerical integral. Add a figure to show limb darkening lightcurve. Do a surface

map like Fig 1 with a transit, then zoom in to show pixel coverage, then a lightcurve

## 5. STELLAR SPECTRAL GRIDS

**TODO**

## 6. SIMPLE CLIMATE MODELS SIMPLIFIED CLIMATE MODEL

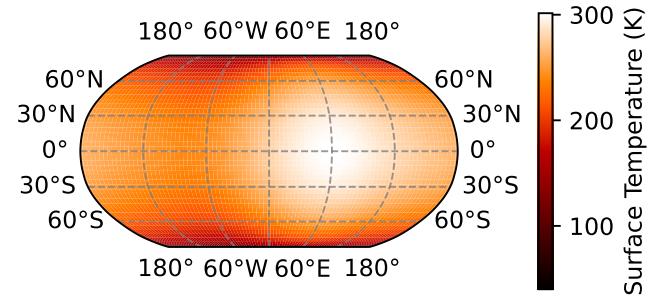
VSPEC allows the user to upload any desired GCM. However, for questions that focus on detectability, the specific physics of the GCM are less important than properties like day/night temperature, surface pressure, or the major absorbers in the atmosphere. To simplify these studies, VSPEC has a built-in GCM that takes in just a few parameters and produces a simple-yet-robust exoplanet climate. Simplicity allows the user to know everything about the model and reduce surprises, but its basis in thermodynamics and robust energy balance make this model fit for publication-level science **is this really true? probably not.**

The VSPEC GCM centers around a 2D temperature map of the planet's surface. Based on Cowan & Agol (2011), this map is constructed completely given the incident flux from the star (parameterized by the host  $T_{\text{eff}}$  and radius and the planet's semimajor axis and Bond albedo) and the unitless thermal redistribution efficiency  $\epsilon$ . This efficiency is 0 for a planet that reradiates instantly and  $\gg 1$  for an atmosphere that efficiently redistributes heat to its night side. Figure 5 shows a temperature map example for an Earth-like planet with  $\epsilon = 2\pi$ . These temperature maps are validated by the energy balance of the planet; VSPEC computes the incident and emergent flux across the planet and asserts that they agree.

Now that we have the surface temperature at each GCM coordinate we can build the atmosphere on top of it. We choose a surface pressure for our model and generate a pressure profile as a function of layer. It is important to note that the choice of the pressure profile – provided it is scaled to the surface pressure and sampled in a reasonable manner – is arbitrary and will not affect the spectrum of the planet. This is because PSG calculates the altitude and thickness of each layer based on the pressure and surface gravity; the pressure profile is essentially a proxy for the sampling of atmosphere layers as a function of altitude.

Next we create an adiabatic temperature profile at each GCM coordinate. An adiabat assumes that

$$T^\gamma P^{1-\gamma} = \text{const.} \quad (15)$$



**Figure 5.** Example of a surface temperature map created by VSPEC. The solar flux at Earth, a Bond albedo of 0.3, and  $\epsilon = 2\pi$  were used in this case. Notice that the hottest point is offset from the sub-stellar point due to thermal inertia.

where  $\gamma$  is the adiabatic index. Therefore the temperature profile can be computed

$$T = T_{\text{surf}} \left( \frac{P}{P_{\text{surf}}} \right)^{1-\frac{1}{\gamma}} \quad (16)$$

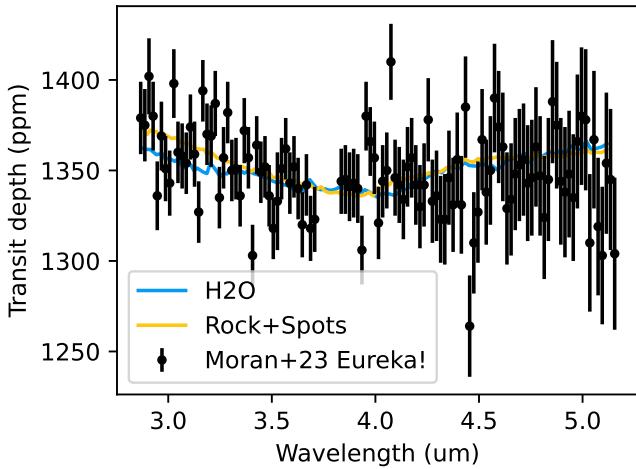
Finally, we populate our atmosphere with molecules. The user specifies the species and abundance and atmosphere is filled accordingly with a constant molecular suite.

The caveats of this approach lie in the simplified physics of the atmosphere structure. The heat-redistribution model does not capture the effects of latitudinal mixing; an adiabatic temperature profile will never exhibit the temperature inversion that we know form in complex atmospheres like Earth's. Similarly, setting a constant abundance does not allow for water vapor to vary with temperature as it does on Earth. We must also be careful not to add so much of an absorbing species that the energy balance of the atmosphere would change; instead we treat them as trace species in an otherwise transparent atmosphere. Our simplified approach allows us to easily examine a large parameter space of atmospheres whose behaviors are easy to predict.

## 7. EXAMPLES

### 7.1. Transit of a spotted star

In this section we will simulate the transit of a bare rock across a spotted star to demonstrate how an inhomogeneous stellar surface can lead to a false atmospheric signal – a phenomenon known as the “transit light source effect” (Rackham et al. 2018). During transit, the planetary atmosphere is illuminated by the portion of the



**Figure 6.** The results from our spotted transit experiment. We find that the addition of spots to a star’s surface can give an otherwise flat transit spectrum features that appear to be molecular bands. This is in agreement with the results of Moran et al. (2023).

star directly behind the planet. The depth of the transit should be measured with respect for the spectrum of this region of the stellar surface. However, the disk-integrated stellar spectrum is often used instead. In the case that the stellar surface is not homogeneous, the transit signal is contaminated by the star. Moran et al. (2023) first observed this effect using JWST from super-Earth GJ 486b.

Whenever possible, we draw parameter values from Moran et al. (2023) and the NExSci Archive<sup>5</sup>. We use the JWST NIRSpec/G395H instrument setup but set  $R = 200$  to account for the resolution of the reduction. Additionally we observe for 3.53 hours with an 8 minute cadence. Mid-transit is reached exactly halfway through the observation.

We will do three VSPEC simulations:

1. Bare rock, no spots.
2. 1 bar H<sub>2</sub>O atmosphere, no spots.
3. Bare rock, 2.5% coverage by 2700 K spots

We expect in case 1 to observe a flat spectrum, and in case 2 to observe additional absorption from the H<sub>2</sub>O atmosphere. Note also that we disabled H<sub>2</sub>O-H<sub>2</sub>O collision-induced absorption (CIA) for these simulations because the effect was not considered in the original analysis of GJ 486b.

We find that, as expected, the addition of spots changes the transit depth in a way that depends on wavelength – mimicking light lost to absorbers in a planetary atmosphere. Figure 6 shows us that, in this case, we observe false absorption; this is due to an increase in the spotted fraction of the surface relative to the quiet photosphere. However, if a spot was blocked by the planet we might see similar features inverted to mimic emission.

## 7.2. Phase Curve

Analysis of JWST MIRI-LRS phase curves of GJ 1214b by Kempton et al. (2023) did not consider stellar contamination because of the slow rotation rate of the host star (approximately 1/80<sup>th</sup> the orbital frequency, Cloutier et al. 2021). In this example we use VSPEC to demonstrate that this is a reasonable assumption.

We initialize VSPEC to run a 41.0 hour observation with a cadence of 15 minutes. We use stellar and planetary parameters from the NExSci Exoplanet Archive<sup>6</sup>. The GCM used has a thermal inertia of  $\epsilon = 6$  and a 1 bar CO<sub>2</sub> atmosphere.

We initialized two stellar models: one with nothing more than a 3250 K surface, and another with 20% coverage by 2700 K spots.

After running the VSPEC simulations, we bin into 0.5  $\mu\text{m}$  wavelength channels and normalize each channel to the two eclipses. We assume the star behaves linearly between eclipse measurements and attribute any deviation to the planet.

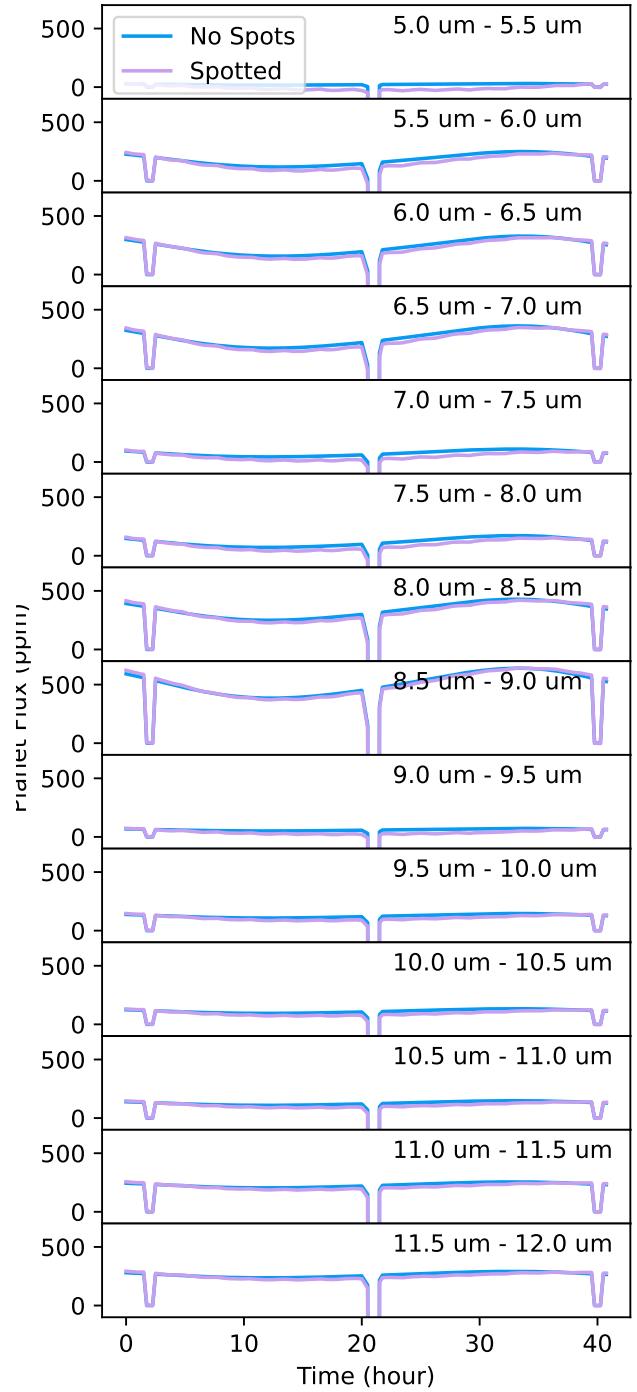
We show in Figure 7 that the stellar properties assumed in this example do not generate enough contamination to significantly change the observed phase curve of GJ 1214b. At it’s worst (i.e. near transit) the observed spectrum varies by 50 ppm due to stellar contamination – compared to the 700 ppm relative flux of the planet at 8.5  $\mu\text{m}$ . Similar studies could be done to assess the potential for stellar contamination in other exoplanetary systems in order to judge fitness as a target.

## 7.3. MIRECLE Phase Curve

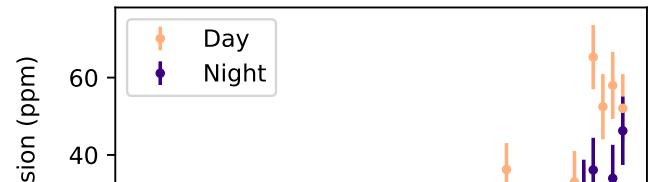
In this example we will look at the best-case target for the Mid-IR Exoplanet CLimate Explorer (MIRECLE, Mandell et al. 2022) mission concept: Proxima Centauri b with no stellar variability. We will instead examine the expected noise produced by this observation and assess the detectability of the planet in a single orbit. VSPEC uses the noise models built into PSG, allowing native

<sup>5</sup> <https://exoplanetarchive.ipac.caltech.edu/overview/GJ1214b>

<sup>6</sup> <https://exoplanetarchive.ipac.caltech.edu/overview/GJ1214b>



**Figure 7.** MIRI phase curves for GJ 1214b binned by



## REFERENCES

- Anderson, E. R., Duvall, Jr., T. L., & Jefferies, S. M. 1990, *The Astrophysical Journal*, 364, 699, doi: [10.1086/169452](https://doi.org/10.1086/169452)
- Apai, D., Rackham, B. V., Giampapa, M. S., et al. 2018, Understanding Stellar Contamination in Exoplanet Transmission Spectra as an Essential Step in Small Planet Characterization, doi: [10.48550/arXiv.1803.08708](https://doi.org/10.48550/arXiv.1803.08708)
- Barclay, T., & Emilygilbert. 2020, Zenodo, doi: [10.5281/zenodo.4156285](https://doi.org/10.5281/zenodo.4156285)
- Barclay, T., Kostov, V. B., Colón, K. D., et al. 2021, *The Astronomical Journal*, 162, 300, doi: [10.3847/1538-3881/ac2824](https://doi.org/10.3847/1538-3881/ac2824)
- Barclay, T., Sheppard, K. B., Latouf, N., et al. 2023, The Transmission Spectrum of the Potentially Rocky Planet L 98-59 c, doi: [10.48550/arXiv.2301.10866](https://doi.org/10.48550/arXiv.2301.10866)
- Bogdan, T. J., Gilman, P. A., Lerche, I., & Howard, R. 1988, *The Astrophysical Journal*, 327, 451, doi: [10.1086/166206](https://doi.org/10.1086/166206)
- Cadieux, C., Doyon, R., MacDonald, R. J., et al. 2024, *The Astrophysical Journal*, 970, L2, doi: [10.3847/2041-8213/ad5afa](https://doi.org/10.3847/2041-8213/ad5afa)
- Cloutier, R., Charbonneau, D., Deming, D., Bonfils, X., & Astudillo-Defru, N. 2021, *The Astronomical Journal*, 162, 174, doi: [10.3847/1538-3881/ac1584](https://doi.org/10.3847/1538-3881/ac1584)
- Cowan, N. B., & Agol, E. 2011, *The Astrophysical Journal*, 726, 82, doi: [10.1088/0004-637X/726/2/82](https://doi.org/10.1088/0004-637X/726/2/82)
- Davenport, J. R. A. 2016, *The Astrophysical Journal*, 829, 23, doi: [10.3847/0004-637X/829/1/23](https://doi.org/10.3847/0004-637X/829/1/23)
- Foreman-Mackey, D., Yu, W., Yadav, S., et al. 2024, Dfm/Tinygp: The Tiniest of Gaussian Process Libraries, Zenodo, doi: [10.5281/ZENODO.6389737](https://doi.org/10.5281/ZENODO.6389737)
- Gao, D.-Y., Liu, H.-G., Yang, M., & Zhou, J.-L. 2022, *The Astronomical Journal*, 164, 213, doi: [10.3847/1538-3881/ac937e](https://doi.org/10.3847/1538-3881/ac937e)
- Garcia, L. J., Moran, S. E., Rackham, B. V., et al. 2022, *Astronomy and Astrophysics*, 665, A19, doi: [10.1051/0004-6361/202142603](https://doi.org/10.1051/0004-6361/202142603)
- Kallinger, T., De Ridder, J., Hekker, S., et al. 2014, *Astronomy and Astrophysics*, 570, A41, doi: [10.1051/0004-6361/201424313](https://doi.org/10.1051/0004-6361/201424313)
- Gordon, T. A., Agol, E., & Foreman-Mackey, D. 2020, *The Astronomical Journal*, 160, 240, doi: [10.3847/1538-3881/abbc16](https://doi.org/10.3847/1538-3881/abbc16)
- Hovis-Afflerbach, B., & Pesnell, W. D. 2022, *Solar Physics*, 297, 48, doi: [10.1007/s11207-022-01977-8](https://doi.org/10.1007/s11207-022-01977-8)
- Kempton, E. M. R., Zhang, M., Bean, J. L., et al. 2023, *Nature*, 620, 67, doi: [10.1038/s41586-023-06159-5](https://doi.org/10.1038/s41586-023-06159-5)
- Magic, Z., & Asplund, M. 2014, The Stagger-grid: A Grid of 3D Stellar Atmosphere Models - VI. Surface Appearance of Stellar Granulation, arXiv. <https://arxiv.org/abs/1405.7628>
- Mandal, S., Karak, B. B., & Banerjee, D. 2017, *The Astrophysical Journal*, 851, 70, doi: [10.3847/1538-4357/aa97dc](https://doi.org/10.3847/1538-4357/aa97dc)
- Mandell, A. M., Lustig-Yaeger, J., Stevenson, K. B., & Staguhn, J. 2022, *The Astronomical Journal*, 164, 176, doi: [10.3847/1538-3881/ac83a5](https://doi.org/10.3847/1538-3881/ac83a5)
- Moran, S. E., Stevenson, K. B., Sing, D. K., et al. 2023, *The Astrophysical Journal*, 948, L11, doi: [10.3847/2041-8213/accb9c](https://doi.org/10.3847/2041-8213/accb9c)
- Rackham, B. V., Apai, D., & Giampapa, M. S. 2018, *The Astrophysical Journal*, 853, 122, doi: [10.3847/1538-4357/aaa08c](https://doi.org/10.3847/1538-4357/aaa08c)
- Solanki, S. K. 2003, *Astronomy and Astrophysics Review*, 11, 153, doi: [10.1007/s00159-003-0018-4](https://doi.org/10.1007/s00159-003-0018-4)
- Spruit, H. C. 1976, *Solar Physics*, 50, 269, doi: [10.1007/BF00155292](https://doi.org/10.1007/BF00155292)
- Topka, K. P., Tarbell, T. D., & Title, A. M. 1997, *The Astrophysical Journal*, 484, 479, doi: [10.1086/304295](https://doi.org/10.1086/304295)
- Villanueva, G. L., Smith, M. D., Protopapa, S., Faggi, S., & Mandell, A. M. 2018, *Journal of Quantitative Spectroscopy and Radiative Transfer*, 217, 86, doi: [10.1016/j.jqsrt.2018.05.023](https://doi.org/10.1016/j.jqsrt.2018.05.023)

<sup>7</sup> <https://exoplanetarchive.ipac.caltech.edu/overview/proxima%20cen>