





# How Do We Optimally Sample Model Grids of Exoplanet Spectra?

Chloe Fisher<sup>1,2</sup>  and Kevin Heng<sup>1,3,4</sup> <sup>1</sup> University of Bern, Center for Space and Habitability, Sidlerstrasse 5, CH-3012, Bern, Switzerland<sup>2</sup> University of Oxford, Department of Physics, Denys Wilkinson Building, Oxford OX1 3RH, UK<sup>3</sup> University of Warwick, Department of Physics, Astronomy and Astrophysics Group, Coventry CV4 7AL, UK<sup>4</sup> Ludwig Maximilian University, University Observatory Munich, Scheinerstrasse 1, Munich D-81679, Germany

Received 2021 November 8; revised 2022 May 13; accepted 2022 June 11; published 2022 July 22

## Abstract

The construction and implementation of atmospheric model grids is a popular tool in exoplanet characterization. These typically vary a number of parameters linearly, containing one model for every combination of parameter values. Here we investigate alternative methods of sampling parameters, including random sampling and Latin hypercube (LH) sampling, and how these compare to linearly sampled grids. We use a random forest to analyze the performance of these grids for two different models, as well as investigate the information content of the particular model grid from Goyal et al. (2019). We also use nested sampling to implement mock atmospheric retrievals on simulated James Webb Space Telescope transmission spectra by interpolating on linearly sampled model grids. Our results show that random or LH sampling outperforms linear sampling in parameter predictability for our higher-dimensional models, requiring fewer models in the grid, and thus allowing for more computationally intensive forward models to be used. We also found that using a traditional retrieval with interpolation on a linear grid can produce biased posterior distributions, especially for parameters with nonlinear effects on the spectrum. In particular, we advise caution when performing linear interpolation on the C/O ratio, cloud properties, and metallicity. Finally, we found that the information content analysis of the grid from Goyal et al. (2019) was able to highlight key areas of the spectra where the presence or absence of certain molecules can be detected, providing good indicators for parameters such as temperature and C/O ratio.

*Unified Astronomy Thesaurus concepts:* [Exoplanets \(498\)](#); [Exoplanet atmospheres \(487\)](#)


## 1. Introduction

With the recent launch of the James Webb Space Telescope (JWST) and the upcoming developments in ground-based observatories, we are stepping into a new era of exoplanet data. These new facilities promise an explosion in the precision and sensitivity of spectra of exoplanet atmospheres, which will require a matching advancement in our analysis techniques. The current state of the art in exoplanet analysis uses atmospheric retrieval to search parameter space for the model that best fits the data (e.g., Madhusudhan & Seager 2009; Benneke & Seager 2013). These traditionally employ a Bayesian sampling algorithm such as an MCMC or nested sampling, in conjunction with an atmospheric model. In a single retrieval, tens of thousands of models are computed on the fly and compared to the data, meaning we are inherently limited in the complexity of the physical models we can use. Most retrievals rely on 1D atmospheric models, with some recent work branching out into forms of 2D models (e.g., Irwin et al. 2020). However, many studies have investigated potential biases in the results from 1D retrievals, and the detrimental effects these can have when attempting to accurately characterize an exoplanet (e.g., Feng et al. 2016; Line & Parmentier 2016; Taylor et al. 2020). With improved data from upcoming instruments such as JWST and the Extremely Large Telescope (ELT), these biases will only worsen.

An alternative method of exoplanet analysis involves the computation of grids of atmospheric models, constructed by varying each parameter in turn. These grids have fewer

computational restrictions, allowing for more complex physics to be included in the model. The linear structure of the grids enables one to study the individual effects of each parameter on the spectrum and assess the sensitivity of observations. The grids can also be used to exclude particular models when analyzing data. For example, de Wit et al. (2018) are able to rule out hydrogen-dominated atmospheres for the Trappist-1 planets simply by visual inspection. In more recent years, several groups have developed techniques that use machine learning to perform atmospheric retrieval and other analyses by training on a set of synthetic spectra (e.g., Waldmann 2016; Márquez-Neila et al. 2018; Zingales & Waldmann 2018; Cobb et al. 2019; Fisher et al. 2020; Ardevol Martinez et al. 2022; Matchev et al. 2022). This form of retrieval allows one to use model grids provided by other groups, without requiring access to the original model code or relying on an interpolation method. Most model grids are either open-source or can be provided on demand, and range from brown dwarf spectra (Burrows et al. 1997; Allard et al. 2001; Allard 2014; Marley et al. 2021) to global circulation models (Edson et al. 2011; Perna et al. 2012; Tan & Komacek 2019; Beltz et al. 2021) to exoplanet spectra (Fortney et al. 2010; Kempton et al. 2017; Mollière et al. 2017; Goyal et al. 2018, 2019, 2020). This presents an interesting opportunity for machine-learning retrievals to take advantage of these grids, and provide some comparison across different models. This was investigated in a study of brown dwarfs (Oreshenko et al. 2020), which compared grids from three different groups to highlight differences in the models.

The linear spacing of these model grids has one key disadvantage—as the number of parameters increases, the number of models required increases exponentially and becomes prohibitive. An advantage of using machine learning is that it is

 Original content from this work may be used under the terms of the [Creative Commons Attribution 4.0 licence](#). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

able to automatically disentangle the parameter effects by learning from a large number of examples. This suggests that the linearly sampled grid is suboptimal for machine-learning retrievals. The use of stratified sampling methods, a type of random sampling that ensures each subdivision (or *strata*) of parameter space is evenly sampled, has been demonstrated to optimize computer experiments for many years (e.g., McKay et al. 1979; Wang 2003; Chalom & de Prado 2012). Therefore, one would expect random and stratified sampling methods to outperform linear sampling in the problem of machine-learning atmospheric retrievals. In this paper we investigated to what degree this applies. We test different methods of sampling exoplanet model grids for various types of analyses. We create our own model grids with an increasing number of parameters and different sampling methods, and then compare the predictability of each parameter using the random forest (Márquez-Neila et al. 2018). We also consider the model grid from Goyal et al. (2019), and use an analytical approximation to create differently sampled versions. We then use the random forest to analyze the grids for different purposes.

## 2. Methods

### 2.1. Modelling

Here we describe the methods and assumptions used to generate our grids of atmospheric models.

#### 2.1.1. Analytical Model

In order to test sampling methods and grid sizes, we use a simplified analytical model, assuming an isothermal, isobaric atmosphere. This follows the work of Lecavelier Des Etangs et al. (2008), de Wit & Seager (2013), Bétrémieux & Swain (2017), Heng & Kitzmann (2017), Jordán & Espinoza (2018), Heng (2019), and Fisher & Heng (2018), and allows one to write down an analytical expression for the transit radius, given by Equation (2) in Fisher & Heng (2018). The atmospheric opacity is given by

$$\kappa = \sum_i \frac{X_i m_i \kappa_i}{m} + \kappa_{\text{CIA}} + \kappa_{\text{haze}} + \kappa_{\text{cloud}}, \quad (1)$$

where  $m$  is the mean molecular mass, and  $X_i$ ,  $m_i$ , and  $\kappa_i$  are the volume mixing ratio, mass, and opacity of species  $i$ , respectively.  $\kappa_{\text{CIA}}$  is the opacity associated with collision-induced absorption (both  $\text{H}_2\text{-H}_2$  and  $\text{H}_2\text{-He}$ ), taken from HITRAN (Rothman et al. 2013).  $\kappa_{\text{haze}}$  and  $\kappa_{\text{cloud}}$  follow different equations in two different models we consider (see Sections 2.1.2 and 2.1.3), but are generally associated with Rayleigh scattering and a gray cloud, respectively. The cross section due to Rayleigh scattering is taken from Vardya (1962).

To mimic spectra we expect to obtain from JWST’s NIRSpec Prism mode, we bin our models to  $\sim 400$  points in the range  $0.6\text{--}5.3\ \mu\text{m}$ , giving a resolution of  $\sim 100$ . We then add random Gaussian noise, assuming the uncertainty on each spectral point to be 20 ppm.

#### 2.1.2. Free Chemistry Models

For our first set of models we assume free chemistry. This means the abundances of each molecule can take any value, which allows for a greater freedom in the models, but could lead to unphysical compositions. For these models, we include a varying subset of the molecules  $\text{H}_2\text{O}$ ,  $\text{CO}$ ,  $\text{CO}_2$ ,  $\text{CH}_4$ ,  $\text{C}_2\text{H}_2$ ,  $\text{HCN}$ , and  $\text{NH}_3$ . The opacities for these molecules are

**Table 1**  
Prior Ranges for All Possible Parameters in Our Free Chemistry Model

Parameter	Prior Range
$T$ (K)	[500, 2900]
$\log X_i$	[−13, −1]
$\log \kappa_0$	[−10, −1]
$a$	[3, 6]
$\log r_c$	[−7, −1]

computed using the open-source HELIOS-K opacity calculator (Grimm & Heng 2015; Grimm et al. 2021), and the line lists are taken from the EXOMOL, HITRAN and HITEMP databases—Polyansky et al. (2018;  $\text{H}_2\text{O}$ ), Li et al. (2015;  $\text{CO}$ ), Rothman et al. (2010;  $\text{CO}_2$ ), Yurchenko & Tennyson (2014;  $\text{CH}_4$ ), Gordon et al. (2017;  $\text{C}_2\text{H}_2$ ), Barber et al. (2014;  $\text{HCN}$ ), and Yurchenko et al. (2011;  $\text{NH}_3$ ). The molecular opacities are sampled every  $0.01\ \text{cm}^{-1}$  in wavenumber space, at a pressure of 1 mbar. In each set of models, we vary the temperature and molecular abundances of the included species. For one version of this set we also include a nongray cloud model, following Equation (9) of Fisher & Heng (2018),

$$\kappa_{\text{cloud}} = \frac{\kappa_0}{Q_0 x^{-a} + x^{0.2}}. \quad (2)$$

This analytical cloud model comes from Kitzmann & Heng (2018). In this work, we vary three of the parameters—the factor  $\kappa_0$ , the index  $a$ , and the cloud particle size  $r_c$  (measured in centimeters). The cloud composition  $Q_0$  is set to 50, since previous studies have shown it to be unconstrained in retrievals (e.g., Fisher & Heng 2018). For this model,  $\kappa_{\text{haze}}$  is simply opacity due to Rayleigh scattering. The range of values spanned by the grids for all the possible parameters in our free chemistry model are shown in Table 1.

For all the models we assume the same planetary parameters as WASP-12 b:  $R_p = 1.79R_J$ ,  $g = 977\ \text{cm s}^{-2}$ ,  $R_* = 1.57R_\odot$ .

#### 2.1.3. Goyal Models

Our second set of models emulates the grid from Goyal et al. (2019), and thus are termed “Goyal models.” Goyal et al. (2019) present a scalable grid of exoplanet transmission spectra, varying the temperature, gravity, metallicity, C/O ratio, haze, and cloud. Their models are computed using ATMO—a 1D radiative-convective equilibrium model for planetary atmospheres (Amundsen et al. 2014; Tremblin et al. 2015; Drummond et al. 2016; Tremblin et al. 2016). They use isothermal  $P - T$  profiles, assuming chemical equilibrium. Full details on their implementation of the model can be found in the paper.

We simulate these models using the analytical model described in Section 2.1.1. To implement equilibrium chemistry in our model, for a given temperature, metallicity, and C/O ratio, we use the validated analytical model of Heng & Tsai (2016) that includes carbon, oxygen, and nitrogen. From this we obtain abundances for  $\text{H}_2\text{O}$ ,  $\text{CO}$ ,  $\text{CO}_2$ ,  $\text{CH}_4$ ,  $\text{C}_2\text{H}_2$ ,  $\text{HCN}$ , and  $\text{NH}_3$ . The opacities used for these molecules are the same as in Section 2.1.2.

In the grid from Goyal et al. (2019), the parameters are sampled as follows. The temperature is sampled every 100 K, from 300 to 2600 K. The surface gravity can take one of the four values, 5, 10, 20, or  $50\ \text{ms}^{-2}$ . The atmospheric metallicity

controls the elemental abundances, including oxygen, and is sampled at 0.1, 1, 10, 50, 100, and 200 times solar. The C/O ratio controls the carbon abundance, and takes the values 0.35, 0.56, 0.7, and 1.0. The haze parameter controls small scattering aerosol particles, and is implemented as  $\alpha_{\text{haze}}$  in the equation  $\sigma(\lambda) = \alpha_{\text{haze}}\sigma_0(\lambda)$ , where  $\sigma(\lambda)$  is the total scattering cross section, and  $\sigma_0(\lambda)$  is the H<sub>2</sub> Rayleigh scattering cross section.  $\alpha_{\text{haze}}$  is sampled at 1, 10, 100, and 1100. The cloud is treated as large particles with a gray opacity, and the parameter is implemented as  $\alpha_{\text{cloud}}$  in the equation  $\kappa(\lambda)_c = \kappa(\lambda) + 2\kappa_{\text{H}_2}\alpha_{\text{cloud}}$ , where  $\kappa(\lambda)_c$  is the total scattering opacity, and  $\kappa_{\text{H}_2}$  is the scattering opacity due to H<sub>2</sub> at 350 nm, which Goyal et al. (2019) states as  $\sim 2.5 \times 10^{-3} \text{ cm}^2 \text{ g}^{-1}$ .  $\alpha_{\text{cloud}}$  is sampled at 0, 0.06, 0.2, and 1.0. This leads to a total of 36,864 models.<sup>5</sup>

#### 2.1.4. Analytical versus Full Model Comparison

There are several key approximations in our analytical model, such as the isobaric atmosphere and constant chemical abundances. Figure 1 shows a comparison of six models with their corresponding model from Goyal et al. (2019). Each model has only one parameter changed with respect to the top left model. The top right model shows the effect of a higher temperature. In this case, the absorption due to TiO and VO dominates in the bluer wavelengths for the Goyal et al. (2019) model. Since we do not include TiO or VO in our models, we see a discrepancy here. The second-row, left-column panel shows a higher metallicity value. The agreement between the two models is very good, as the dominant absorbers for this set of parameters are present in both. The second-row, right-column panel shows a higher C/O ratio. Here we start to see another discrepancy between the models, which worsens with an increasing C/O ratio. This is due to the differing chemical models. Goyal et al. (2019) include many more species in their equilibrium model, leading to different abundances for the main absorbers. As a test, we took the abundances from the Goyal et al. (2019) model and ran the analytical model, which resulted in a very good agreement between the two (not shown). The bottom-row, left-column panel shows a higher haze parameter value, which controls the level of the Rayleigh scattering slope. Again, the agreement here is good. Similarly, the bottom-row, right-column panel shows a higher cloud value, and the agreement between the two models is good.

## 2.2. Sampling Techniques

Here we describe three possible sampling methods for model grids—linear sampling, random sampling, and Latin hypercube sampling. We perform a comparison of these methods, with the results shown in Section 3.2.1.

### 2.2.1. Linear Sampling

Traditional grids of models are typically sampled linearly (e.g., Allard et al. 2001; Goyal et al. 2019; Marley et al. 2021). This involves having one model for each possible combination of parameters, leading to  $X^n$  models, where  $n$  is the number of parameters and  $X$  is the number of values sampled for each one. This is shown in the left panel of Figure 2, for an example with two parameters, each sampled twice. One of the benefits of a

linear grid is that one is able to easily study the effects of a single parameter by comparing consecutive models. Varying only one parameter at a time prevents the effects from multiple parameters becoming entangled. This is extremely useful in forward modeling, where the main goal is to study these effects.

However, in the field of atmospheric retrieval these grids can prove challenging. Unless strong assumptions are made (e.g., chemical equilibrium), retrievals regularly contain  $\sim 10$ – $20$  parameters. This quickly escalates the linear grid to a completely unfeasible size. Nevertheless, linear grids are sometimes used with interpolation to perform traditional Bayesian retrievals of exoplanets (e.g., Miller et al. 2020; Mollière et al. 2020; Carrión-González et al. 2021).

### 2.2.2. Random Sampling

In our previous work on machine-learning retrievals (Márquez-Neila et al. 2018), we used randomly sampled grids for our training sets. This involves simply drawing each parameter at random from a uniform distribution inside the desired range. Unlike linear sampling, this grid does not allow one to compare models with only one differing parameter. However, for a fixed number of models, the random grid allows for more points in each parameter dimension to be sampled. The right panel of Figure 2 shows an example of a random grid with four models. This method proves beneficial for the random forest, which is able to automatically disentangle the effects of each parameter.

### 2.2.3. Latin Hypercube Sampling

A common sampling technique in machine learning is Latin hypercube sampling (LHS; McKay et al. 1979). Starting from a square grid with fixed sampling positions, a Latin square has exactly one sample in each row and column. A Latin hypercube (LH) is the generalization of this to higher dimensions. The middle panel of Figure 2 shows an example of a Latin square with four models. Latin squares and hypercubes have been used in the design of experiments for almost a century, and provide a method for improving inference from a sparse sampling of high-dimensional space. This has already proven extremely useful in the field of cosmology (e.g., Kaufman et al. 2011; Albers et al. 2019; Rogers et al. 2019; Wibking et al. 2020), but has yet to be taken advantage of in other fields of astrophysics.

In comparison to simple random sampling, the key advantage of LHS is that it guarantees a better representation of the real variability of the parameters. Random sampling has no such guarantees. Therefore, for an inference problem with a large number of parameters, LHS typically requires fewer samples than random sampling. See the Appendix for more details on the history and applications of LHs in experiments.

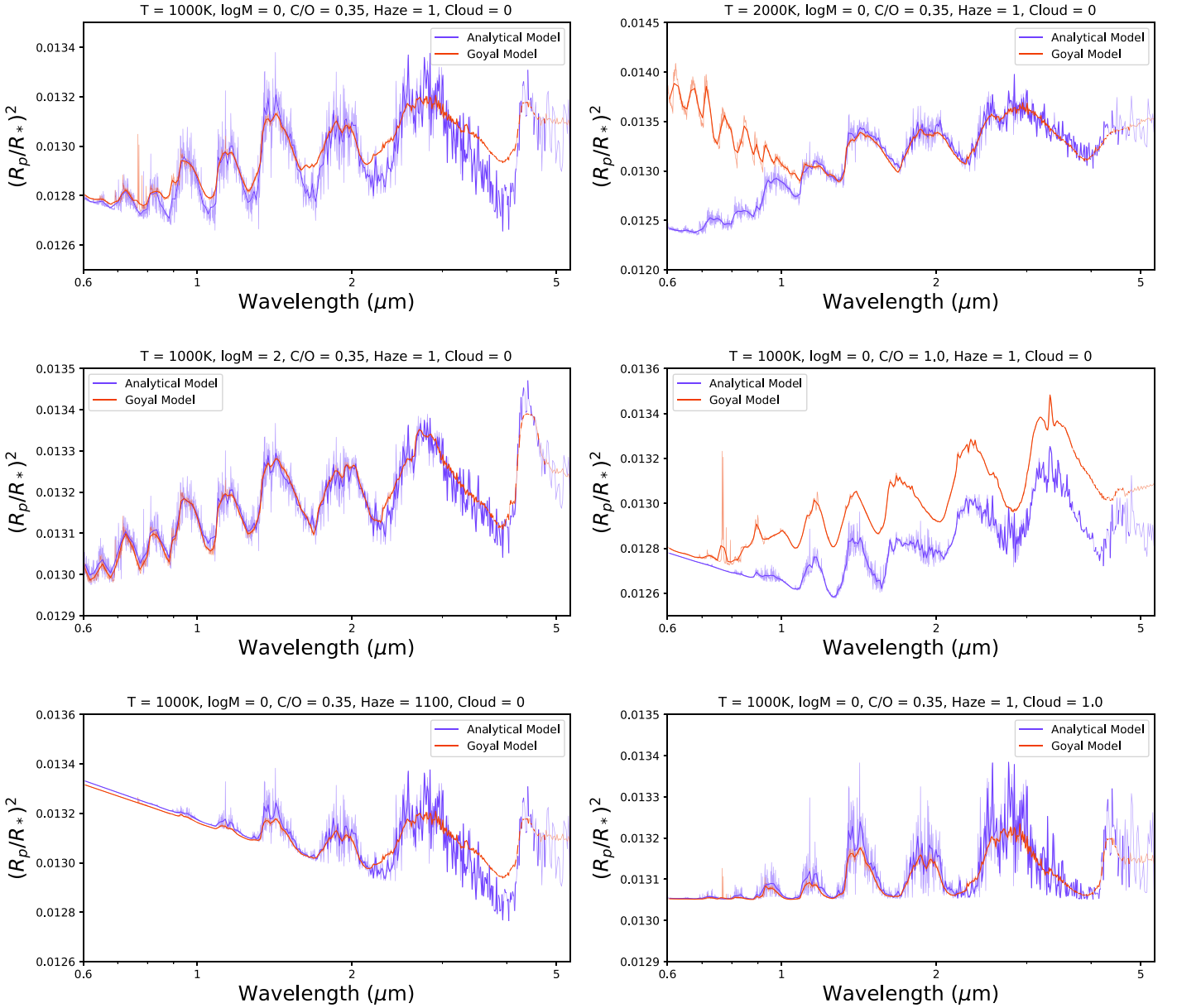
## 3. Results

### 3.1. Free Chemistry Grid

#### 3.1.1. Sampling Comparison

To compare the different methods, we created training sets using each of the three sampling techniques, and then trained and tested a corresponding random forest. For a fair comparison, we assumed a fixed number of models can be generated, such that all three training sets have the same size.

<sup>5</sup> Note that some of the parameter values in the open-source grid have changed since Goyal et al. (2019) was published, and the information on this can be found in the `readme.txt` file in their Google Drive.



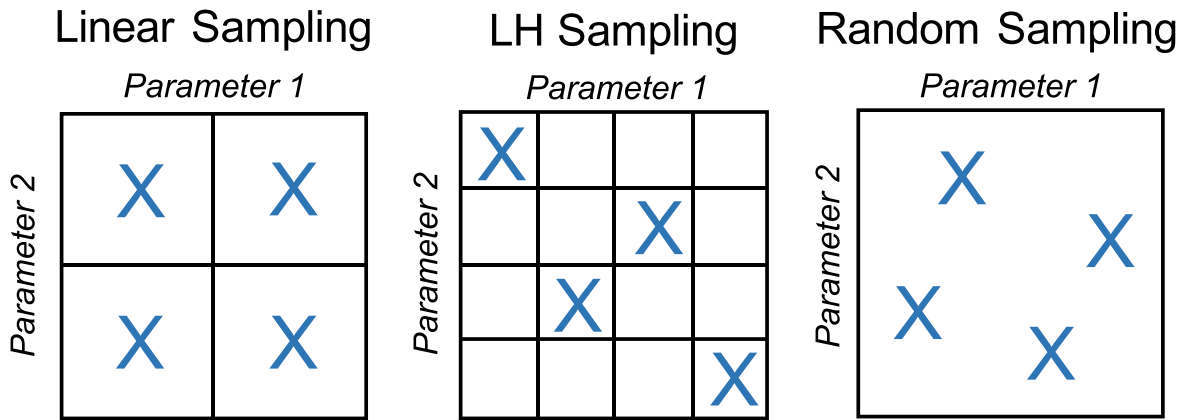
**Figure 1.** Comparison of our analytical models with the corresponding models from Goyal et al. (2019), shown for different parameter values. The analytical model is shown in blue and the model from Goyal et al. (2019) is shown in red.

This is also a realistic situation in which one is limited by computation time. We considered five different models, with 2, 4, 6, 8, and 11 parameters. For linear sampling, we sampled each parameter an equal number of times, given by  $X$ , and calculated as the highest integer such that  $X^n \sim 100,000$ . This results in the training set sizes shown in Table 2. These sizes are kept the same for the random sampling and LHS training sets. The parameters in the linear and LHS grids are evenly spaced inside the prior range (shown in Table 1), while the parameters in the random grids are drawn randomly from a uniform distribution across this range.

For each model, a forest is trained on each of the differently sampled grids. Due to its ability to make fast predictions, the forest can be tested on a large set of models, spanning the whole parameter space. We randomly generated a testing set of 10,000 models, keeping it the same across each sampling case. Once tested, the forest provides a useful predictability analysis, by calculating the coefficient of determination ( $R^2$  score)

between the real and predicted values of each parameter. The  $R^2$  score varies from  $-1$  to  $1$ , where values near unity indicate strong anticorrelations and correlations, respectively, between the real and predicted values of a given parameter. Figure 3 shows the coefficient of determination for each parameter and sampling technique in the different models.

For the 2 and 4 parameter models, all three training sets perform comparably well. This is expected as even in the linear case, each parameter is still sampled sufficiently densely for the forest to learn its effects. For the 6 parameter model, the linear case starts to drop in performance, when compared with the random and LHS cases. For the 8 parameter model, the linear case shows an extremely poor predictability for certain parameters, including temperature, which had previously been relatively easy to predict. This is because each parameter is only sampled four times, providing the forest with little information on the parameters' effects on the spectrum. The random and LHS cases show fairly low  $R^2$  scores for the newly



**Figure 2.** Schematic of different methods for sampling four models from a grid with two parameters. The first panel shows linear sampling, where each parameter has two values and every combination of values is sampled. The second panel shows a Latin square, where the value of each parameter is chosen randomly, but each value is chosen exactly once, allowing for four parameter values each in the grid. The third panel shows completely random sampling, without using a grid of values.

**Table 2**

Table Showing the Training Set Size for Our 2, 4, 6, 8, and 11 Parameter Models

# Parameters	# Parameter Samples	Training Set Size
2	316	99,856
4	17	83,521
6	6	46,656
8	4	65,536
11	3	177,147

added parameters ( $\text{NH}_3$  and  $\text{C}_2\text{H}_2$ ), but the other parameters remain fairly well predicted. The lower  $R^2$  scores for  $\text{NH}_3$  and  $\text{C}_2\text{H}_2$  is likely due to their lack of strong, distinctive molecular features, and degeneracies with the other molecules. The 11 parameter model shows an even more extensive difference, with several parameters in the linear grid dropping into negative predictability.

Generally these results are unsurprising, as it is trivial that parameters sampled only three or four times in the training set will be hard to retrieve. However, these linear grids are often used with an interpolation scheme in traditional Bayesian retrievals (e.g., Mollière et al. 2017; Miller et al. 2020; Carrión-González et al. 2021), and these results highlight issues that can arise from this. One interesting takeaway from these results is that the  $R^2$  score for each parameter in the random and LHS models remains at a very similar value across each model, as more parameters are added. This suggests that the addition of extra parameters does not necessarily require a higher number of samples in the training set. However, it is possible that this could be due to the distinct effects each parameter has on the spectrum in this specific case, as more interconnected parameters could be harder to disentangle. So far, we see very little difference between the random and LHS cases, most likely because the number of parameters is still relatively low.

### 3.1.2. Mock Retrieval

One potential advantage of a linearly spaced grid is that it allows for easy interpolation, and can therefore be used in a traditional, Bayesian retrieval with an MCMC or nested sampling. This is beneficial when the forward model is quite slow, as a traditional retrieval needs to compute models on the fly, typically tens of thousands of times for a single run. By

computing a grid in advance, the computational burden is shifted offline, and the models can be reused for multiple retrievals. To study this, we ran several retrievals on a simulated spectrum using our free chemistry models. Figure 4 shows three retrievals performed on the same simulated spectrum for the 11 parameter model. The first uses nested sampling by interpolating on the linear grid. The second uses the random forest trained on the random grid. The third uses nested sampling with the full analytical model.

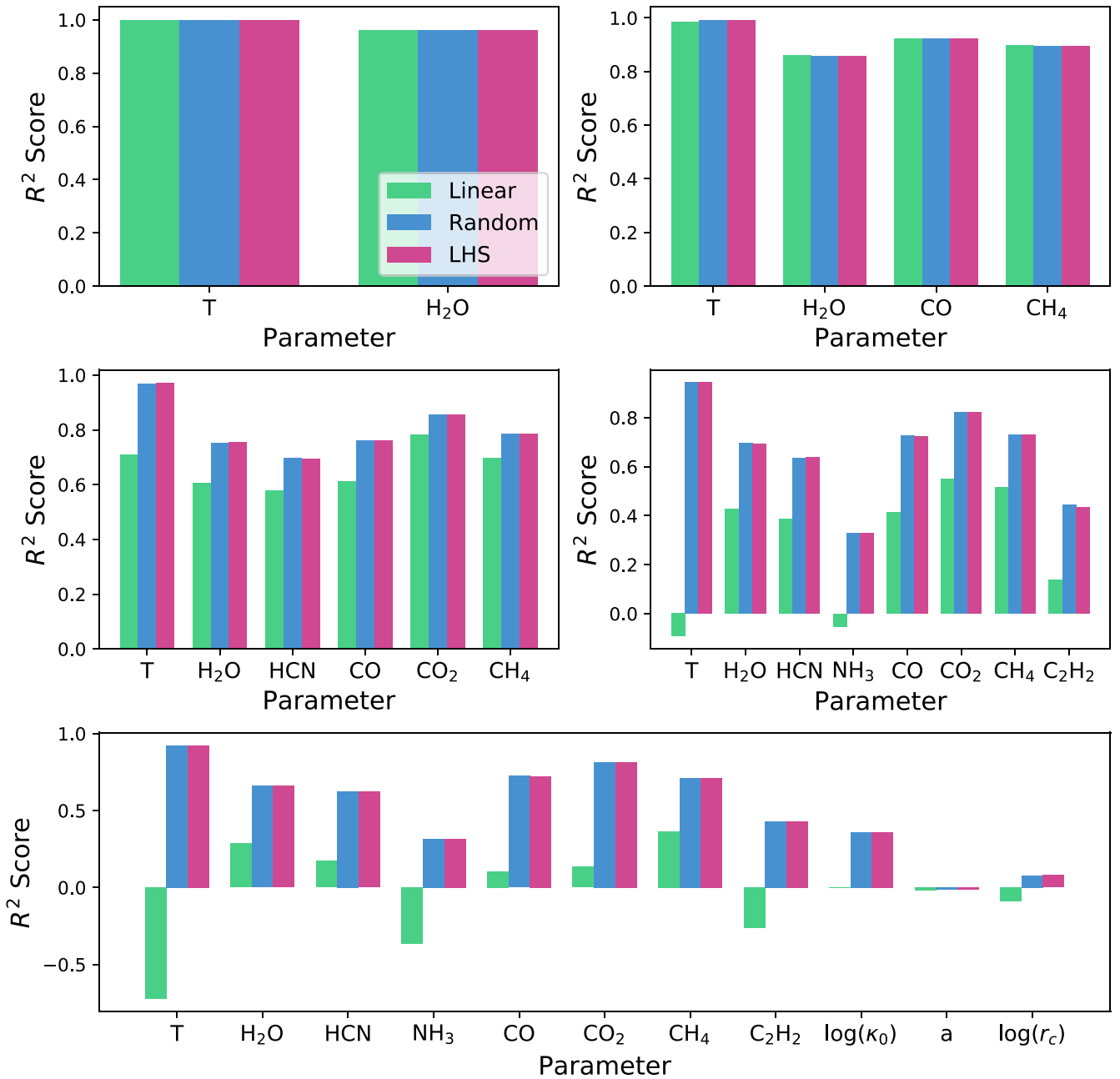
The key difference between the full analytical retrieval and the random forest trained on the random grid is the width of the posteriors. For the tightly constrained parameters in the full retrieval, such as temperature, CO, and  $\text{CH}_4$ , the forest’s posterior generally peaks in the same place, but with a wider distribution. This is due to nested sampling’s ability to hone in on a small part of parameter space. For the parameters with an upper bounded full retrieval posterior, such as  $\text{H}_2\text{O}$ , HCN,  $\text{NH}_3$ ,  $\text{CO}_2$ , and  $\text{C}_2\text{H}_2$ , the upper limit for the forest’s posterior is about 2 dex higher. For the completely unconstrained cloud parameters, the forest’s posteriors are comparable. Improvements can be made on the forest’s posteriors by adjusting parameters in the forest such as the number of trees or tree depth, increasing the size of the training set, or incorporating a likelihood into the posterior computation (Nixon & Madhusudhan 2020). However, the latter negates one advantage of the traditional likelihood-free random forest, which does not rely on assuming a functional form of the likelihood.

Unsurprisingly, the nested-sampling retrieval using interpolation on the linear grid performs poorly due to the sparse sampling of the parameters. It would not be appropriate to use interpolation for a grid with this many dimensions.

## 3.2. Goyal Grid

### 3.2.1. Sampling Comparison

Using the analytical model described in Section 2.1.1, we created four different versions of the Goyal grid. The first two are linearly spaced grids of different sizes. We started by following the parameter spacing from Goyal et al. (2019), described in Section 2.1.3. This consists of 24 temperatures, 4 gravities, 6 metallicities, 4 C/O ratios, 4 haze parameters, and 4 cloud parameters, leading to a total of 36,864 models. Next we created a sparser version of this linear grid, sampling fewer



**Figure 3.**  $R^2$  scores for each parameter in each of the free chemistry models, using the three different grid-sampling methods—linear sampling, random sampling, and Latin hypercube sampling.

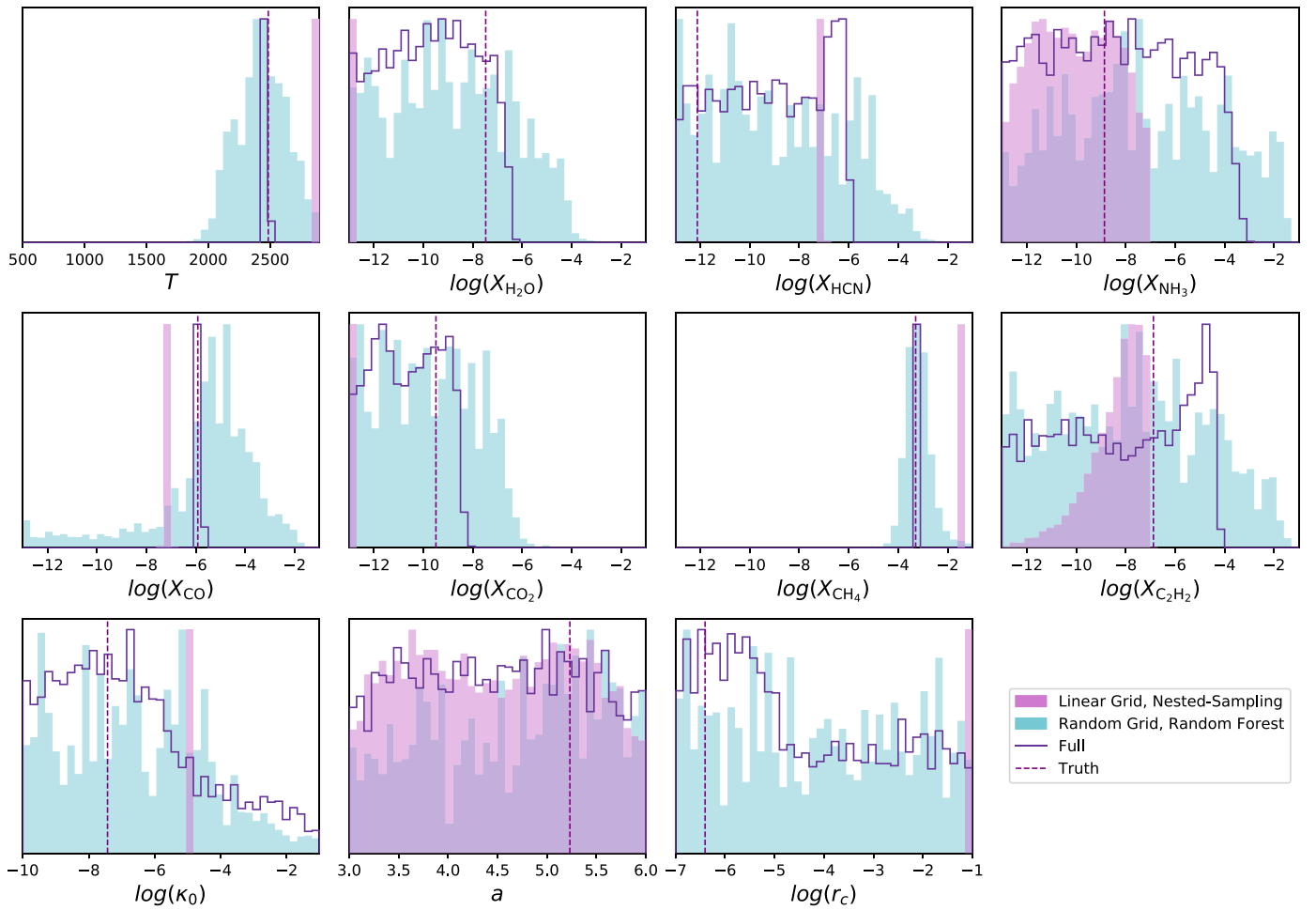
parameter values. This consists of 5 temperatures (300, 900, 1500, 2100, 2600 K), 3 gravities (5, 14, 50  $\text{ms}^{-2}$ ), 4 metallicities (0.1, 1, 50, 200), 3 C/O ratios (0.35, 0.63, 1.0), 3 haze parameters (1, 31.6, 1100), and 3 cloud parameters (0, 0.13, 1.0), leading to a total of 1620 models. A summary of the parameter spacing for both versions of the linear grid is shown in Table 3. Note that in the analysis we converted the gravity, metallicity, and haze parameters to log quantities.

The second two grids use different sampling methods. First is a randomly sampled set of 1620 models, created by drawing each parameter from a uniform distribution (or log-uniform for metallicity, gravity, and the haze parameter) in the same range as in Goyal et al. (2019). The final grid is a set of 1620 models using LHS, where each parameter dimension has 1620 evenly

spaced points (or even in log-space for metallicity, gravity, and the haze parameter) in the same range as in Goyal et al. (2019).

We trained a random forest on each of the four grids, and then tested them on a randomly generated set of 5000 models. Figure 5 shows the  $R^2$  scores for each parameter across the different grids.

For every parameter, the sparse linear grid is outperformed somewhat by all other grids, including the random and LHS grids that contain the same number of models. For most parameters, the denser linear grid, the random grid, and the LHS grid give comparable results, despite differing in size by more than a factor of 20. Although the predictability of the C/O ratio is significantly higher for the random and LHS grids, this is actually due to the uneven spacing adopted for the linear



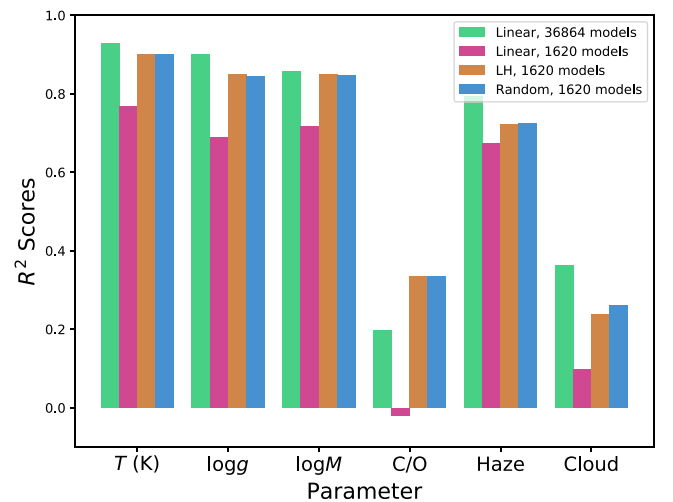
**Figure 4.** Retrieval of a mock spectrum from the 11 parameter model using different methods. This shows the retrievals using nested sampling, interpolating on the linear grid, and the random forest trained on the random grid, compared with the full retrieval using nested sampling with the analytical model computed on the fly. The dashed purple lines show the true parameter values for the spectrum.

**Table 3**

Parameter Values for the Grid from Goyal et al. (2019), and the Values in Our Sparsely Sampled Linear Grid

Parameter	Goyal Grid	Sparse Grid
$T$ (K)	(300–2600), in steps of 100	(300, 900, 1500, 2100, 2600)
$g$ ( $\text{ms}^{-2}$ )	(5, 10, 20, 50)	(5, 14, 50)
metallicity (x solar)	(0.1, 1, 10, 50, 100, 200)	(0.1, 1, 50, 200)
C/O	(0.35, 0.56, 0.7, 1.0)	(0.35, 0.63, 1.0)
Haze	(1, 10, 100, 1100)	(1, 31.6, 1100)
Cloud	(0, 0.06, 0.2, 1.0)	(0, 0.13, 1.0)

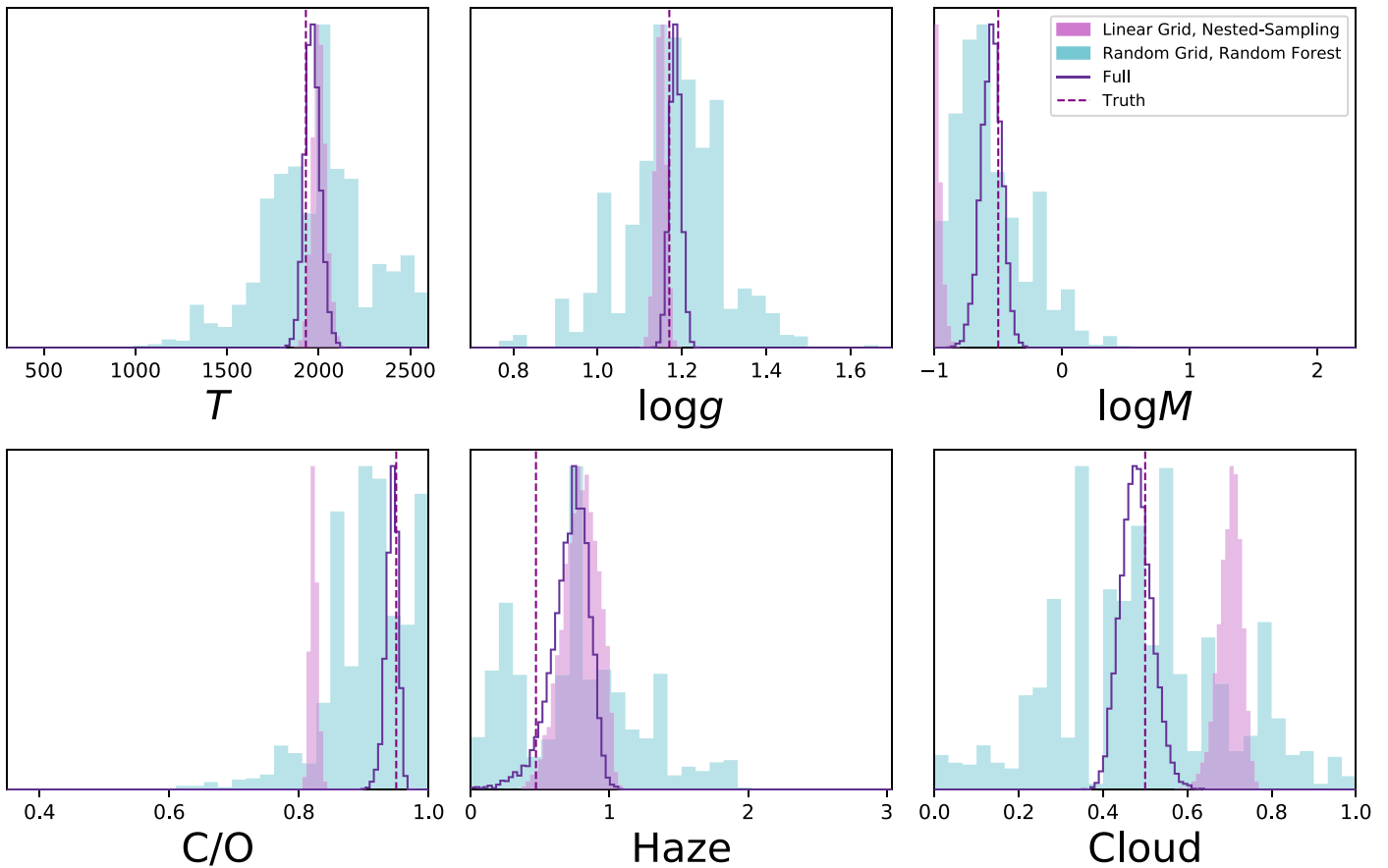
grid. In Goyal et al. (2019), the C/O ratio takes the values 0.35, 0.56, 0.7, and 1.0, leaving a larger gap at the higher values. Due to the nonlinear effect of the C/O ratio on the spectrum, it proves challenging for the forest to accurately interpret models in this high C/O range. In contrast, for temperature even the sparse linear grid performs relatively well. In fact, all four well-predicted parameters (i.e., temperature, gravity, metallicity, and haze) have comparable  $R^2$  scores across all models, to within  $\sim 0.1$ . This contrasts the results from Section 3.1.1, but could be explained by the relatively low number of parameters in the model and the linear effects of these four parameters on the spectrum. This motivates a differently structured grid, with



**Figure 5.**  $R^2$  scores for each parameter in the Goyal model, using the three different grid-sampling methods—linear sampling, random sampling and Latin hypercube sampling. Linear sampling is tested for a dense grid of 36,864 models, and a sparse grid of 1620 models.

denser sampling for parameters with highly nonlinear effects on the spectrum.

Since LHS guarantees a better representation of the real variability of the parameters, we might expect it to outperform



**Figure 6.** Retrieval of a mock spectrum using different methods. This shows the retrievals using nested sampling, interpolating on the dense linear grid, and the random forest trained on the dense random grid, compared with the full retrieval using nested sampling with the analytical model computed on the fly. The dashed purple lines show the true parameter values for the spectrum.

random sampling. The fact that the results in the bar chart in Figure 5 look comparable between the two could be due to the relatively low number of parameters. Perhaps a higher number of dimensions could lead to a divergence in their performances.

### 3.2.2. Mock Retrieval

We also ran mock retrievals using the Goyal grids. One example is shown in Figure 6. This figure shows three retrievals. The first uses nested sampling with interpolation on the dense linear grid. The second uses the random forest, trained on a randomly sampled grid of 36,864 models (i.e., the same size as the dense linear grid). The third uses nested sampling with the full analytical model, computed on the fly. In contrast to Figure 4, the linear interpolation retrieval does not perform so badly. Here, the temperature, gravity, and haze posteriors are very similar to the full retrieval, with only minor offsets. Of course this is due to denser sampling of the parameters, although the gravity and haze, for example, only sample one extra point than the parameters in the free chemistry model from Figure 4. It could be that the effects of these parameters are more linear than those in the free chemistry model. The poor performance of the linear interpolation retrieval is seen again in the metallicity, C/O ratio, and the cloud parameters. This is likely due to the more nonlinear effects of these parameters, and the uneven spacing (for the C/O ratio in particular).

The random forest posteriors exhibit a similar behavior as in Figure 4, with wider, less constrained distributions, but with the

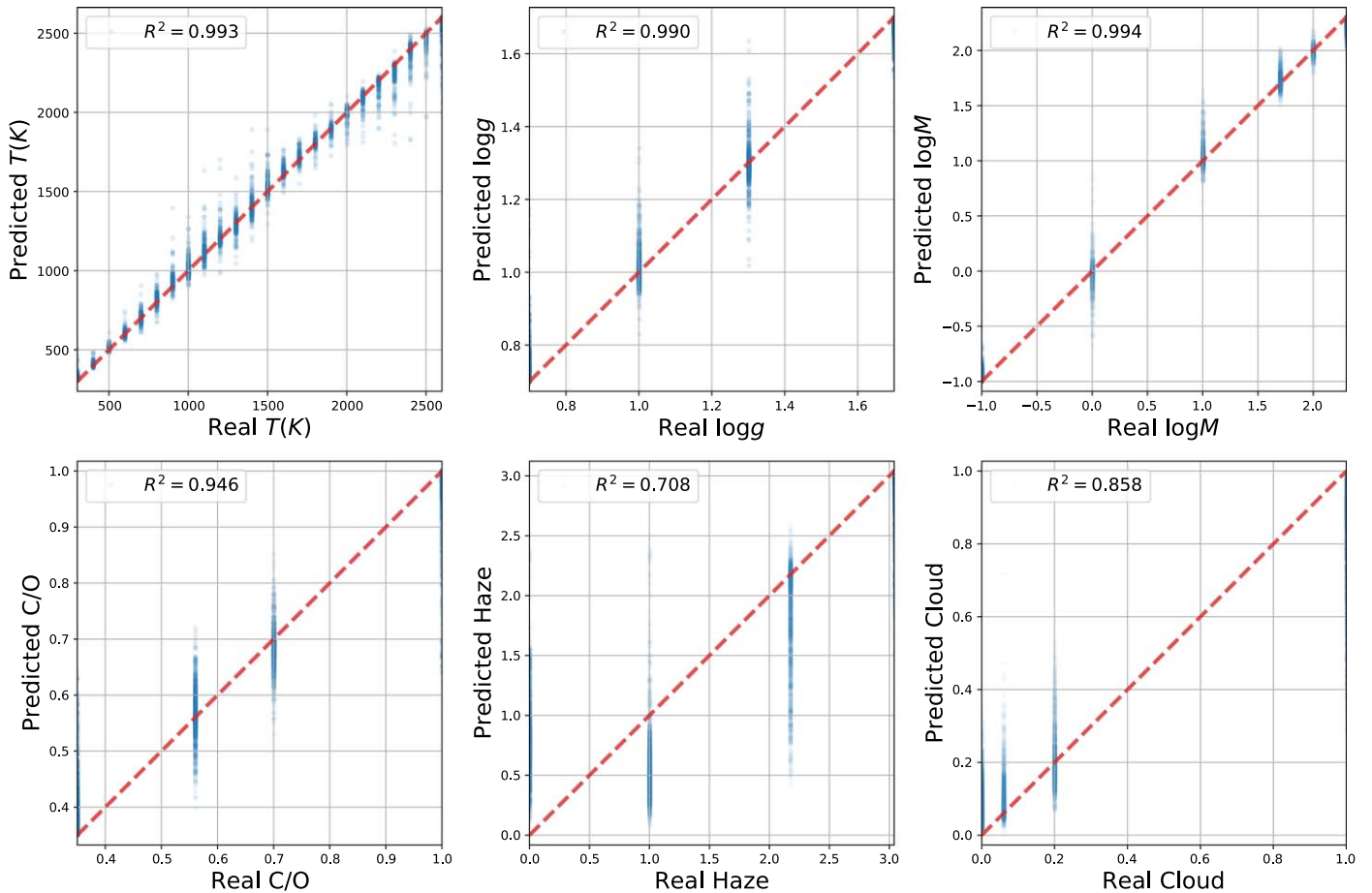
same peak location. This effect is exaggerated for the cloud parameter, for which the forest’s posterior encompasses the entire range of values. This is consistent with the  $R^2$  scores in Figure 5, which shows that the forest struggles to accurately retrieve the cloud parameter.

### 3.2.3. Information Content of the Goyal Grid

We also trained a forest on the original grid of models from Goyal et al. (2019). To be consistent, we binned these models down to the same resolution and wavelength range, and added the same noise level, as in Section 2.1.1. We then randomly selected 6864 models to be the testing set, leaving 30,000 for training. Figure 7 shows the predicted versus real values for this forest. Although the  $R^2$  scores are high for all six parameters, an important caveat is that the testing set contains the same evenly spaced parameter values as the training set. This makes it significantly easier for the forest to predict the correct answer. The sparse testing set provides no information about how the forest performs on spectra with parameters in between these values. We would need to generate our own ATMO models with randomly chosen parameters in order to test this forest’s ability to generalize.

In addition, we computed the “feature importance” for this forest. This determines the information content of each spectral point with respect to each parameter in the retrieval. Figure 8 shows the results for the original Goyal grid. In this figure, the feature importance is shown against two spectra with a high and low value of the relevant parameter, to provide context.





**Figure 7.** Real vs. predicted values for the random forest trained on the models from Goyal et al. (2019), binned down to match the resolution and wavelength coverage of JWST NIRSpec Prism. The  $R^2$  score varies from  $-1$  to  $1$ , where values near unity indicate strong anticorrelations and correlations, respectively, between the real and predicted values of a given parameter.

Some aspects of the feature importance are intuitive, such as the haze parameter drawing most information from the bluer wavelengths, where the Rayleigh slope is visible. Parameters that are less well constrained typically have a slightly more uniform feature importance across all spectral points, as is seen for the cloud parameter. Temperature has major peaks at the TiO features, implying that this species performs well as a thermometer for the atmosphere. The C/O feature importance appears to peak around the water features at  $2$  and  $3 \mu\text{m}$ , which are present in spectra with lower C/O ratios. These features coincide with troughs in the methane opacity, which dominates the high-carbon spectra, making these areas good indicators of the C/O ratio. The same behavior in the feature importance is found for the forests trained on the analytical model, using both the linear and random grids (not shown).

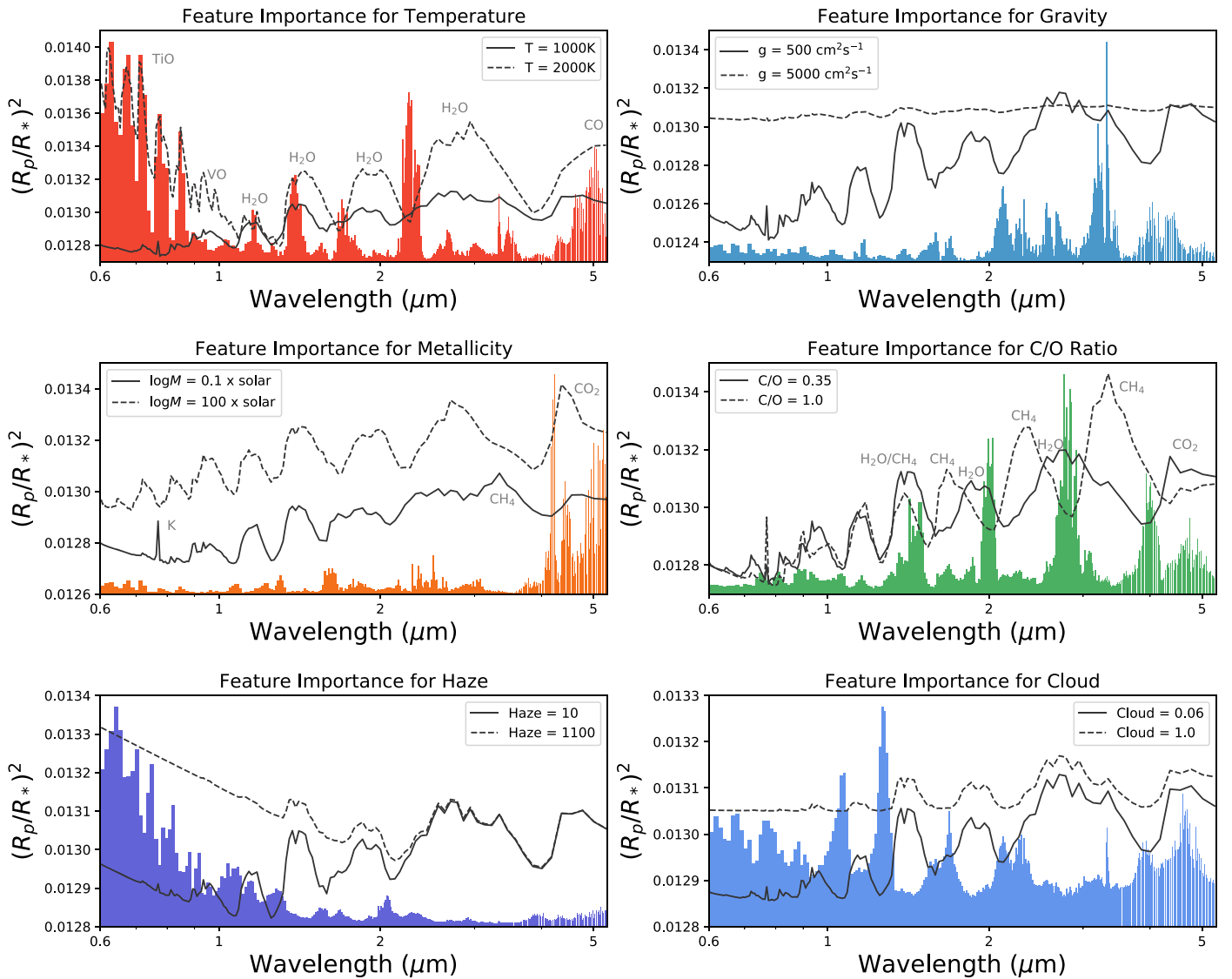
#### 4. Conclusion

We generated differently sampled grids for two types of atmospheric models, and compared their performance using different types of analyses. For our free chemistry model, we found that random and LH sampling outperformed linear sampling for our 8 and 11 parameter models, but obtained comparable results for our 2, 4, and 6 parameter models (Figure 3). Our free chemistry mock retrieval for the 11 parameter model clearly demonstrated that linear interpolation

is not appropriate for high-dimensional models, as expected (Figure 4). For our models simulating those of Goyal et al. (2019), we found that the difference between the linear and random or LHS was less significant, particularly for parameters with generally high predictability, likely due to the lower dimensionality of the model. Our mock retrieval showed that the linear interpolation retrieval performed well for several parameters, but struggled with parameters with nonlinear effects, such as the C/O ratio.

These results warn against the use of linear interpolation of precomputed linear model grids for atmospheric retrievals. They also demonstrate the known results that random and LHS enable inference from a sparsely sampled parameter space. We did not find an improvement in LHS over random sampling, likely due to the relatively low dimensionality of our models. Although they are not ideal for building into a retrieval, linearly sampled grids have their own advantages. For example, they enable one to easily compare spectra differing in only one parameter, allowing for the clear analysis of individual parameter effects (e.g., Goyal et al. 2019). Furthermore, the feature importance for the linear grid from Goyal et al. (2019) provided extremely useful analysis of the information content of the spectra, although this could also be performed on the other grids.

Therefore, the answer for which type of sampling to use when constructing a model grid depends heavily on the



**Figure 8.** Relative feature importance of each wavelength point for each parameter of the model. Plotted over the top are two models with high and low values of the corresponding parameter, showing examples of the behavior of different spectra. This allows for the comparison between spectral features and their relative importance for each parameter. The y-axes correspond to the transit depth for the models. The feature importance values add up to unity, though their values are not shown.

intended use of the grid. For retrievals, in particular for machine learning, randomly sampled grids are likely to provide better results, with higher predictability for each model parameter, and require far fewer models. However, for analysis of the model physics and spectral sensitivity, a linearly sampled grid is preferable, due to the ease of model comparisons. Information content analysis can be performed in either case, providing similar results, beneficial for observing proposals or even informing future telescope development.

We thank Daniel Kitzmann for helpful discussions about the model, and Pablo Márquez-Neila for advice about the random forest. We acknowledge financial support from the Swiss National Science Foundation, the European Research Council (via a Consolidator Grant to KH; grant number 771620), the PlanetS National Center of Competence in Research (NCCR), the Center for Space and Habitability (CSH), and the Swiss-based MERAC Foundation.

## Appendix Latin Hypercubes in Computer Experiments

The use of Latin squares in the design of experiments, particularly in agriculture, dates back almost a century (Fisher 1935). Its modern applications to computer experiments enables inference from a sparse coverage of high-dimensional parameter space (Santner et al. 2003; Fang et al. 2006; Kleijnen 2015). However, the use of Latin squares, and the higher-dimensional LHs, has been relatively limited in astrophysics, with the majority of applications in cosmology. Kaufman et al. (2011) use LHs to improve the efficiency of emulators, and apply it to photometric redshifts of galaxies. More recently, Wibking et al. (2020) use LHs for their emulation framework modeling galaxy clustering, while Albers et al. (2019) implement them in the training of a neural network to speed up Einstein–Boltzmann solvers in cosmological simulations. Rogers et al. (2019) also use LHs in their emulator for the Ly $\alpha$  forest, for example.

There has been a great amount of work in the statistics community on the optimization of Latin hypercube designs (LHDs) to improve their efficiency and apply them to ensemble models, such as orthogonal LHDs (e.g., Sun et al. 2010), sliced LHDs (e.g., Qian 2012; Ba et al. 2015), and progressive LHS (e.g., Sheikholeslami & Razavi 2017). There are also many packages in the R public domain software environment for using LHS for computer experiments (e.g., lhs (Carnell 2020); DiceDesign and DiceEval (Dupuy et al. 2015); DiceKriging (Roustant et al. 2012); DiceView (Richet et al. 2020); tgp: Bayesian Treed Gaussian Process Models (Gramacy 2007; Gramacy & Taddy 2010)).

### ORCID iDs

Chloe Fisher  <https://orcid.org/0000-0003-0652-2902>  
Kevin Heng  <https://orcid.org/0000-0003-1907-5910>

### References

- Albers, J., Fidler, C., Lesgourgues, J., Schöneberg, N., & Torrado, J. 2019, *JCAP*, 2019, 028
- Allard, F. 2014, in Proc. of the IAU 299, Exploring the Formation and Evolution of Planetary Systems (Cambridge: Cambridge Univ. Press), 271
- Allard, F., Hauschildt, P. H., Alexander, D. R., Tamanai, A., & Schweitzer, A. 2001, *ApJ*, 556, 357
- Amundsen, D. S., Baraffe, I., Tremblin, P., et al. 2014, *A&A*, 564, A59
- Ardevol Martinez, F., Min, M., Kamp, I., & Palmer, P. I. 2022, arXiv:2203.01236
- Ba, S., Myers, W. R., & Brenneman, W. A. 2015, *Technometrics*, 57, 479
- Barber, R. J., Strange, J. K., Hill, C., et al. 2014, *MNRAS*, 437, 1828
- Beltz, H., Rauscher, E., Brogi, M., & Kempton, E. M. R. 2021, *AJ*, 161, 1
- Benneke, B., & Seager, S. 2013, *ApJ*, 778, 153
- Bétrémieux, Y., & Swain, M. R. 2017, *MNRAS*, 467, 2834
- Burrows, A., Marley, M., Hubbard, W. B., et al. 1997, *ApJ*, 491, 856
- Carnell, R. 2020, lhs: Latin Hypercube Samples, R package v1.1.5, <https://CRAN.R-project.org/package=lhs>
- Chalom, A., & de Prado, P. I. d. K. L. 2012, arXiv:1210.6278
- Cobb, A. D., Himes, M. D., Soboczenski, F., et al. 2019, *AJ*, 158, 33
- de Wit, J., & Seager, S. 2013, *Sci*, 342, 1473
- de Wit, J., Wakeford, H. R., Lewis, N. K., et al. 2018, *NatAs*, 2, 214
- Drummond, B., Tremblin, P., Baraffe, I., et al. 2016, *A&A*, 594, A69
- Dupuy, D., Helbert, C., & Franco, J. 2015, *JoSS*, 65, 1
- Edson, A., Lee, S., Bannon, P., Kasting, J. F., & Pollard, D. 2011, *Icar*, 212, 1
- Fang, K.-T., Li, R., & Sudjianto, A. 2006, Design and Modeling for Computer Experiments (London: Chapman & Hall), doi:10.1201/9781420034899
- Feng, Y. K., Line, M. R., Fortney, J. J., et al. 2016, *ApJ*, 829, 52
- Fisher, C., & Heng, K. 2018, *MNRAS*, 481, 4698
- Fisher, C., Hoeijmakers, H. J., Kitzmann, D., et al. 2020, *AJ*, 159, 192
- Fisher, R. A. 1935, The Design of Experiments (Edinburgh: Oliver and Boyd)
- Fortney, J. J., Shabram, M., Showman, A. P., et al. 2010, *ApJ*, 709, 1396
- Carrión-González, Ó., García Muñoz, A., Santos, N. C., et al. 2021, *A&A*, 655, A92
- Gordon, I. E., Rothman, L. S., Hill, C., et al. 2017, *JQSRT*, 203, 3
- Goyal, J. M., Mayne, N., Drummond, B., et al. 2020, *MNRAS*, 498, 4680
- Goyal, J. M., Mayne, N., Sing, D. K., et al. 2018, *MNRAS*, 474, 5158
- Goyal, J. M., Wakeford, H. R., Mayne, N. J., et al. 2019, *MNRAS*, 482, 4503
- Gramacy, R. B. 2007, *JoSS*, 19, 1
- Gramacy, R. B., & Taddy, M. 2010, *JoSS*, 33, 1
- Grimm, S. L., & Heng, K. 2015, *ApJ*, 808, 182
- Grimm, S. L., Malik, M., Kitzmann, D., et al. 2021, *ApJS*, 253, 30
- Heng, K. 2019, *MNRAS*, 490, 3378
- Heng, K., & Kitzmann, D. 2017, *MNRAS*, 470, 2972
- Heng, K., & Tsai, S.-M. 2016, *ApJ*, 829, 104
- Irwin, P. G. J., Parmentier, V., Taylor, J., et al. 2020, *MNRAS*, 493, 106
- Jordán, A., & Espinoza, N. 2018, *RNAAS*, 2, 149
- Kaufman, C. G., Bingham, D., Habib, S., Heitmann, K., & Frieman, J. A. 2011, *AnApS*, 5, 2470
- Kempton, E. M. R., Lupu, R., Owusu-Asare, A., Slough, P., & Cale, B. 2017, *PASP*, 129, 044402
- Kitzmann, D., & Heng, K. 2018, *MNRAS*, 475, 94
- Kleijnen, J. 2015, Design and Analysis of Simulation Experiments, International Series in Operations Research & Management Science (2nd edn.; Germany: Springer Verlag),
- Lecavelier Des Etangs, A., Pont, F., Vidal-Madjar, A., & Sing, D. 2008, *A&A*, 481, L83
- Li, G., Gordon, I. E., Rothman, L. S., et al. 2015, *ApJS*, 216, 15
- Line, M. R., & Parmentier, V. 2016, *ApJ*, 820, 78
- Madhusudhan, N., & Seager, S. 2009, *ApJ*, 707, 24
- Marley, M. S., Saumon, D., Visscher, C., et al. 2021, *ApJ*, 920, 85
- Márquez-Neila, P., Fisher, C., Sznitman, R., & Heng, K. 2018, *NatAs*, 2, 719
- Matchev, K. T., Matcheva, K., & Roman, A. 2022, arXiv:2201.02696
- McKay, M. D., Beckman, R. J., & Conover, W. J. 1979, *Technometrics*, 21, 239
- Miller, L. P., Roudier, G., Swain, M., & Welsh, W. 2020, AAS Meeting Abstract, 235, 173.16
- Mollière, P., Stolker, T., Lacour, S., et al. 2020, *A&A*, 640, A131
- Mollière, P., van Boekel, R., Bouwman, J., et al. 2017, *A&A*, 600, A10
- Nixon, M. C., & Madhusudhan, N. 2020, *MNRAS*, 496, 269
- Oreshenko, M., Kitzmann, D., Márquez-Neila, P., et al. 2020, *AJ*, 159, 6
- Perna, R., Heng, K., & Pont, F. 2012, *ApJ*, 751, 59
- Polyansky, O. L., Kyuberis, A. A., Zobov, N. F., et al. 2018, *MNRAS*, 480, 2597
- Qian, P. Z. G. 2012, *J. Am. Stat. Assoc.*, 107, 393
- Richet, Y., Deville, Y., & Chevalier, C. 2020, DiceView: Methods for Visualization of Computer Experiments Design and Surrogate, <https://CRAN.R-project.org/package=DiceView>
- Rogers, K. K., Peiris, H. V., Pontzen, A., et al. 2019, *JCAP*, 2019, 031
- Rothman, L. S., Gordon, I. E., Babikov, Y., et al. 2013, *JQSRT*, 130, 4
- Rothman, L. S., Gordon, I. E., Barber, R. J., et al. 2010, *JQSRT*, 111, 2139
- Roustant, O., Ginsbourger, D., & Deville, Y. 2012, *JoSS*, 51, 1
- Santner, T., Williams, B., & Notz, W. 2003, The Design and Analysis of Computer Experiments (New York: Springer), doi:10.1007/978-1-4757-3799-8
- Sheikholeslami, R., & Razavi, S. 2017, *Environ. Model. Softw.*, 93, 109
- Sun, F., Liu, M.-Q., & Lin, D. K. 2010, *J. Stat. Plan. Inference*, 140, 3236
- Tan, X., & Komacek, T. D. 2019, *ApJ*, 886, 26
- Taylor, J., Parmentier, V., Irwin, P. G. J., et al. 2020, *MNRAS*, 493, 4342
- Tremblin, P., Amundsen, D. S., Chabrier, G., et al. 2016, *ApJ*, 817, L19
- Tremblin, P., Amundsen, D. S., Mourier, P., et al. 2015, *ApJ*, 804, L17
- Vardya, M. S. 1962, *ApJ*, 135, 303
- Waldmann, I. P. 2016, *ApJ*, 820, 107
- Wang, G. 2003, *J. Mech. Design*, 125, 210
- Wibking, B. D., Weinberg, D. H., Salcedo, A. N., et al. 2020, *MNRAS*, 492, 2872
- Yurchenko, S. N., Barber, R. J., & Tennyson, J. 2011, *MNRAS*, 413, 1828
- Yurchenko, S. N., & Tennyson, J. 2014, *MNRAS*, 440, 1649
- Zingales, T., & Waldmann, I. P. 2018, *AJ*, 156, 268