Toward a multidimensional analysis of transmission spectroscopy

Aurélien Falco, Tiziano Zingales, William Pluriel, Jérémy Leconte

A&A 658, A41 (2022)
https://doi.org/10.1051/0004-6361/202141940

Astronomy & Astrophysics

Toward a multidimensional analysis of transmission spectroscopy

I. Computation of transmission spectra using a 1D, 2D, or 3D atmosphere structure

Aurélien Falco[1], Tiziano Zingales[1,2], William Pluriel[1], and Jérémy Leconte[1]

[1] Laboratoire d'Astrophysique de Bordeaux, Univ. Bordeaux, CNRS, B18N, allée Geoffroy Saint-Hilaire, 33615 Pessac, France
e-mail: aurelien.falco@u-bordeaux.fr
[2] Università di Padova, Dipartimento di Astronomia, vicolo dell'Osservatorio 3, 35122 Padova, Italy

Received 3 August 2021 / Accepted 13 October 2021

ABSTRACT

Considering the relatively high precision that will be reached by future observatories, it has recently become clear that one dimensional (1D) atmospheric models, in which the atmospheric temperature and composition of a planet are considered to vary only in the vertical, will be unable to represent exoplanetary transmission spectra with a sufficient accuracy. This is particularly true for warm to (ultra-) hot exoplanets because the atmosphere is unable to redistribute all the energy deposited on the dayside, creating a strong thermal and often compositional dichotomy on the planet. This situation is exacerbated by transmission spectroscopy, which probes the terminator region. This is the most heterogeneous region of the atmosphere. However, if being able to compute transmission spectra from 3D atmospheric structures (from a global climate model, e.g.) is necessary to predict realistic observables, it is too computationally expensive to be used in a data inversion framework. For this reason, there is a need for a medium-complexity 2D approach that captures the most salient features of the 3D model in a sufficiently fast implementation. With this in mind, we present a new open-source documented version of Pytmosph3R that handles the computation of transmission spectra for atmospheres with up to three spatial dimensions and can account for time variability. Taking the example of an ultrahot Jupiter, we illustrate how the changing orientation of the planet during the transit can allow us to probe the horizontal variations in the atmosphere. We further implement our algorithm in TauREx to allow the community to perform 2D retrievals. We describe our extensive cross-validation benchmarks and discuss the accuracy and numerical performance of each model.

Key words. planets and satellites: atmospheres – methods: numerical – techniques: spectroscopic – radiative transfer – planets and satellites: general

1. Introduction

Transmission spectroscopy is a powerful method for studying exoplanetary atmospheres. During the past decade, space missions and ground-based surveys have provided numerous data on exoplanetary atmospheres (Tinetti et al. 2007; Tsiaras et al. 2018; Seidel et al. 2019; Mikal-Evans et al. 2020).

In addition, new space telescopes will be launched in the current decade (*James Webb* Space Telescope (JWST) and Ariel) whose goals will not only be the detection of molecular features, but also the estimation, with greater accuracy, of the atmospheric molecular abundances. However, a higher precision in the observation must be followed by an improvement of theoretical and statistical tools.

Several global climate models (GCM) are available (Showman & Guillot 2002; Showman et al. 2008; Menou & Rauscher 2009; Wordsworth et al. 2011; Leconte et al. 2013a; Charnay et al. 2015; Kataria et al. 2016; Drummond et al. 2016; Tan & Komacek 2019) that can describe very complex 3D atmospheric structures and might in principle give us deep insight into the physics of a planetary atmosphere. GCM simulations are a useful tool for characterizing from a theoretical point of view the chemical composition of an atmosphere and its physical properties (Showman et al. 2008; Leconte et al. 2013b; Guerlet et al. 2014; Venot et al. 2014; Parmentier et al. 2018). These simulations can also help validate the parameters inferred by retrieval procedures (Irwin et al. 2008; Al-Refaie et al. 2021; Mollière et al. 2019). Finally, it is also possible to use the 3D structure of these simulations to compute transmission spectra, as discussed in Caldas et al. (2019).

However, the computational cost of a 3D GCM simulation is very high. Their usage as forward models in Markov chain Monte Carlo (MCMC) retrieval procedures is therefore not currently considered. If we were to simplify the parameterization of the 3D structure of the planet (to avoid the cost of the full GCM simulation), then the question of the parameters and their number would also affect the computational cost of the retrieval. Moreover, retrieving too many parameters might be an issue: it would create many degeneracies in which the information would be lost, which would make the results difficult to interpret.

Bayesian retrieval codes include a critical trade-off between the precision of a model and its computational cost (Al-Refaie et al. 2021; Waldmann et al. 2015a,b; Line et al. 2013; Irwin et al. 2008). To increase the reliability of the solution given with a Bayesian approach, it is crucial to be able to quickly generate a forward model, without sacrificing meaningful physical phenomena that could lead to a significant spectral contribution. So far, models that were used to infer atmospheric parameters in

transmission or emission spectroscopy within Bayesian frameworks have mainly relied on a one-dimensional structure. The use of such models was partly justified by the low precision in the available observational data (Stevenson et al. 2014; Line et al. 2016; Tsiaras et al. 2018; Edwards et al. 2020; Pluriel et al. 2020a). In some cases, 1D models are a good approximation because the features in the transmission spectra come from a thin annulus around the limb, so that the region probed is almost homogeneous (Barstow et al. 2017; Tsiaras et al. 2018; Guilluy et al. 2021; Swain et al. 2021).

However, 1D atmospheric models will hardly explain the spectral shapes detected with the new generation of instruments because they reveal physical and chemical effects due to the 3D geometry of the atmosphere (Caldas et al. 2019; Changeat et al. 2019; MacDonald et al. 2020; Pluriel et al. 2020b). In particular, Pluriel et al. (2020b) showed that the atmospheric parameters of ultra hot Jupiters retrieved using 1D models can be biased: the solution that fits the observation best could be very different from the reality. In the case of ultra hot Jupiters, 1D retrieval codes cannot fit transmission spectra well because of the significant day to night thermal and chemical dichotomy. In these atmospheres, the 1D vertical assumption is no longer valid because the region that is probed extends significantly across the limb on both the day- and the nightside of the planet. The transmission spectra carry the information of the absence of water on the dayside (and its presence in the colder nightside) due to the thermal dissociation, and the presence of strong CO features (which does not dissociate because of its stronger triple bond; Lodders & Fegley 2002). A 1D retrieval will try to fit the water on the nightside (where it is not dissociated) and retrieves a colder temperature. It compensates for this low temperature by overestimating CO to try to fit the CO features.

From this observation, the effect of the large difference between a hot temperature on the dayside and a cold temperature on the nightside may be expressed with a simpler model that is composed of only two dimensions: (1) a vertical dimension, and (2) an angular dimension following the star-observer axis. Nevertheless, there are two caveats to keep in mind for the transition from 1D to 2D retrievals. First, the computational time needed to converge using MCMC or nested sampling would be increased. Second, and more importantly, we need to be aware that increasing the number of parameters in these retrieval methods could bring more degeneracy than information because the parameter space that is to be explored becomes larger. Therefore we need a 2D geometry with the simplest parameters space, so that we can reproduce the 2D effects of the atmosphere without adding too many parameters that are to be retrieved.

In this first paper of a series, we present our general method for computing transmission spectra using atmospheric simulations with a different number of spatial dimensions and validate the numerical tools that we have developed for this purpose. We first introduce a new open-source documented version of Pytmosph3R (Caldas et al. 2019) that computes transmission spectra for atmospheres with up to three spatial dimensions. The code is very flexible and can use 3D time-varying atmospheric structures from a GCM as well as simpler, parameterized 1D or 2D structures. Section 5 shows some application examples, highlighting the benefits of realistically describing the complex structure of the atmosphere. To allow our approach to be used in a retrieval framework, we have implemented our 2D algorithm in TauREx (Waldmann et al. 2015a; Al-Refaie et al. 2021; Changeat et al. 2019). We then demonstrate that our implementation is sufficiently fast to allow converging on a realistic retrieval solution in a reasonable amount of time.

2. Model description

We discuss here the computation of transmission spectra in the case of 3D, 2D, or 1D simulations. A simple representation of each model is shown in Fig. 1. An example of a 3D GCM simulation, representing WASP-121b, is included. It shows that the higher temperature on the dayside (on the left) affects the scale height, enlarging the atmosphere. We validate and study the performance of each configuration in Sects. 3 and 4.

2.1. Three-dimensional case

The computation of the optical depth, and eventually the transmission spectrum in the case of 3D simulations, has already been discussed in Caldas et al. (2019). We reintroduce the method and notations needed hereafter in the context of a new version of the implementation, `Pytmosph3R 2.0`, which is more robust and user friendly, of which the documentation is available online[1].

Global climate model simulations such as the LMDZ GCM (Wordsworth et al. 2011) provide a description of physical properties such as pressure, temperature, and volume-mixing ratios of absorbing or scattering molecules and aerosols in a three-dimensional grid. The vertical dimension of this grid may rely on pressure levels, for example, while the horizontal grid relies on latitude and longitude coordinates. The number of pressure levels is noted $N_{\mathrm{p,lay}}$, the number of latitudes N_{lat}, and the number of longitudes N_{lon}.

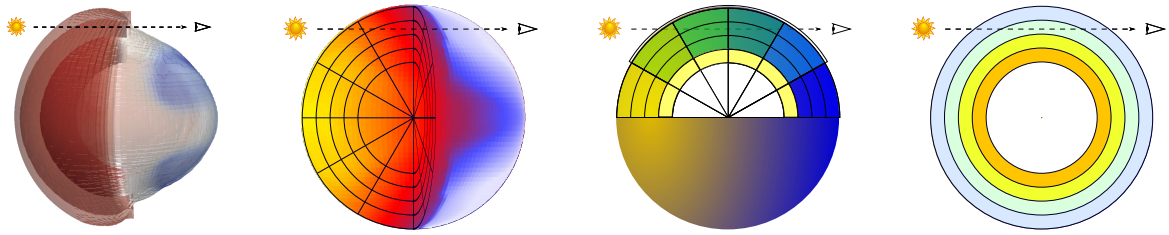
Pytmosph3R 2.0 offers several options to compute the volume-mixing ratio of all gases in each cell of the atmospheric grid: (1) it can extract this information directly from the input simulation when it is present, (2) it can interpolate from a separate table providing the mixing ratios on a pressure and temperature grid (e.g., when thermodynamical equilibrium is assumed), (3) it can use analytical formulae (e.g., those of Parmentier et al. 2018 where thermal dissociation of key molecules is accounted for), (4) it can call a chemistry module like FastChem (Stock et al. 2018) to compute the abundances on the fly, and (5) the users can specify their own constant mixing ratios. Other types of chemistry may be easily implemented by the users to adjust to the context of the simulation.

As discussed in Caldas et al. (2019), because (i) isobaric surfaces are generally not iso-altitude surfaces and (ii) altitude is the key variable in transmission geometry, the input simulation needs to be interpolated into a (radius, latitude, longitude) spherical coordinate system. The vertical axis of this new grid is discretized in $N_{z,\mathrm{lay}}$ layers delimited by $N_{z,\mathrm{lay}} + 1 \equiv N_{\mathrm{lev}}$ points called levels. The number of latitudes and longitudes of the new grid is identical to that of the input grid. The coordinates of the points in this new system are noted $(\rho, \varphi_0, \lambda_0)$.

The user can specify the position of the observer in this latitude-longitude system $(\varphi_{0,\mathrm{obs}}, \lambda_{0,\mathrm{obs}})$. This can be used to study the spectral variations caused by the rotation of the planet during the transit, as discussed in Sect. 5.

The optical depth is computed for a set of rays that are parallel to the planet-observer axis. As described extensively in Caldas et al. (2019), each ray is uniquely defined by its intersection with the plane normal to the planet-observer axis and passing through the planet center (hereafter, the plane of the sky). In this plane, the rays are arranged on a polar grid with N_{r} points along the impact parameter axis that are uniformly distributed between the surface of the planet and the top of the

[1] http://perso.astrophy.u-bordeaux.fr/~jleconte/pytmosph3r-doc/index.html



(a) WASP-121b (GCM) (b) 3D (c) 2D (d) 1D

Fig. 1. Visual representation of the dimensions of the models we considered using an equatorial view seen from the east. The colors of (*a*) indicate the temperature (redder is hotter) at isobar levels. The higher temperature on the dayside (on the left) affects the scale height, enlarging the atmosphere. The 3D model (*b*) follows a spherical (radius, latitude, longitude) coordinate system, while the 2D model (*c*) uses a polar grid whose the radial axis is the altitude and whose angular axis follows the solar altitude angle (which incidentally is also the angle of the zenith of the considered point with the zenith of the terminator). The 1D model (*d*) simply relies on the altitude.

atmosphere, and N_θ points along the azimuthal coordinate that goes around the limb of the planet. Even though the number of radial points N_r in this grid of rays may be different from the number of $N_{z,\mathrm{lay}}$ by definition, they are ultimately connected. Adding too many rays per layers does not increase the precision of the model because the information contained in the simulation does not increase, while too few rays would mean to lose some of this information. We use $N_\mathrm{r} = N_{z,\mathrm{lay}}$ in this paper and discuss the position of the rays in the layers in Sect. 3.2.

This new version relies on `Exo_k` (Leconte 2021) for the interpolation of the opacities (including regular molecular and atomic transitions, collision-induced absorptions, Rayleigh scattering by the gas, and Mie scattering by aerosols). As a result, the user can use both high-resolution cross sections or correlated-*k* coefficient tables.

2.2. Two-dimensional case

To reduce the cost of the computation of a transmission spectrum, we may simplify the simulation by assuming symmetry around the star-observer axis. We may thus reduce the simulation over a 2D grid that includes this axis. This allows us to describe the day to night temperature differences of certain types of planets such as ultra hot Jupiters with a lower cost than a full 3D simulation, and with a better precision than 1D models (Pluriel et al. 2020b). Because this model is entirely new (although it is based on the same method as the 3D model), we describe the details of its computation here.

2.2.1. Input data

The two-dimensional grid that we consider in this paper is centered on the planet center and contains the star-observer axis. Because atmospheric flows tend to follow isobars, the vertical dimension follows pressure levels, as is usually done in GCM simulations. The angular dimension follows the solar altitude angle, and the star is placed in the equatorial plane.

This 2D model could be applied on a complex 2D temperature, but we rely on the following thermal structure in this paper. The temperature is defined for every level and angle (i, α^*) using

$$\begin{cases} P > P_\mathrm{iso}, & T = T_\mathrm{deep}, \\ P < P_\mathrm{iso}, \begin{cases} 2\alpha^* \geq \beta, & T = T_\mathrm{day}, \\ 2\alpha^* \leq -\beta, & T = T_\mathrm{night}, \\ -\beta < 2\alpha^* < \beta, & T = T_\mathrm{night} \\ & \quad + (T_\mathrm{day} - T_\mathrm{night}) \frac{\alpha^* + \beta/2}{\beta}, \end{cases} \end{cases} \tag{1}$$

where T_day, T_night are scalar parameters that define temperatures of the day- and nightside. The β angle defines the area around the terminator, where the temperature decreases linearly from T_day to T_night. T_deep defines the isothermal structure of the atmosphere for pressures higher than P_iso. This simple parameterization of the temperature structure is inspired by GCM simulations of hot and ultra hot Jupiters (Showman et al. 2015; Parmentier et al. 2018; Tan & Komacek 2019) where the models can be approximated according to Eq. (1). More particularly, Pluriel et al. (2020b) have shown that this 2D representation could approximate ultra hot Jupiters with a satisfying precision.

Like in the 3D model (see Sect. 2.1), the chemical composition of the gas can parameterized by the user or computed using more complex chemical models.

2.2.2. Interpolation into an altitude-based polar grid

To compute the intersections of the rays with the atmospheric grid (discussed in Sect. 2.2.3), a regular geometric grid is to be preferred. Because every grid point in our 2D input grid can have different physical and chemical properties (pressures, temperatures, abundances, etc.) and hence different scale-heights, however, the altitude of the nth layer will change from one column (parameterized by the angle α^*) to the next, as illustrated by Fig. 2.

The altitude can be calculated from the hydrostatic equilibrium through the hypsometric equation,

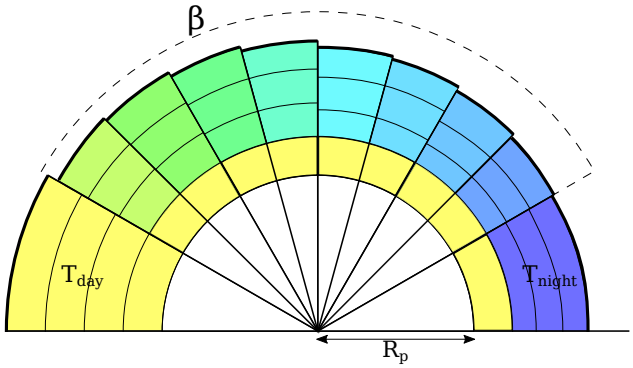
$$\mathrm{d}p = -\rho \cdot g \cdot \mathrm{d}z, \tag{2}$$

$$z_{i+1} = z_i + \frac{M \cdot R \cdot T_i}{g(z_i)} \cdot \ln\left(\frac{P_i}{P_{i+1}}\right), \tag{3}$$

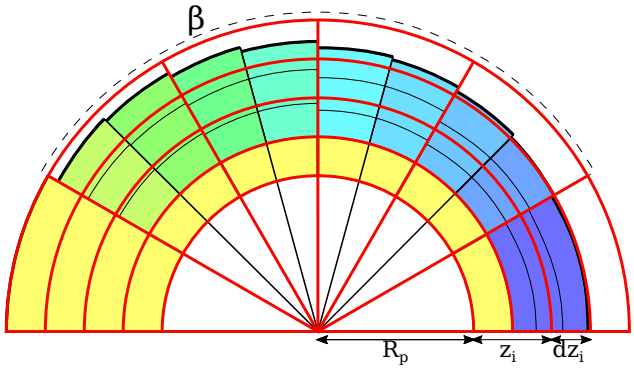
where ρ is the mass density, M is the molar mass of the gas, R is the universal gas constant, and $g(z_i)$ is the gravity at altitude z_i. From this equation, we can infer that a difference of temperature in the angular dimension will indeed affect the scale height.

With the altitude computed in the input grid, we can construct a new grid based on the altitude, as is represented in Fig. 2b, in a linear discretization of N_lev points that define $N_{z,\mathrm{lay}} = N_\mathrm{lev} - 1$ layers, up to a maximum altitude of our choice. The data in this new polar grid can be interpolated using a logarithmic interpolation for the pressure and a linear interpolation for the temperature and chemistry. This new grid facilitates computing the intersection points between the rays and the grid, as we discuss in Sect. 2.2.3.

The number of slices or angles in the 2D model is N_α, of which $N_\alpha - 2$ discretize the angle between $-\beta/2$ and $\beta/2$. The



(a) Input grid. The pressure levels separating the $N_{p,\mathrm{lay}}$ layers are identical in each column. Because of the temperature and composition differences, the altitude of a layer changes from one column to the next.



(b) Altitude-based 2D grid (in red) with $N_{z,\mathrm{lay}}$ layers superimposed on the input grid. In this example, we have $N_{z,\mathrm{lay}} = N_{p,\mathrm{lay}}$, but this is not a requirement.

Fig. 2. Illustration of the two grids used in the model. On the top, the pressure-based input grid (*a*). On the bottom, the regular altitude grid (*b*). The location of each of the N_α columns is parameterized by its solar elevation angle, α^*, equal to 90° at the substellar point, 0° at the terminator, and −90° at the subobserver point. β is the angle over which the atmosphere transitions from dayside to nightside temperatures. R_p is the planetary radius at the bottom of the grid. z_i is the altitude of level i, and dz_i is the thickness of layer i between levels i and $i+1$.

first and last angular slices account for the day- and nightside (not subdivided as they are uniform). The points in this 2D altitude-based coordinate system are noted hereafter (ρ, α).

2.2.3. Intersections of the rays with the altitude grid

In this model, we consider $N_\mathrm{r} = N_{z,\mathrm{lay}}$ rays. A ray may be described using its impact parameter r, defined as the normal distance between the ray and the center of the planet (see Fig. 3). We chose the grid of impact parameters so that the rays cross the plane of the sky (or equivalently here, the terminator of the planet) exactly at the center of the atmospheric layers. To compute the optical depth (Eq. (6)) of a ray r, we need the length of each segment of the ray crossing individual cells of the altitude grid, as well as the physical properties of these cells.

The coordinates of the intersections points of the ray with the grid may be computed using

$$r = \rho \cos(\alpha^*). \tag{4}$$

This returns the angle of intersection when applied on the N_lev levels of radius ρ (duplicated for positive and negative values),

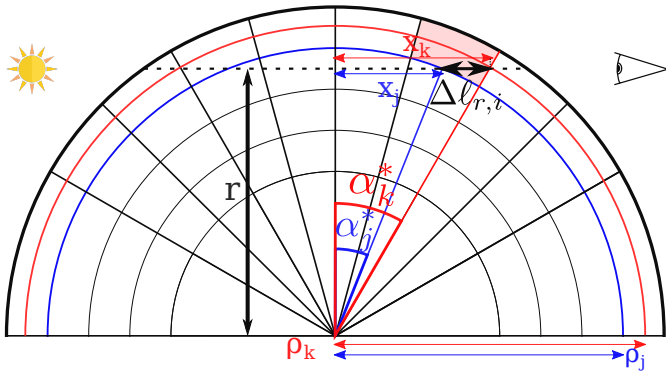


Fig. 3. Intersection of a ray with the 2D grid. An example of intersections for one level and one angle is highlighted by points j and k at coordinates (ρ_j, α^*_j) and (ρ_k, α^*_k), respectively. The length $\Delta\ell_{r,i}$ of the corresponding segment is deduced from the computation of the distances x_j and x_k of both points to the terminator. The coordinates of the segment must also be computed to obtain the physical properties of the cell.

and the radius of intersection when applied on the $N_\alpha - 1$ solar elevation angles α^* that separate the slices of the grid. The two types of intersections (with levels and with angles) are visually represented in blue and red (or indices j and k), respectively, in Fig. 3.

The intersection points of a given ray are then sorted using their angular coordinate. The length of each segment (as highlighted in Fig. 3) is equal to the subtraction of the distance of both extremities of the segment (j and k in our example) to the terminator. The distance of a point k to the terminator is given by

$$\mathbf{x_k^2} = \rho_\mathbf{k}^2 - \mathbf{r^2}. \tag{5}$$

This equation returns two solutions, corresponding to a point before and a point after the terminator (following the sign of the solution). By convention, a negative distance corresponds to a point on the dayside.

We note $\Delta\ell_{r,i}$ the distance of a segment i between two consecutive intersection points of a ray r.

2.2.4. Optical depth

Based on the previous definitions, the optical depth of each ray r at a wavelength λ can be computed using Eq. (6),

$$\tau_r^\lambda = \sum_i \frac{P_{r,i}}{k_\mathrm{B} T_{r,i}} \left(\sum_{m=1}^{N_\mathrm{gas}} \chi_{m,r,i} \sigma_{m,\lambda} + \sum_{j=1}^{N_\mathrm{con}} k_{mie,j} \right) \Delta\ell_{r,i}, \tag{6}$$

where $P_{r,i}$ and $T_{r,i}$ are the pressure and temperatures of the cell corresponding to the segment i of the ray r, k_B is the Boltzmann constant, $\chi_{m,r,i}$ is the volume-mixing ratio of the mth molecule, $\sigma_{m,\lambda}$ is the total cross-section of Rayleigh scattering and molecular and continuum absorptions, and finally, $k_{mie,j}$ is the extinction coefficient associated with the Mie scattering for the jth aerosol. N_gas is the number of molecules, and N_con is the number of aerosols. The cross-sections and Mie coefficients can be computed by `Exo_k` (Leconte 2021).

2.2.5. Transmittance map and spectrum

A transmittance map may be computed from the optical depth through

$$\mathcal{T}_r^\lambda = e^{-\tau_r^\lambda}. \tag{7}$$

(a) Rays at levels (b) Rays at mid-layers

Fig. 4. Schematic of the two possible methods for positioning the rays (dashed lines). In *panel a*, the rays are tangent to the pressure levels (solid arcs), as done in TauREx 3.0. In *panel b*, the rays pass the middle of the layers, as done in Pytmosph3R and throughout this paper. A comparison of the accuracy of these two methods is shown in Fig. 5.

In the case of a homogeneous stellar disk, the relative dimming of the stellar flux is given by

$$\Delta_\lambda = \frac{\pi {R_\mathrm{p}}^2 + \sum_r \left(1 - e^{-\tau_r^\lambda}\right) S_r}{\pi {R_\mathrm{s}}^2}, \tag{8}$$

with $S_r = 2\pi(r + \frac{\mathrm{d}r}{2})\,\mathrm{d}r$. R_p is the radius of the planet, R_s is the radius of the star, and $\mathrm{d}r$ is the distance between two consecutive r. The computation of the transmission spectrum is complete.

2.3. One-dimensional case

To simplify the simulation even further, we can approximate the model using only one dimension: the vertical axis. This approximation has been extensively used in the past and will serve as a reference.

In this case, the temperature and composition are horizontally uniform, so that isobaric and iso-altitude surfaces are equivalent and a single grid can be used. The steps required to compute the transmission spectrum are then exactly identical to those in the 2D model, except that we do not need to use Eq. (4) because there is no slice in this model.

However, although there is consensus about the theoretical equations for computing the transmission spectrum of a horizontally uniform atmosphere, some differences in the numerical algorithm can lead to significant quantitative differences in the results. In particular, there are two main options when the impact parameters of the rays are chosen, as shown in Fig. 4. The first option (used in TauREx 3.0) is to place the rays so that their impact parameter r is equal to the radius of the levels ρ. The second option is to place the rays so that their impact parameter r is equal to the radius of the midpoints of the layers $\rho + \frac{\mathrm{d}\rho}{2}$.

As we show in Sect. 3.2, the second option (rays at midlayers) appears to results in a faster convergence of the numerical scheme. We therefore implemented this method in our 2D and 3D codes.

3. Model accuracy

In this section, we use a series of test cases of increasing complexity to cross-validate our different implementations. As a byproduct, this will allow us to test the accuracy of our algorithms for various grid resolutions. The efficiency in terms of computing time is discussed in Sect. 4.

We emphasize that our test cases are based on the atmosphere of an ultra hot Jupiter, WASP-121b, where hydrogen partially dissociates so that the scale height is extremely large. In addition, the planet is extremely inflated. All these factors add up to increase the atmospheric signal up to several thousand ppms. As a result, this is a particularly stringent test for models, which explains the relatively high resolution needed to achieve a given precision. A lower resolution could be used for cooler objects.

Table 1. Characteristics of the isothermal case, including the abundances (volume mixing ratios) of each molecule.

Planet radius	1.807 R_J
Surface gravity	9.39 m s^{-2}
Temperature	2500 K
[H_2O]	5.01×10^{-4}
[CO]	4.4×10^{-4}
[H_2]	0.740
[He]	0.259

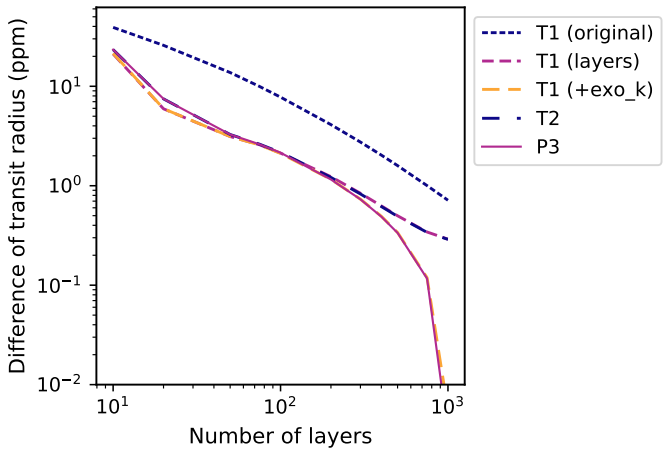


Fig. 5. Average of the absolute difference (in ppm) between each model and the converged solution of Pytmosph3R (with 1000 layers) as a function of the number of layers. P3 stands for Pytmosph3R, and T1 and T2 for TauREx 1D and 2D, respectively. The original implementation of TauREx (denoted "T1 (original)", with rays that are tangent to levels) is included.

3.1. Experimental setup

The 1D model is based on the TauREx 3 implementation (Al-Refaie et al. 2021). We developed the 2D model in the same framework, extending each 1D data profile to a 2D alternative, and using Eq. (1) for the temperature map. The 3D model, developed in a first version by Caldas et al. (2019) was reimplemented here in a second version, `Pytmosph3R 2.0` [1]. All models are coded in Python 3, with computationally intensive operations relying on numba (Lam et al. 2015). The machines on which the numerical experiments were performed are Intel®Xeon®Gold 6138 CPUs at 2.00 GHz, with 128 GB of RAM.

3.2. Validation for the 1D test cases

A test case that can be reproduced by each model is the case of isothermal atmospheres. We therefore study here an isothermal example of a hot Jupiter planet with the physical properties listed in Table 1. The spectrum is generated for 39 124 wavelengths from 0.3 to 15 µm.

We have ensured that all codes indeed converge toward a solution when the number of layers is increased, as is shown by Fig. 4 of Caldas et al. (2019) for the previous version of Pytmosph3R.

Figure 5 shows the convergence of the TauREx and Pytmosph3R 2.0 with respect to a converged Pytmosph3R (1000 layers).

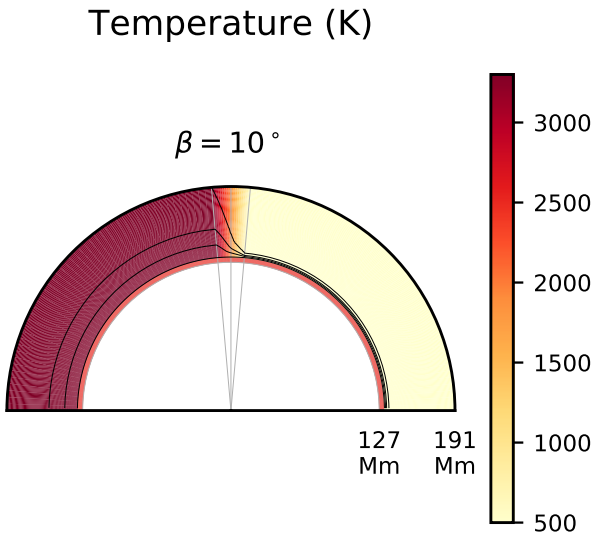
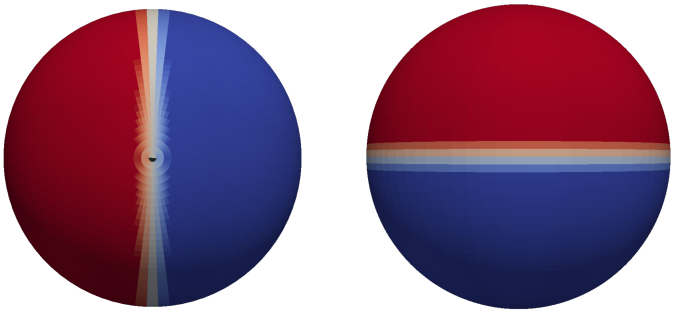


Fig. 6. Temperature map for our reference 2D test case. The substellar point is on the left. $T_{\rm day} = 3300$ K while $T_{\rm night} = 500$ K and $T_{\rm deep} = 2500$ K, with $\beta = 10°$. The black lines correspond to isobar levels. The altitude is in Mm, i.e., thousands of km.



(a) $\varphi_\star = 0°$ (b) $\varphi_\star = 90°$

Fig. 7. Discrete horizontal temperature maps at high altitude computed using Eq. (1) with 80/80 latitude and longitude grid points with two different choices of grid orientation (redder is hotter). These maps are visualized in 3D through ParaView (Ahrens et al. 2005). (*a*) The star is located in the equatorial plane (the planet is seen from the pole, with the star on the left). (*b*) The star is located at the pole (the star and the north pole are located at the top). In this idealized setup, choice b allows us to take advantage of the symmetries of the system and to use only one longitude point for our grid, which considerably speeds up the computation.

To accelerate the convergence of TauREx 1D, we implemented a new version that places the rays at the center of the layers ("T1 (layers)") and not at the levels. The original implementation of TauREx is given by the curve "T1 (original)". See Fig. 4 for a visual representation of the two methods. This simple algorithmic change improves the model accuracy by a factor of 3 on average at no cost.

For more than ∼1000 layers, the difference between T1 (layers) and our reference seems to reach a plateau. When the interpolation of the opacities in TauREx is replaced by that of `Exo_k` (the one used in Pytmosph3R) in addition to the position of the rays at the center of the layers, we obtain the method "T1 (+exo_k)" for which the difference with the reference model continues to decrease as the number of layers increases. This shows that in the case of relatively hot giant planets, the opacity interpolation scheme can lead to errors of ∼0.3 ppm.

Overall, all (new) codes converge to a difference smaller than 1 ppm with a few hundred layers. From this point forward, all methods rely on `Exo_k` for the computation of the opacities and place the rays at the center of the layers.

For horizontally uniform atmospheres, the choice of the orientation of the longitude or latitude grid used in Pytmosph3R is arbitrary and should not affect the results. We verified that the output spectrum is independent of this choice down to machine precision, which further validates our implementation.

3.3. 2D test case

We now cross-validate TauREx 2D and Pytmosph3R using a 2D temperature structure that is symmetrical around the planet-observer axis (see Eq. (1)). The chosen temperature structure, shown in Fig. 6, has a large difference between the dayside and nightside temperature.

Our temperature structure does not directly depend on the longitude and latitude: it only depends on the solar elevation angle, which is given by

$$\sin\alpha^* = \sin\varphi_0 \sin\varphi_\star + \cos\varphi_0 \cos\varphi_\star \cos(\lambda_0 - \lambda_\star), \quad (9)$$

where (φ_0, λ_0) are the latitude and longitude of the current cell, and $(\varphi_\star, \lambda_\star)$ are the latitude and longitude of the substellar point. As a result, the choice of the grid orientation is arbitrary. In other words, the choice of the direction of the star in our arbitrary reference frame is a free parameter that should not affect the results (as long as we align star, planet, and observer).

However, as illustrated in Fig. 7, for a grid with a finite size, the orientation of the grid can slightly affect the way the temperature is represented in the model, and consequently, the resulting spectrum. We find that this effect is about 1 ppm. For the rest of this section, we choose to place the star at the pole ($\varphi_\star = 90°$) because it allows us to use only one longitude ($N_{\rm lon} = 1$) and one azimuthal angle ($N_\theta = 1$, see Sect. 2.1), which considerably speeds up computations.

We can now compare the convergence of TauREx 2D with that of Pytmosph3R. The results are shown in Fig. 8. This figure shows the convergence of TauREx with an increasing number of slices and Pytmosph3R with a increasing number of latitudes for an increasing number of layers. With an equivalent angular resolution (e.g., $N_\alpha = 10$ and $N_{\rm lat} = 180$, which leads to an angular width of 1° for each angular point), TauREx and Pytmosph3R follow a very similar trend. The models converge to a difference smaller than 1 ppm with a sufficiently high resolution.

3.4. 3D simulations

We study here a 3D GCM simulation based on WASP-121b (Parmentier et al. 2018; Pluriel et al. 2020b). One of the characteristics of this simulation is the strong dichotomy between the temperature on the dayside and that of the nightside. The simulation is shown in Fig. 1a from the east in the equatorial plane. Temperature maps in the equatorial plane and at high altitude are given in Fig. 9. This simulation also has a slight east-west asymmetry; the hottest point is shifted toward the east. This feature is most visible in Fig. 9. The chemistry is given by tables from Parmentier et al. (2018) and includes He, H_2, H, H_2O, CO, TiO, and VO.

Using this (heterogeneous) 3D GCM simulation, the number of latitudes $N_{\rm lat}$ and longitudes $N_{\rm lon}$ is fixed. The number of layers $N_{z,\rm lay}$ of the altitude grid may be chosen as different from

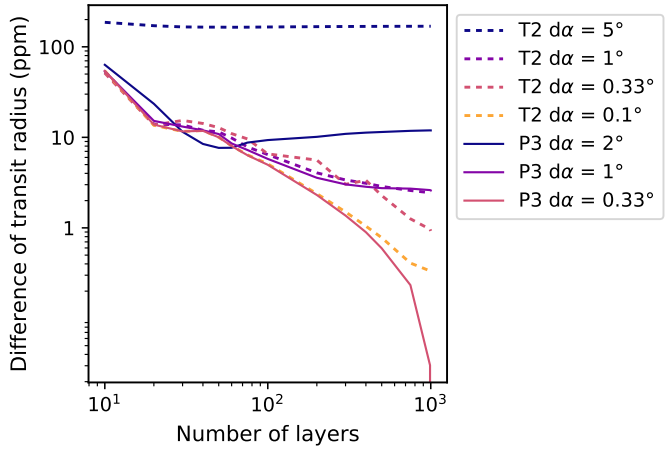


Fig. 8. Convergence of TauREx 2D and Pytmosph3R when the atmospheric grid resolution is increased. The Y-axis indicates the average of the absolute difference over all wavelengths considered between each model and Pytmosph3R ($N_{z,\mathrm{lay}}, N_{\mathrm{lat}}$) = (1000, 540). The X-axis indicates the number of layers $N_{z,\mathrm{lay}}$ in the model. The legend indicates the angular resolution of the model, dα, equal to β/N_α for TauREx (T2) and $180/N_{\mathrm{lat}}$ for Pytmosph3R (P3). For example, d$\alpha = 1°$ leads to $N_\alpha = 10$ and $N_{\mathrm{lat}} = 180$. Pytmosph3R was run with $\varphi_\star = 90°$ (see Fig. 7), so that only one longitude and one azimuthal angle are necessary, i.e., $N_{\mathrm{lon}} = N_\theta = 1$.

that of the input simulation. To simplify the number of parameters, we simply set this number to the number of radial points N_{r} in the polar grid of rays ($N_{\mathrm{r}} \times N_\theta$). To study the accuracy of the model, we can therefore change the number of rays, as we show in Fig. 10. The model tends to converge if we have enough radial and angular points to correspond to the GCM resolution, that is, at least 32 angular points, although we might apparently need to double the number of radial points to obtain better accuracy.

4. Performance and optimizations

We now study the performance of our models and ways to reduce the time and memory required for the computation of a transmission spectrum.

4.1. TauREx

TauREx is mainly used for its retrieval feature. We must therefore ensure that the forward model can be computed fast enough for the retrieval to be achieved in a reasonable amount of time for the user.

In its 1D variant, an important part of the arithmetic complexity of TauREx is the computation of the optical depth, which is related to the number of cells that the rays are passing through, and for which we must compute the formula of Eq. (6). The number of calculations in 1D leads to a complexity for the optical depth of

$$\mathcal{O}\left(\frac{N_{z,\mathrm{lay}}^2}{2} \cdot N_\lambda\right), \tag{10}$$

where N_λ is the number of wavelengths. As a result of the assumption that day and night are identical, a ray at the altitude of level i only intercepts the $N_{z,\mathrm{lay}} - i$ levels above i. With $N_{z,\mathrm{lay}}$ rays, the number of layers for which the optical depth has to be

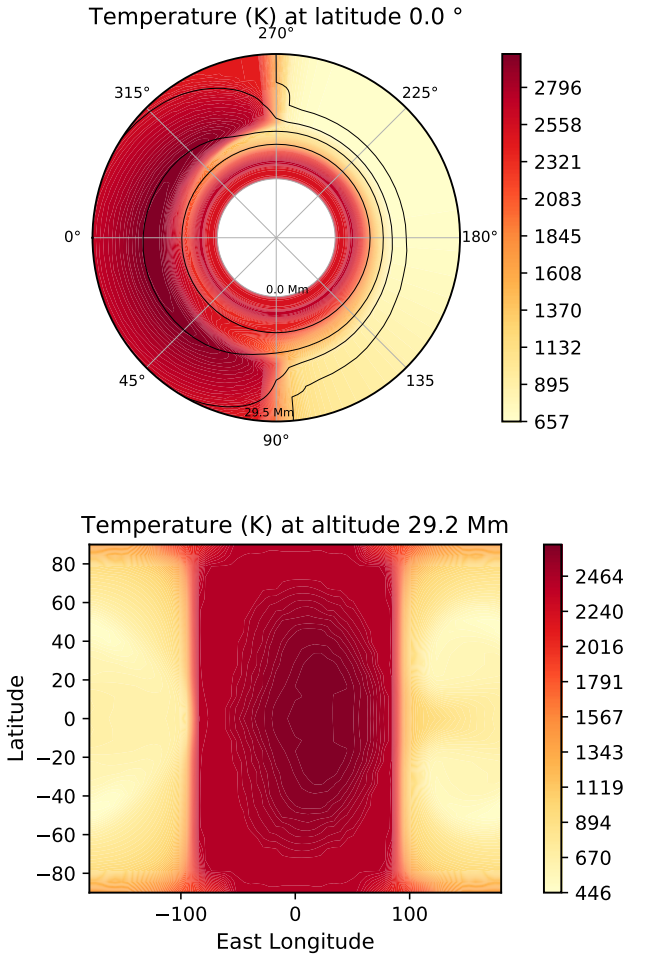


Fig. 9. Temperature maps of WASP-121b. The *top* map shows an equatorial slice. The planet radius is divided by 10 for visual reasons. The black lines correspond to different isobar levels. The *bottom* map shows a slice at high altitude (29.2 Mm, i.e., 29 200 km). The hottest point is slightly shifted to the east limb, i.e., the trailing limb.

computed is thus $\sum_{i=1}^{N_{z,\mathrm{lay}}} i = \frac{N_{z,\mathrm{lay}}(N_{z,\mathrm{lay}}+1)}{2}$. We focus here on molecular absorption and leave Rayleigh scattering and the continuum absorption out of this study.

In the 2D version of the model, however, we must account for the day- and nightside (therefore doubling the number of computations of Eq. (10)), as well as all the intersections of the layers with the slices in the linear transition around the terminator. This leads to a complexity for the 2D optical depth of

$$\mathcal{O}\left(\left(N_{z,\mathrm{lay}}^2 + N_{z,\mathrm{lay}} \cdot N_\alpha\right) \cdot N_\lambda\right). \tag{11}$$

We show in the results that the main calculations in 2D are done for the number of opacities $\chi_{\mathrm{m},\lambda}[\alpha, r]$ that are to be interpolated. The arithmetic complexity of computing the opacities is

$$\mathcal{O}\left(C \cdot N_{z,\mathrm{lay}} \cdot N_\alpha \cdot N_\lambda \cdot N_{\mathrm{mol}}\right), \tag{12}$$

where C is the cost of computing one interpolation, and N_{mol} is the number of molecules. Incidentally, adding molecules will increase the cost of the interpolation of the opacities (Eq. (12)), but not of the optical depth itself (Eq. (11)). Equation (12) is also valid in 1D, but in this case, $N_\alpha = 1$.

To ensure that the model behaves as expected, we performed a series of measures following the method we used and the

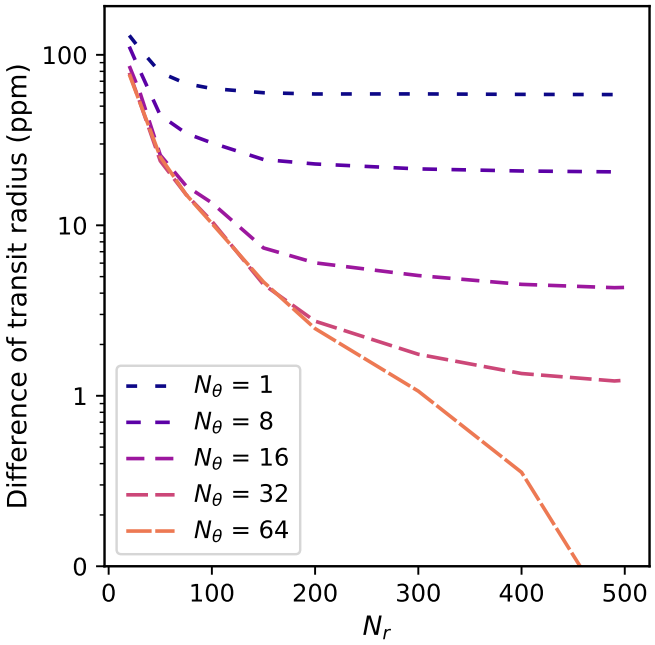


Fig. 10. Convergence of Pytmosph3R when the number of rays $[N_r \times N_\theta]$ crossing the atmosphere is increased, using as a reference point the converged model ($[N_r \times N_\theta] = [500 \times 64]$). The simulation is a GCM with an equilibrium temperature $T_{eq} = 2100$ K based on the characteristics of WASP-121b, and including TiO and VO. The number of cells in this simulation is $(N_{z,lay}, N_{lat}, N_{lon}) = (100, 32, 64)$.

Table 2. Average time (s) to run one TauREx model, considering the molecular absorption only (no Rayleigh scattering or CIA) of four molecules on 39 124 wavelengths from 0.3 to 15 µm.

		N_α (2D)			
$N_{z,lay}$	1D	2	10	20	30
100	0.26	0.52	1.20	1.99	2.80
200	0.78	1.73	3.05	4.72	6.32

dimensions of the atmospheric grid that were considered. These measures are gathered in Table 2. Because the whole calculation scales linearly with the number of wavelength points (N_λ) at which the model is run, it is set to a constant value (39 124 wavelength points) for this study.

There are multiple interesting trends that we can observe in this table. First, we can observe that there is a factor of 2 between the 1D and the 2D model with two slices because there is a day- and a nightside. Second, the computational time seems to increase less than quadratically with respect to the number of layers in 1D, showing that there are enough molecules to make the interpolation of the opacities a significant part of the computations. Third, for a large number of slices, the time also increases linearly with the number of slices, which means that the cost of one interpolation, C, in Eq. (12) and the number of molecules N_{mol} are large enough to make the interpolation of the opacities the dominant part of the computations. Increasing the number of layers will change this behavior, as the quadratic complexity of Eq. (11) will start to be dominant again.

A good compromise between accuracy and computational time seems to be running 2D models with 200 layers and 30 slices.

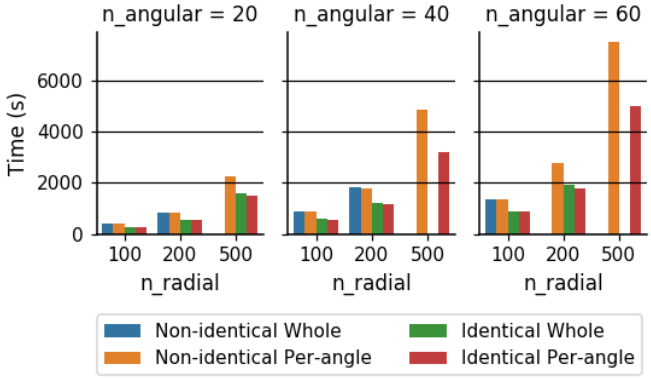


Fig. 11. Time (s) required by Pytmosph3R to compute the transmission spectrum of a 2D atmospheric grid defined with Eq. (1). Two methods are turned on or off: the Per-angle method (off is "Whole"), and the Identical method (off is "Non-identical"). The number of rays $N_r \times N_\theta =$ n_radial × n_angular also varies. Missing points have run out of memory.

4.2. Pytmosph3R

In 3D, the great majority of the calculations lies in the interpolation of the opacities. This interpolation scales with the number of cells for which we need to compute the opacity. To decrease the number of calculations, we can therefore first identify how the number of cells for which we need to compute an interpolation can be reduced. The first step is to realize that we do not need (and cannot afford in terms of time and memory) to compute the opacity of all $N_{z,lay} \times N_{lat} \times N_{lon}$ cells in the model. A large part of the cells is not crossed by any light ray, and therefore we do not need to compute their opacity.

A naive algorithm would simply iterate over all the rays and compute the opacity for each segment that is crossed by a ray. However, depending on the resolution of the model, many cells may be crossed by multiple rays, so that we can reuse the information from one ray to another. In addition, when we have the opacities of all segments of one ray, the optical depth of that ray can be computed, and the opacities may be discarded. In light of these two facts, we developed an algorithm that will group light rays together at each azimuthal angle θ. This means we can benefit from the reusability of opacities within that angle (along the radial axis) while being able to release the memory of all opacities when the optical depth of all rays within this angle have been computed. The memory will therefore almost be a constant throughout the execution of the program, if we consider each angle to be equivalent (this depends on the resolution of the grid). This method, which we refer to as "Per-angle" in the experiments, works quite well for every kind of problem (completely heterogeneous to isothermal).

However, the computational cost can be decreased further by making a few assumptions. Some atmospheric simulations may have a number of cells that have identical physical properties, for example, in the case of the 2D representation of Eq. (1). If this number is sufficiently large, we can also aggregate these cells together to further reduce the number of opacities that are to be computed. This method is referred to as "Identical" in the experiments.

We compare these two methods in Figs. 11 and 12 for a 2D problem defined with Eq. (1). These two figures show the computational time required for a model to run and its memory consumption peak, respectively.

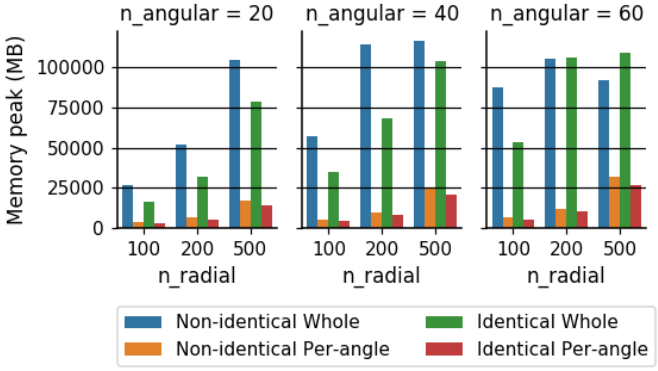


Fig. 12. Real memory peak (MB) of the corresponding points in Fig. 11. The machine has 128 GB of RAM, and the recorded peak is below that point for those that have run out of memory (missing points in Fig. 11).

The first observation we can make here is the effect of the Identical method, which reduces the time and memory complexity of the program by approximately one-third in this case. However, this is possible only due to the characteristics of the simulation, in which many cells are actually identical. We must emphasize that a completely heterogeneous simulation would not benefit at all from this method.

A second observation we can make is the drastic memory saving due to the Per-angle method. As we mentioned earlier, this method allows us to discard the opacities after each angle and keep the memory due to the opacities below a constant. However, other variables such as the transmittance (if needed) will still increase in size with respect to the number of angles.

In conclusion, the Per-angle method provides a drastic memory reduction, while the Identical should be used only for data that contain redundancies. A method to ensure that the computation will not run out of memory is also under study, as well as a parallel version.

5. Examples of applications

Pytmosph3R allows us to do more than just compute a more realistic spectrum from a static atmospheric structure. The transmission signal from a planet is expected to vary over time due to two main reasons: (1) because the atmospheric structure itself is variable, whether it is the temperature, the composition, or the distribution of clouds and hazes; (2) because the planet rotates during a transit (even when in synchronous rotation), showing us a slightly different cross section of its atmosphere. In this section, we illustrate these two effects.

5.1. Atmospheric variability

In a recent study, Charnay et al. (2021) investigated the dynamics of clouds on temperate mini-Neptunes, taking the example of K2-18b. They revealed that the abundance of clouds at the terminator was highly variable. To assess the effect of these clouds on the transmission spectrum, we ran Pytmosph3R on the climate model results at various time steps. Figure 13 shows transmittance maps calculated using Eq. (7). These maps provide more information than the sole transmission spectrum that can be obtained by their spatial integration (see Eq. (8)). Here we can use these maps to infer the effective fraction of the limb covered by clouds, as well as the difference in the effective absorption altitude for a clear versus cloudy atmsophere.

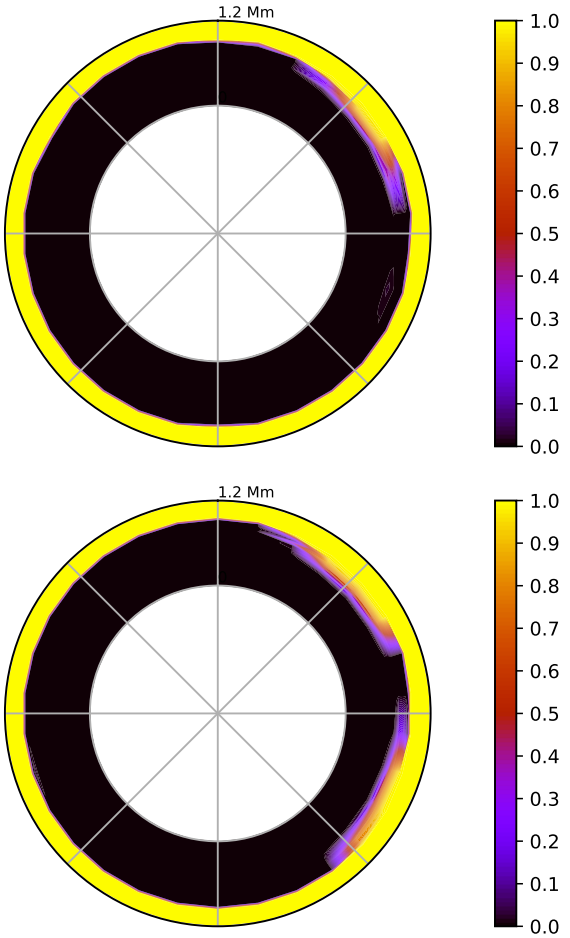


Fig. 13. Example of transmittance maps at two different times of a simulation of K2-18b (Charnay et al. 2021) at a wavelength of 0.6 µm from the observer's point of view. The atmosphere is transparent when the transmittance is equal to 1, and it is opaque when it is equal to 0. The absence of clouds is visible on the right side, which corresponds to the west limb. The atmosphere scale height has been multiplied by 10 for visual reasons. The altitude is in Mm, i.e., thousands of km.

As the two maps in Fig. 13 correspond to two time steps in the simulation, we can also follow the movement of aerosols and the variation of the cloud fraction over time.

As the climate model is expected to predict a realistic evolution of the clouds in time, this can allow us to quantify how the variability of the cloud structure during a single transit could affect the observed signal.

5.2. Rotation of the planet during transit

An interesting aspect of our 3D model is also that the geometry of the observation can be changed. When a planet passes in front of its star, it rotates slightly, showing the observer a phase that varies with time. This occurs even when the planet is in a tidally synchronized rotation. In this case, the star (and the terminator) remains fixed in the reference frame corotating with the planet, but the observer (and the limb it probes) is moving.

In the case of an asymmetric and heterogeneous 3D atmosphere, a change in the planet's phase angle implies that the light rays will not probe the same areas of the atmosphere. This therefore completely changes the associated transmittance map and the resulting transmission spectrum. Interestingly, this should

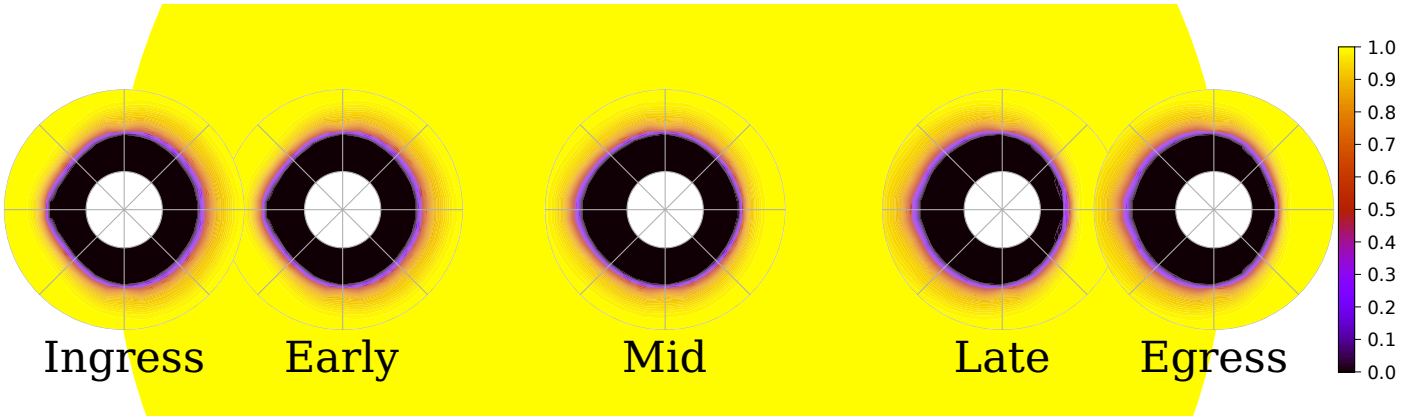


Fig. 14. Transmittance maps of WASP-121b at 0.6 μm for the five orbital phase angles whose spectra are shown in Fig. 15. The orbital phase angle is $\phi = -15°$ for ingress and $\phi = 15°$ for egress. For visual reasons, the planet atmosphere has been enlarged with respect to its radius, and the early and late transmittance maps are slightly shifted. Only half the planet covers the star at ingress and egress.

create asymmetries in the transit light-curve (Espinoza & Jones 2021).

While we are in the process of implementing a full-fledged light-curve generator, we wish to quantify in a simple way here how much the spectral transit depth might vary due to these effects.

To do this, we took the example of WASP-121b (Fig. 9; details of the simulation can be found in Sect. 3.4) and ran Pytmosph3R at five different phases during the transit, as illustrated in Fig. 14. The planet was considered to be tidally locked. This figure shows the transmittance maps at different stages of the transit at a wavelength of 0.6 μm. It gives an example of how transmittance maps may evolve during a transit, and which information we can retrieve from them. We only took the effect of the (synchronous) rotation of the planet between ingress and egress into account because we assumed a stationary atmosphere during the transit. The selected phases are listed below.

1. Ingress: half of the planet (west limb) is in front of the star. For the system parameters we used, this corresponds to an orbital phase angle of $\phi = -15°$. The center of the planet is located at the edge of the star.
2. Early: the planet has completed entering the transit (second contact; $\phi = -13°$).
3. Mid: the planet is at mid-transit, $\phi = 0°$.
4. Late: the planet is preparing to exit the transit, $\phi = 13°$.
5. Egress: the east limb is in front of the star. The planet is exiting the transit, $\phi = 15°$.

Figure 15 shows the relative transit depth (Eq. (8)) for each phase. To facilitate comparison, we removed the effect of stellar limb darkening, even though it will be accounted for when computing realistic light-curves. For the ingress and egress spectra, only one limb is in front of the star, so that we multiplied the covered area by two to facilitate comparison.

The figure shows that these transit depth variations are not negligible: as an example, the early spectrum is 110 ppm below mid-transit on average (with a minimum at −200 ppm in certain wavelength regions, especially in TiO/VO bands, i.e., from around 0.4 to 1 μm). The early and the late spectra follow similar patterns, including in the TiO/VO bands and higher wavelengths, indicating a general symmetry in the TiO/VO repartition. In our simulation, TiO and VO are mainly located around the terminator. The decrease in absorption in early and late spectra (with respect to mid-transit) is due to the rotation of the terminator disk, which shows the highest cross section at mid-transit.

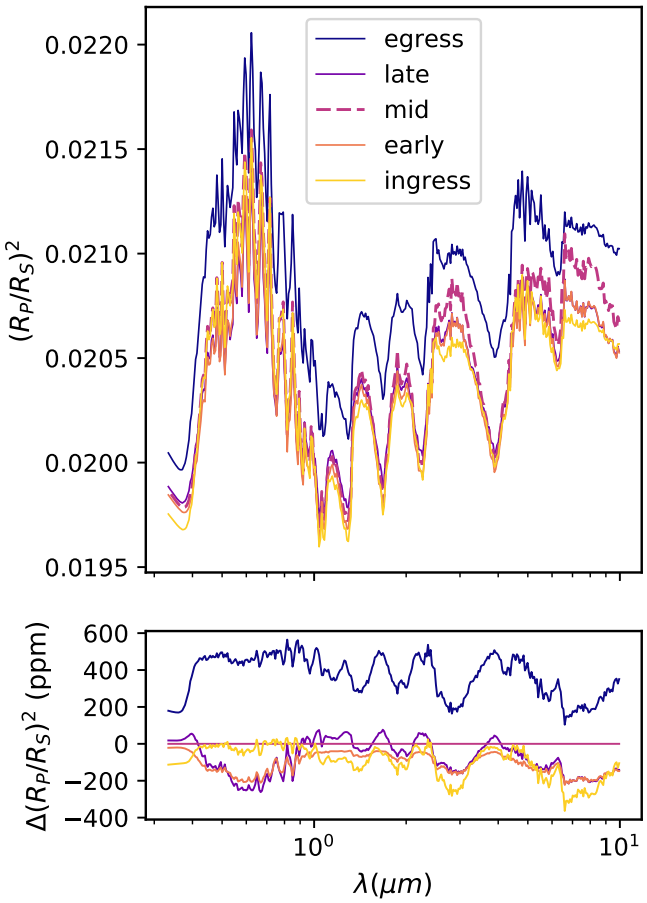


Fig. 15. Spectral variations of the transit depth of WASP-121b during a transit for the phases listed in Fig. 14. The *bottom plot* shows the difference between each spectrum and the mid-transit spectrum, taken as a reference.

As shown in Fig. 9, the temperature structure is not completely symmetrical because there is an eastward shift of the hottest region of about 20°, even though the day-night transition remains very sharp and symmetric. This eastward shift of the hot spot results in an east limb that is larger than the west one, which is visible in all transmittance maps. The elongation is less visible during ingress as the hot spot is behind the planet

from the observer, and this is exacerbated during egress as the spot is closer to the east limb.

During the early part of the transit, the limb is different from the terminator (rotated by $-\phi$), and we probe deeper in the dayside west of the planet and deeper in the nightside east of the planet. The situation is exactly reversed at the late position. Figure 14 shows that the transmittance map at the early position is mostly symmetrical because the asymmetry of the temperature map is compensated for by the rotation of the planet during its orbit. Then, moving to the late step, the reverse situation occurs: The temperature on the east limb is hotter, implying a greater scale height, but the west limb is colder, inducing a smaller scale height. The temperature eastward shift seems to lead to a difference smaller than 100 ppm (the largest spectral difference). The location of the molecules (around the terminator) decreases this difference further, and the two spectra are very similar in most parts (with an average difference of less than 40 ppm). Although the changes from the mid-transit to the early and the late transmittance maps are (longitudinally) reversed, these differences disappear when the transmittance maps are spatially integrated to generate the spectra (Eq. (8)).

For ingress and egress, only the planet and atmospheric half that is in front of the star (the west and east limbs, respectively) are considered for the integration of the transmittance into a spectrum. The rotation of the planet at egress means that the east limb of the planet is hotter (accentuated by the eastward shift of the temperature map), while the west limb is colder. Because the east side alone is considered, the scale height of the atmosphere is larger and we observe a larger effective radius (see Fig. 15). This leads to stronger spectral differences with an average of 370 ppm and peaks up to 560 ppm.

For the ingress, the eastward shift of the hottest region (see Fig. 9 and the orbital phase of the planet implies that the light crosses colder regions of the atmosphere on the west limb. This results in a smaller effective radius because of a smaller scale height.

The key points of this study are therefore threefold:

1. The rotation of the planet during transit results in variations of the transit depth of up to 300 ppm for a hot Jupiter such as WASP-121b[2]. It should therefore be detectable. The noise in observations from the HST currently reaches around 50 ppm and can be as low as 20 ppm at best. This noise could be lowered to 10 ppm or less with the upcoming JWST (Greene et al. 2016).
2. The most important differences are between ingress and egress (when only half the planet covers the star) and are mainly due to the asymmetry caused by the eastward shift of the hot spot.
3. Measuring these light-curve asymmetries would allow us to place constraints on the rotation of the planet and/or the direction of the hotspot shift without the need for a complete and expansive phase-curve.

5.3. Toward a time-domain analysis

As Pytmosph3R can simulate any position for the observer, it can provide spectra and transmittance maps at any position during a transit. The transmittance maps can be used to extract the part of the atmosphere (and planet) that covers the star, for example, during ingress and egress (see Sect. 5.2). This information can also be used in the future for a time-domain analysis of the transit with Pytmosph3R. We are in the process of extending Pytmosph3R to generate transit light-curves, which could be very useful to theoretically study transit observations. As Pytmosph3R is fully 3D, the light-curves that are generated would be the closest to a real observation and would avoid biases due to 1D model assumptions. It will be interesting to compare the information extracted from light-curves by other codes (Kreidberg et al. 2015; Tsiaras et al. 2018; Espinoza & Jones 2021; Feliz et al. 2021) to the input model provided to Pytmosph3R.

6. Conclusion

We have discussed the computation of transmission spectra for exoplanetary atmospheric simulations with a varying number of dimensions. This method, implemented in Pytmosph3R, handles atmospheric simulations with up to three dimensions, including GCM models. The 2D formulation has also been integrated into the (initially one-dimensional) TauREx framework (Al-Refaie et al. 2021). We then discussed the computational requirements and efficiency of each model, which is especially critical in the context of retrievals, as well as possible applications.

We have introduced a new version of `Pytmosph3R` that is more robust and flexible, which is open-source and under a BSD license[1]. Taking into account the 3D structure of the atmosphere during a transit is essential for generating consistent observations because assuming atmospheres to be homogeneous leads to strong differences in the transmittance maps and in the final integrated spectrum (Caldas et al. 2019; Pluriel et al. 2020b). We will further study the biases due to the 1D assumption of retrievals with Pytmosph3R 2.0 in the second part of this series of articles, and quantify these biases.

The two-dimensional model was shown to be a good compromise between accuracy and computational requirements, making it a valid forward model for a retrieval. Thanks to this method, we can remove the biases that were observed when a 1D forward model is used to retrieve very hot exoplanets. We will discuss the relevance, precision, and reliability of this 2D retrieval in the third part of this series.

However, it should be noted that there might be other ways to parameterize 2D retrievals (discussed in Sect. 1). For instance, we know that east-west effects might also bias transmission spectra in warm atmospheres (MacDonald et al. 2020), where the jet stream cools down the west limb and heats up the east limb. In these configurations, our 2D model, which is symmetric with respect to the star-observer line, would not be able to give a better solution than a 1D model because its limb is homogeneous by definition. 2D retrievals using adapted configurations depending on the type of observed exoatmospheres are required. We could also develop hybrid 2D models that would take several geometric effects into account, keeping in mind that too many parameters in a retrieval code may create degeneracies. Overall, TauREx 2D can infer the atmospheric parameters of specific exoplanetary types, that is, ultra hot Jupiters, with a good compromise between computational time and model precision. This 2D version of TauREx will be made publicly available in the near future.

Acknowledgements. We are grateful to all of the TauREx developping team. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement n°679030/WHIPLASH). We thank the Programme National de Planétologie (CNRS/INSU/PNP) and the CNES for their financial support.

[2] The egress and ingress spectra were multiplied by 2 to compare them to the other phases, therefore the differences shown in Fig. 15 for these phases are twice larger than for the real signal.

References

Ahrens, J., Geveci, B., & Law, C. 2005, The Visualization Handbook (Amsterdam: Elsevier), 717
Al-Refaie, A. F., Changeat, Q., Waldmann, I. P., & Tinetti, G. 2021, ApJ, 917, 37
Barstow, J. K., Aigrain, S., Irwin, P. G. J., & Sing, D. K. 2017, ApJ, 834, 50
Caldas, A., Leconte, J., Selsis, F., et al. 2019, A&A, 623, A161
Changeat, Q., Edwards, B., Waldmann, I. P., & Tinetti, G. 2019, ApJ, 886, 39
Charnay, B., Meadows, V., Misra, A., Leconte, J., & Arney, G. 2015, ApJ, 813, L1
Charnay, B., Blain, D., Bézard, B., et al. 2021, A&A, 646, A171
Drummond, B., Tremblin, P., Baraffe, I., et al. 2016, A&A, 594, A69
Edwards, B., Changeat, Q., Baeyens, R., et al. 2020, AJ, 160, 8
Espinoza, N., & Jones, K. 2021, AJ, 162, 165
Feliz, D. L., Plavchan, P., Bianco, S. N., et al. 2021, AJ, 161, 247
Greene, T. P., Line, M. R., Montero, C., et al. 2016, ApJ, 817, 17
Guerlet, S., Spiga, A., Sylvestre, M., et al. 2014, Icarus, 238, 110
Guilluy, G., Gressier, A., Wright, S., et al. 2021, AJ, 161, 19
Irwin, P. G. J., Teanby, N. A., de Kok, R., et al. 2008, J. Quant. Spectr. Rad. Transf., 109, 1136
Kataria, T., Sing, D. K., Lewis, N. K., et al. 2016, ApJ, 821, 9
Kreidberg, L., Line, M. R., Bean, J. L., et al. 2015, ApJ, 814, 66
Lam, S. K., Pitrou, A., & Seibert, S. 2015, Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC, 1–6
Leconte, J. 2021, A&A, 645, A20
Leconte, J., Forget, F., Charnay, B., Wordsworth, R., & Pottier, A. 2013a, Nature, 504, 268
Leconte, J., Forget, F., Charnay, B., et al. 2013b, A&A, 554, A69
Line, M. R., Wolf, A. S., Zhang, X., et al. 2013, ApJ, 775, 137
Line, M. R., Stevenson, K. B., Bean, J., et al. 2016, AJ, 152, 203
Lodders, K., & Fegley, B. 2002, Icarus, 155, 393
MacDonald, R. J., Goyal, J. M., & Lewis, N. K. 2020, ApJ, 893, L43
Menou, K., & Rauscher, E. 2009, ApJ, 700, 887
Mikal-Evans, T., Sing, D. K., Kataria, T., et al. 2020, MNRAS, 496, 1638
Mollière, P., Wardenier, J. P., van Boekel, R., et al. 2019, A&A, 627, A67
Parmentier, V., Line, M. R., Bean, J. L., et al. 2018, A&A, 617, A110
Pluriel, W., Whiteford, N., Edwards, B., et al. 2020a, AJ, 160, 112
Pluriel, W., Zingales, T., Leconte, J., & Parmentier, V. 2020b, A&A, 636, A66
Seidel, J. V., Ehrenreich, D., Wyttenbach, A., et al. 2019, A&A, 623, A166
Showman, A. P., & Guillot, T. 2002, A&A, 385, 166
Showman, A. P., Menou, K., & Cho, J. Y. K. 2008, ASP Conf. Ser., 398, 419
Showman, A. P., Lewis, N. K., & Fortney, J. J. 2015, ApJ, 801, 95
Stevenson, K. B., Désert, J.-M., Line, M. R., et al. 2014, Science, 346, 838
Stock, J. W., Kitzmann, D., Patzer, A. B. C., & Sedlmayr, E. 2018, MNRAS, 479, 865
Swain, M. R., Estrela, R., Roudier, G. M., et al. 2021, AJ, 161, 213
Tan, X., & Komacek, T. D. 2019, ApJ, 886, 26
Tinetti, G., Vidal-Madjar, A., Liang, M.-C., et al. 2007, Nature, 448, 169
Tsiaras, A., Waldmann, I. P., Zingales, T., et al. 2018, AJ, 155, 156
Venot, O., Agúndez, M., Selsis, F., Tessenyi, M., & Iro, N. 2014, A&A, 562, A51
Waldmann, I. P., Tinetti, G., Rocchetto, M., et al. 2015a, ApJ, 802, 107
Waldmann, I. P., Rocchetto, M., Tinetti, G., et al. 2015b, ApJ, 813, 13
Wordsworth, R. D., Forget, F., Selsis, F., et al. 2011, ApJ, 733, L48