An Algorithm to Compress Line-transition Data for Radiative-transfer Calculations

Patricio E. Cubillos®
Space Research Institute, Austrian Academy of Sciences, Schmiedlstrasse 6, A-8042, Graz, Austria; patricio.cubillos@oeaw.ac.at
Received 2017 August 24; revised 2017 October 4; accepted 2017 October 5; published 2017 November 14

Abstract

Molecular line-transition lists are an essential ingredient for radiative-transfer calculations. With recent databases now surpassing the billion-line mark, handling them has become computationally prohibitive, due to both the required processing power and memory. Here I present a temperature-dependent algorithm to separate strong from weak line transitions, reformating the large majority of the weaker lines into a cross-section data file, and retaining the detailed line-by-line information of the fewer strong lines. For any given molecule over the 0.3–30 μm range, this algorithm reduces the number of lines to a few million, enabling faster radiative-transfer computations without a significant loss of information. The final compression rate depends on how densely populated the spectrum is. I validate this algorithm by comparing Exomol’s HCN extinction-coefficient spectra between the complete (65 million line transitions) and compressed (7.7 million) line lists. Over the 0.6–33 μm range, the average difference between extinction-coefficient values is less than 1%. A Python/C implementation of this algorithm is open-source and available at https://github.com/p cubillos/repack. So far, this code handles the Exomol and HITRAN line-transition format.

Key words: atomic data – methods: numerical – radiative transfer

1. Introduction

The study of exoplanet atmospheres and their spectra critically depends on the available laboratory and theoretical data of gaseous species (Fortney et al. 2016). The discovery of highly irradiated substellar atmospheres has motivated the compilation of molecular line-transition lists at temperatures far above those of the Earth’s atmosphere (Rothman et al. 2010; Tennyson et al. 2016). However, these newest databases are starting to grow into ~ billions of line transitions (e.g., Rothman et al. 2010; Yurchenko et al. 2011; Yurchenko & Tennyson 2014).

To date, medium- to low-resolution multi-wavelength observations of exoplanets cover a broad wavelength range (~0.3 to 30 μm), requiring the use of the line-transition data nearly in their entirety. With the arrival of future facilities, like the James Webb Space Telescope (JWST), this picture will remain. Such large line-transition data files render radiative-transfer calculations computationally prohibitive, both in terms of the necessary memory and processing power.

To keep the molecular line lists manageable, authors commonly set a fixed opacity cutoff, discarding all lines weaker than a certain threshold (e.g., Sharp & Burrows 2007). However, this approach is at best below optimal, as one could remove entire absorption bands at certain wavelengths, or retain a large number of line transitions that do not significantly contribute to the opacity.

Inspired by the idea of Hargreaves et al. (2015) of separating line-by-line and continuum line-transition information, I devised an algorithm to reduce the amount of line-transition data required for radiative-transfer calculations, with minimal loss of information. Using this approach, one retains the full information only of the stronger lines that dominate the absorption spectrum, and compresses the combined information of the many additional weak lines into a cross-section data file, as a function of wavenumber and temperature. Since, the interpretation of mid- and low-resolution observations relies more on the total opacity contribution rather than the individual line transitions, this algorithm allows for a significant performance improvement of radiative-transfer calculations.

2. Methods

The integrated absorption intensity (or opacity or extinction coefficient) of a line transition (in cm$^{-1}$) can be expressed as

$$S_j = \frac{e^2}{m_e c^2} \left(\frac{\hbar c \nu_j}{k_B T}\right) \frac{1}{n_e} \exp \left(-\frac{\hbar c \nu_j}{k_B T}\right)$$

$$\left[1 - \exp \left(-\frac{\hbar c \nu_j}{k_B T}\right)\right]$$

where $\nu_j$, $E_{\text{low}}^j$, and $n_e$ are the weighted oscillator strength, central wavenumber, and lower-state energy level of the line transition $j$, respectively, $Z_i$ and $n_i$ are the partition function and number density of the isotope $i$, respectively, $T$ is the temperature, $e$ and $m_e$ are the electron’s charge and mass, respectively, $c$ is the speed of light, $h$ is Planck’s constant, and $k_B$ is the Boltzmann’s constant.

For any given molecule, the number density of an isotope can be written as $n_i = n_m q_i$, where $n_m$ is the molecule number density and $q_i$ is the isotopic abundance fraction (which can be assumed to be at Earth values). Then, by knowing the set of isotopic fractions for a given molecule, we can express the line intensities per unit of the molecule’s number density (in cm² mole⁻¹) as $s_j = S_j/n_m$.

To compute the extinction-coefficient spectrum, one needs to broaden each line according to the Voigt profile function (the convolution of a Doppler and a Lorentz profile), and then summing the contribution from all lines. Thus, to identify the dominant lines, one has to consider the dilution of the line intensity by the Voigt broadening.

In practice, for a given molecule, the Voigt broadening profile varies weakly over neighboring lines. Since the Doppler broadening is simpler to compute than the Lorentz profile, I approximate the Voigt broadening by the Doppler broadening profile:

$$I_D(\nu) = \frac{1}{\delta_D^2} \exp \left(-\frac{(\nu - \nu_D)^2}{\delta_D^2}\right)$$
with line width

\[ \delta_{D,j} = \frac{\nu_j}{c} \sqrt{\frac{2kT}{m_i}}, \]

where \( m_i \) is the mass of the isotope.

Figure 1 shows the typical values for the Doppler and Lorentz half-width at half-maximum (HWHM) for exoplanet atmospheres. The Doppler broadening dominates the line profile above \( \sim 0.3 \) bar, across temperature ranges of 500 to 2500 K and wavelengths of 0.6–12 \( \mu \)m. In this pressure range, the line-transition widths range between \( \sim 10^{-3} \) cm\(^{-1}\) and \( \sim 10^{-1} \) cm\(^{-1}\), depending on the wavelength and atmospheric parameters. (Cubillos et al. 2017, submitted) studied the atmospheric pressures probed by transmission and emission spectroscopy of gaseous exoplanets, for a wide range of planetary properties. They found that the typical optical-to-infrared transmission observations probe pressures between 1 and \( 10^{-3} \) bar, whereas day-side emission observations probe between 10 and \( 10^{-2} \) bar. Therefore, most of the observable atmosphere is under the Doppler broadening regime. If one approximates the line shape by the Doppler profile, Equation (2) then states that the maximum intensity of a line is approximately \( s_j' = s_j/\delta_{D} \sqrt{\pi} \) (hereafter, called diluted line intensity). In Section 3.1, we show that this is an acceptable assumption, even when there is significant Lorentz broadening, for medium- or low-resolution observations.

### 2.1. Line-flagging Algorithm

To efficiently identify dominant from weak lines, one can start by selecting the strongest line in a given wavelength range, say, line transition \( j \), and compute its Doppler profile \( s_j(D) \). Then, one flags out the surrounding lines whose diluted intensity \( s_k' \) is smaller than the profile of line \( j \), with a threshold tolerance \( f_{\text{tol}} \), i.e.:

\[ s_k' < f_{\text{tol}} s_j(D_j(\nu_k)). \] (4)

Since the line intensity decays exponentially as one moves away from its center, the flagged (weak) lines do not significantly contribute to the extinction coefficient. This effectively avoids the need to broaden most of the lines in a database. Line intensities span several orders of magnitude, and thus, only the few strongest lines at any given wavelength dominate the absorption spectrum (Figure 2). The algorithm then proceeds with the next unflagged strongest line transition, and so on.

Now, different lines can dominate the spectrum depending on the atmospheric temperature (see Equation (1)). To account for this, one repeats the flagging process at the two extreme temperatures to consider (a user choice, for example, 300 and 3000 K is appropriate for substellar objects).

After one identifies strong and weak lines, one preserves the full line-by-line information only for the strong lines (for radiative transfer, the \( \nu_j, E_j, g_j, f_j \), and the isotope ID suffice). The information from the large majority of weak lines can be compressed into a continuum extinction-coefficient table as a function of wavelength and temperature. To avoid broadening each line, one simply adds the line intensity to the nearest tabulated wavenumber point, diluting the line according to the wavenumber sampling rate (following Sharp & Burrows 2007).

### 3. Open-source Implementation

Along with this article, I provide an open-source version of this algorithm (under the MIT license), available at https://github.com/cubillos/repack. This is a Python package (accelerated with C subroutines) compatible with Python2 and Python3, running on both Linux and OSX. This package handles the Exomol and HITRAN input line-transition formats. The routine’s performance, varies with the size of the initial database, the number of evaluated profiles (which depends on \( f_{\text{tol}} \)), and how densely packed the line transitions are. For an Intel Core i7-4790 3.60 GHz CPU, the routine runs the Exomol HCN (65 million lines, Harris et al. 2006, 2008; Barber et al. 2014), NH\(_3\) (\( \sim \)1 billion, Yurchenko et al. 2011), and CH\(_4\) (10 billion, Yurchenko & Tennyson 2014) databases in \( \sim \)10 minutes, \( \sim \)7 hr, and \( \sim \)5 days, respectively. The performance scales somewhat faster than linear with the number of line transitions. Certainly,
the gain of working with the compressed line lists more than compensates for the time spent running this routine (a one-time run) for the largest databases. Ultimately the line-by-line compression rate will depend on how compact or saturated a line list is. For example, with a threshold tolerance of 0.01, this algorithm compresses the Exomol HCN database by ∼90% in the 0.6–33 μm range. The algorithm compresses the denser Exomol NH₃ line list (∼1 billion) by ∼95%.

3.1. Validation

To validate the compression algorithm, I compare the extinction-coefficient spectra produced from the compressed and the complete line-by-line database for HCN from Exomol. This database comprises ∼65 million lines over the 0.6–33.0 μm range. Adopting a threshold tolerance of $f_{tol} = 0.01$, and temperature values between 500 and 3000 K, the algorithm retains 7.7 million lines.

To compute the HCN’s extinction-coefficient spectrum, I use the Python Radiative-transfer in a Bayesian framework package (Pyrat Bay, \(^1\) P. Cubillos et al. 2017a, in preparation). Pyrat Bay is based on the Bayesian Atmospheric Radiative Transfer package (Blecic 2016; Cubillos 2016). Figure 3 (top panel) shows the resulting spectra for a typical Jupiter-composition planet, at an atmospheric temperature of 1540 K.

---

\(^1\) http://pcubillos.github.io/pyratbay

---

Figure 3. Top: HCN extinction coefficient from the full Exomol line list (blue), the compressed line-by-line (LBL) data set (red), and the compressed continuum data (green) at 0.1 bar and 1540 K. Note that, at this scale, the full and compressed LBL curves are indistinguishable, and thus, the red curve falls on top of the blue curve. The spectra in this panel are sampled over a regular wavenumber grid with 0.1 cm⁻¹ sampling rate, without instrumental broadening. Middle: HCN extinction-coefficient spectrum difference between the full and compressed line lists at 0.1 bar and 1540 K. The orange and black curves show the difference using the total (LBL plus continuum) components and only the LBL component of the compressed line list, respectively. These spectra simulate an instrumental resolving power of 1 cm⁻¹ (Gaussian filter applied after the radiative-transfer calculation). Bottom: same as the middle panel, evaluated at 50 bar.
and pressure of 0.1 bar. Figure 3 (middle panel) shows the difference in extinction coefficient between the full line list and the compressed data set. The spectra for this panel simulate an instrumental resolving power of 1 cm\(^{-1}\) (approximately the highest resolution that the JWST instruments will achieve). The LBL component alone reproduces the full line-list extinction coefficient down to \(\sim 1\%\) on average, and down to a few percent in the worst case. When, considering both the LBL and continuum components, the compressed data set reproduces the full line-list spectrum well under 1%.

At higher pressures, where the Lorentz profile dominates the line broadening, the compressed data set still reproduces well the full line-list spectrum (Figure 3, bottom panel). Some overestimated values arise because the compressed continuum opacity does not consider the Lorentz broadening. However, the differences are still on the order of a few percent. The mismatch varies with the given instrumental resolution, with a coarser resolution producing smaller differences.

In summary, I presented an efficient compression algorithm that identifies the line transitions that dominate a spectrum. This algorithm is aimed to serve radiative-transfer modeling of medium to low spectral resolution data over broad wavelength ranges, like that of the JWST.

I thank Dr. L. Fossati and Dr. J. Blecic for useful comments. I also thank the anonymous referee for his/her time and valuable comments. I thank contributors to Numpy (van der Walt et al. 2011), SciPy (Jones et al. 2001), Matplotlib (Hunter 2007), the Python Programming Language, the developers of the AASTeX latex template (Schwarz & Muench 2016); and the free and open-source community. I acknowledge the Austrian Forschungsförderungsgesellschaft FFG projects “RASEN” P847963 and “TAPAS4CHEOPS” P853993, the Austrian Science Fund (FWF) FNF projects S11607-N16 and S11604-N16, and the FWF project P27256-N27. The Reproducible Research Compendium (RRC) of this article is available at https://github.com/pcubillos/Cubillos2017_repack.

Software: Repack: https://github.com/pcubillos/repack, the Python Radiative Transfer in a Bayesian framework: http://pcubillos.github.io/pyratbay, and Latex template: https://github.com/pcubillos/ApJtemplate.

**ORCID iDs**

Patricio E. Cubillos @ https://orcid.org/0000-0002-1347-2600

**References**

Barber, R. J., Strange, J. K., Hill, C., et al. 2014, MNRAS, 437, 1828
Blecic, J. 2016, PhD Dissertation (arXiv:1604.02692)
Cubillos, P., Kubyshkina, D., Fossati, L., Mordasini, C., & Lendl, M. 2017, A&A, submitted
Cubillos, P. E. 2016, PhD Dissertation (arXiv:1604.01320)
Fortney, J. J., Robinson, T. D., Domagal-Goldman, S., et al. 2016, arXiv:1602.06305
Hargreaves, R. J., Bernath, P. F., Bailey, J., & Dulick, M. 2015, ApJ, 813, 12
Harris, G. J., Larver, F. C., Tennyson, J., et al. 2008, MNRAS, 390, 143
Harris, G. J., Tennyson, J., Kaminsky, B. M., Pavlenko, Y. V., & Jones, H. R. A. 2006, MNRAS, 367, 400
Hunter, J. D. 2007, CSE, 9, 90
Jones, E., Oliphant, T., Peterson, P., et al. 2001, SciPy: Open Source Scientific Tools for Python
Rothman, L. S., Rinsland, C. P., Goldman, A., et al. 2010, JQSRT, 111, 2139
Schwarz, G., & Muench, A. 2016, AASTeX v6.0.1, Zenodo, doi:10.5281/zenodo.45282
Sharp, C. M., & Burrows, A. 2007, ApJS, 168, 140
Tennyson, J., Yurchenko, S. N., Al-Refaie, A. F., et al. 2016, JMoSp, 327, 73
van der Walt, S., Colbert, S. C., & Varoquaux, G. 2011, CSE, 13, 22
Yurchenko, S. N., Barber, R. J., & Tennyson, J. 2011, MNRAS, 413, 1828
Yurchenko, S. N., & Tennyson, J. 2014, MNRAS, 440, 1649