ACCEPTED IN *ApJ Supplement Series*.
Preprint typeset using LaTeX style emulateapj v. 8/13/10

TEA: A CODE FOR CALCULATING THERMOCHEMICAL EQUILIBRIUM ABUNDANCES

JASMINA BLECIC[1], JOSEPH HARRINGTON[1], M. OLIVER BOWMAN [1]
[1] Planetary Sciences Group, Department of Physics, University of Central Florida, Orlando, FL 32816-2385, USA
Accepted in ApJ Supplement Series.

ABSTRACT

We present an open-source Thermochemical Equilibrium Abundances (TEA) code that calculates the abundances of gaseous molecular species. The code is based on the methodology of White et al. (1958) and Eriksson (1971). It applies Gibbs free-energy minimization using an iterative, Lagrangian optimization scheme. Given elemental abundances, TEA calculates molecular abundances for a particular temperature and pressure or a list of temperature-pressure pairs. We tested the code against the method of Burrows & Sharp (1999), the free thermochemical equilibrium code CEA (Chemical Equilibrium with Applications), and the example given by White et al. (1958). Using their thermodynamic data, TEA reproduces their final abundances, but with higher precision. We also applied the TEA abundance calculations to models of several hot-Jupiter exoplanets, producing expected results. TEA is written in Python in a modular format. There is a start guide, a user manual, and a code document in addition to this theory paper. TEA is available under a reproducible-research, open-source license via `https://github.com/dzesmin/TEA`.

Subject headings: astrochemistry – molecular processes – methods: numerical – planets and satellites: atmospheres – planets and satellites: composition – planets and satellites: gaseous planets

1. INTRODUCTION

There are two methods to calculate chemical equilibrium: using equilibrium constants and reaction rates, i.e., kinetics, or minimizing the free energy of a system (Bahn & Zukoski 1960, Zeleznik & Gordon 1968).

The kinetic approach, where the pathway to equilibrium needs to be determined, is applicable for a wide range of temperatures and pressures (Line et al. 2011, Moses et al. 2011, Visscher et al. 2010b). However, using kinetics for high-temperature thermochemical equilibrium calculations can be challenging. Chemical equilibrium can be calculated almost trivially for several reactions present in the system, but as the number of reactions increases, the set of numerous equilibrium constant relations becomes hard to solve simultaneously. To have an accurate kinetic assessment of the system, one must collect a large number of reactions and associate them with the corresponding rates. This is not an issue at lower temperatures, where reaction rates are well known. However, at high temperatures, where thermochemical equilibrium should prevail, one needs to know forward and reverse reactions and corresponding reaction rates, which are less well known or conflicted (Venot et al. 2012, Visscher et al. 2010b).

The advantage of the free energy minimization method is that each species present in the system can be treated independently without specifying complicated sets of reactions a priori, and therefore, a limited set of equations needs to be solved (Zeleznik & Gordon 1960). In addition, the method requires only knowledge of the free energies of the system, which are well known, tabulated, and can be easily interpolated or extrapolated.

Thermochemical equilibrium calculations have been widely used in chemical engineering to model combustion, shocks, detonations and the behaviour of rockets and compressors (e.g., Miller et al. 1990, Belford & Strehlow 1969). In astrophysics, they have been used to model the solar nebula, the atmospheres and circumstellar envelopes of cool stars, and the volcanic gases on Jupiter's satellite Io (e.g., Lauretta et al. 1997, Lodders & Fegley 1993, Zolotov & Fegley 1998).

Thermochemistry also governs atmospheric composition in vast variety of giant planets, brown dwarfs, and low-mass dwarf stars (Lodders & Fegley 2002, Visscher et al. 2010a, Allard & Hauschildt 1995, Tsuji et al. 1996, Marley & Robinson 2015, and references therein). Recent studies (Moses et al. 2011, Zahnle et al. 2009, Line et al. 2010, Kopparapu et al. 2012, Venot et al. 2012, Line & Yung 2013, Visscher et al. 2006, 2010b, Venot et al. 2012) show that in hot exoplanetary atmospheres ($T > 1200$ K), disequilibrium effects are so reduced that thermochemical equilibrium prevails. For the hottest planets, Line & Yung (2013) show that that at the pressure of 100 mbar CH_4, CO, H_2O, and H_2 should be in thermochemical equilibrium even under a wide range of vertical mixing strengths. In addition, the exoplanet photospheres observed with current instruments are sampled within the region of the atmosphere dominated by vertical mixing and quenching, but not by photochemistry (Line & Yung 2013).

Thermochemical equilibrium calculations are the starting point for initializing models of any planetary atmosphere. In general, thermochemical equilibrium governs the composition of the deep atmospheres of giant planets and brown dwarfs, however, in cooler atmospheres thermoequilibrium calculations are the necessary baseline for further disequilibrium assessment. They can also provide a first-order approximation for species abundances as a function of pressure, temperature, and metallicity for a variety of atmospheres (e.g., Visscher et al. 2010b, Lodders & Fegley 2002).

The Gibbs free energy minimization method for calculating thermochemical equilibrium abundances of complex mixtures was first introduced by White et al. (1958). Prior to 1958 all equilibrium calculations were done using equilibrium constants of the governing reactions. White et al. (1958) were the first to develop a method that makes no distinction among the constituent species and does not need a list of all possible chemical reactions and their rates. Rather, it depends only on

jasmina@physics.ucf.edu

the chemical potentials of the species involved.

In this paper, we present an open-source code, Thermochemical Equilibrium Abundances (TEA) based on the Gibbs free energy minimization approach by White et al. (1958) and Eriksson (1971). Given a single T,P point or a list of T,P pairs (the thermal profile of an atmosphere) and elemental abundances, TEA calculates mole fractions of the desired gaseous molecular species. It uses 84 elemental species and the thermodynamical data for more then 600 gaseous molecular species available in the provided JANAF (Joint Army Navy Air Force) tables (`http://kinetics.nist.gov/janaf/`, Chase 1998). TEA can adopt any initial elemental abundances. For user convenience a table with solar photospheric elemental abundances from Asplund et al. (2009) is provided.

The TEA code is a part of the open-source Bayesian Atmospheric Radiative Transfer project (`https://github.com/exosports/BART`). This project consists of three major parts: TEA - this code, a radiative-transfer code that models planetary spectra, and a statistical module that compares theoretical models with observations. TEA is written in Python in an architecturally modular format. It is accompanied by detailed documentation, a start guide, the TEA User Manual (Bowman and Blecic), the TEA Code Description document (Blecic and Bowman), and the TEA Theory document (this paper), so the user can easily modify it. The code is actively maintained and available to the scientific community via the open-source development website `GitHub.com` (`https://github.com/dzesmin/TEA`, `https://github.com/dzesmin/TEA-Examples`). This paper covers an initial work on thermochemical calculations of species in gaseous phases. Implementation of condensates is left for future work.

In this paper, we discuss the theoretical basis for the method applied in the code. Section 2 explains the Gibbs Free energy minimization method; Section 3 describes the general Lagrangian optimization method and its application in TEA; in Section 4 we introduce the Lambda Correction algorithm for handling negative abundances that follow from the Lagrangian method; Section 5 describes the layout of the TEA code; Section 6 explores chemical equilibrium abundance profiles of several exoplanetary atmospheres; Section 7 compares our code to other methods available, and Section 9 states our conclusions.

2. GIBBS FREE ENERGY MINIMIZATION METHOD

Equilibrium abundances can be obtained by using different combinations of thermodynamical state functions: temperature and pressure – (t,p), enthalpy and pressure – (H,p), entropy and pressure – (S,p), temperature and volume – (t,v), internal energy and volume – (U,v), etc. Depending on how the system is described, the condition for equilibrium can be stated in terms of Gibbs free energy, helmholtz energy, or entropy. If a thermodynamic state is defined with temperature and pressure, Gibbs free energy (G) is most easily minimized, since those two states are its natural, dependent variables.

Gibbs free energy represents a thermodynamic potential that measures the useful work obtainable by the system at a constant temperature and pressure. Thus, the Gibbs free energy minimization method minimizes the total chemical potential of all involved species when the system reaches equilibrium.

The Gibbs free energy of the system at a certain temperature is the sum of the Gibbs free energies of its constituents:

$$G_{sys}(T) = \sum_{i}^{n} G_i(T)\,, \tag{1}$$

where $G_{sys}(T)$ is the total Gibbs free energy of the system for n chemical species, $G_i(T)$ is the Gibbs free energy of a gas species i, and T is the temperature. The total Gibbs free energy of the system is expressed as the sum of the number of moles x of the species i, x_i, and their chemical potentials $g_i(T)$:

$$G_{sys}(T) = \sum_{i}^{n} x_i\, g_i(T)\,. \tag{2}$$

The chemical potential $g_i(T)$ depends on the chemical potential at the standard state $g_i^0(T)$ and the activity a_i,

$$g_i(T) = g_i^0(T) + RT\ln a_i\,, \tag{3}$$

where R is the gas constant, $R = k_B N_A$, and k_B and N_A are the Boltzmann constant and Avogadro's number, respectively. Activities for gaseous species, which are treated as ideal, are equal to the partial pressures, and for condensates they equal 1:

$$a_i = P_i = P\,\frac{x_i}{N}\,, \quad \text{for gases} \tag{4}$$

$$a_i = 1\,, \quad \text{for condensates}\,, \tag{5}$$

where P is the total pressure of the atmosphere, N is the total number of moles of all species involved in the system. Hence, Equation (3) for gaseous species becomes:

$$g_i(T) = g_i^0(T) + RT\ln P_i\,. \tag{6}$$

Combining Equation (6) with Equation (2), the Gibbs free energy of the system becomes:

$$G_{sys}(T) = \sum_{i}^{n} x_i \left(g_i^0(T) + RT\ln P_i\right), \tag{7}$$

or,

$$G_{sys}(T) = \sum_{i}^{n} x_i \left(g_i^0(T) + RT\ln P + RT\ln\frac{x_i}{N}\right), \tag{8}$$

For our purposes, it is more convenient to write Equation (8) in unitless terms:

$$\frac{G_{sys}(T)}{RT} = \sum_{i=1}^{n} x_i \left[\frac{g_i^0(T)}{RT} + \ln P + \ln\frac{x_i}{N}\right]. \tag{9}$$

Equation (9) requires a knowledge of the free energy of each species as a function of temperature. These can be obtained from the JANAF tables (`http://kinetics.nist.gov/janaf/`, Chase 1998, Burrows & Sharp 1999), or easily derived from other tabulated functions.

To extract free energies, $g_i^0(T)/RT$, from the JANAF tables, we used the expression given in Eriksson (1971), Equation (2):

$$\frac{g_i^0(T)}{RT} = 1/R\left[\frac{G_i^0 - H_{298}^0}{T}\right] + \frac{\Delta_f H_{298}^0}{RT}, \quad (10)$$

where $g_i^0(T)$ is given in J/mol, $R = 8.3144621$ J/K/mol, H_{298}^0 is the enthalpy (heat content) in the thermodynamical standard state at a reference temperature of 25°C = 298.15 K, G_i^0 is the Gibbs free energy in J/mol, $(G_i^0 - H_{298}^0/T)$ is the free-energy function in J/K/mol, and $\Delta_f H_{298}^0$ is the heat of formation at 298.15 K in kJ/mol. Thus, our conversion equation becomes:

$$\frac{g_i^0(T)}{RT} = 1/R\left[\frac{G_i^0 - H_{298}^0}{T}\right] + \frac{\Delta_f H_{298}^0 1000}{RT}, \quad (11)$$

$G_i^0 - H_{298}^0/T$ is the fourth term in the JANAF tables and $\Delta_f H_{298}^0$ is the sixth. The free energy function of a species corresponding to a temperature other than those provided in the JANAF tables is calculated using spline interpolation.

Alternatively, the free energies can be calculated using the eighth term in the JANAF tables, following Equation (3) from Eriksson (1971):

$$\frac{g_i^0(T)}{RT} = -ln(10)\, log_{10}(K_f), \quad (12)$$

where K_f is the equilibrium constant of formation.

To determine the equilibrium composition, we need to find a non-negative set of values x_i that minimizes Equation (9) and satisfies the mass balance constraint:

$$\sum_{i=1}^{n} a_{ij} x_i = b_j, \quad (j = 1, 2, ..., m), \quad (13)$$

where the stoichiometric coefficient a_{ij} indicates the number of atoms of element j in species i (e.g., for CH_4 the stoichiometric coefficient of C is 1 and the stoichiometric coefficient of H is 4), and b_j is the total number of moles of element j originally present in the mixture.

We use the reference table containing elemental solar abundances given in Asplund et al. (2009) Table 1 for b values. Asplund et al. (2009) adopt the customary astronomical scale for logarithmic abundances, where hydrogen is defined as log $\epsilon H = 12.00$, and log $\epsilon X = \log(N_X/N_H)+12$, where N_X and N_H are the number densities of element X and H, respectively. Thus, their values are given in *dex* (decimal exponent) units. We transform these values into elemental fractions by number, i.e., ratio of number densities. We convert each species dex elemental abundance into number density and divide it by the hydrogen number density (Asplund et al. 2009, Section 3). The final output are fractional abundances (mole mixing fractions), i.e., the ratio of each species' number of moles to the number of moles in the mixture.

3. LAGRANGIAN METHOD OF STEEPEST DESCENT

To find equilibrium abundances of the desired molecular species at a given temperature and pressure, we need to minimize Equation (9). To do so, we have to apply a technique that minimizes a multi-variate function under constraint. There are many optimization techniques used to find the minima of a function subject to equality constraints (e.g., line search method, Dantzig-simplex method for linear programming, Newton-Raphson method, Hessian-conjugate gradient method, Lagrangian steepest-descent method). The main advantage of the Lagrangian steepest-descent method is that the number of equations to solve scales with the number of different types of atoms present in the mixture, which is usually a much smaller number than the possible number of molecular constituents. This allows the code to be executed much faster than in other methods.

Gradient descent, also known as steepest descent, is an algorithm for finding a local minimum of a function. At each iteration, the method takes steps towards the minimum, where each step is proportional to the negative gradient of the function at the current point. If a function $f(x)$ is defined and differentiable in the neighborhood of a point a, then $f(x)$ decreases most rapidly in the direction of the negative gradient, $-\nabla f(\mathrm{a})$. From this, it follows that if $b = a - \lambda \nabla f(\mathbf{a})$, then $f(a) > f(b)$ if λ is small enough. Starting with a guess x_0 for a local minimum of f, and considering a sequence $x_0, x_1, x_2, ...$ such that $x_{n+1} = x_n - \lambda \nabla f(x_n)$, $n \geq 0$, one gets $f(x_0) \geq f(x_1) \geq f(x_2) \geq ...$. This sequence of x_n converges to a desired local minimum if the correct λ value is assigned. The value of λ can vary at each iteration. If the function f is convex, the local minimum is also the global minimum.

Our code implements a more complex version of the method outlined above. The problem consists of some function $f(x, y)$ subject to a constraint $g(x, y) = C$. In this case, we need both f and g to have continuous first partial derivatives. Thus, we introduce a new variable called the *Lagrangian multiplier*, π, where:

$$\Lambda(x, y, \lambda) = f(x, y) \pm \pi\,(g(x, y) - C), \quad (14)$$

which allows us to find where the contour of $g(x, y) = C$ tangentially touches $f(x, y)$ (Figure 1). The point of contact is where their gradients are parallel:

$$\nabla_{xy} f(x, y) = -\pi \nabla_{xy} g(x, y). \quad (15)$$

The constant π allows these gradients to have different magnitudes. To find the minimum, we need to calculate all partial derivatives of the function Λ, equate them with zero,

$$\nabla_{x,y,\pi}\, \Lambda(x, y, \pi) = 0, \quad (16)$$

and follow the same iteration procedure as explained above.

3.1. *Lagrangian Method in TEA*

To implement this in our code, we followed the methodology derived in White et al. (1958). We applied an iterative solution to the energy minimization problem, where the mole numbers of the desired molecular species are recomputed at each step and the new direction of steepest descent is calculated. This produces improved mole number values, which however, could be negative. Thus, two short procedures are required in each iteration cycle: solving a set of simultaneous linear equations for an improved direction of descent (described in this Section) and approximately minimizing a convex function of one variable, λ, to ensure that all improved mole number values are positive (Section 4).

To calculate the direction of steepest descent (following the methodology derived in Section 3) and initiate the first iteration cycle, we first need to solve the mass balance equation,

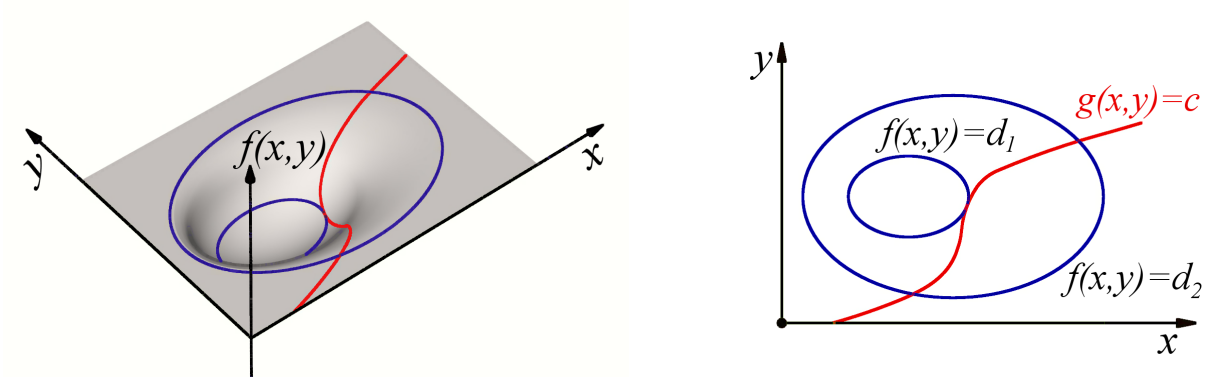

FIG. 1.— Example of the Lagrangian minimization approach. **Left:** The 3D illustration of the minimization problem. Blue lines indicate starting and ending values of $f(x,y)$ during minimization. An (x,y) pair is found that minimizes $f(x,y)$ (bottom blue line) subject to a constraint $g(x,y)=C$ (red line). **Right:** Contour map of the left figure. The point where the red line (constraint) tangentially touches a blue contour is the solution. Since $d_1 < d_2$, the solution is the minimum of $f(x,y)$.

Equation (13). We start from any positive set of values for the initial mole numbers, $y=(y_1,y_2,...,y_n)$, as our initial guess:

$$\sum_{i=1}^{n} a_{ij}\, y_i = b_j \quad (j=1,2,...,m). \tag{17}$$

To satisfy the mass balance Equation (17), some y_i variables must remain as free parameters. In solving these equations, we leave as many free parameters as we have elements in the system, thus ensuring that the mass balance equation can be solved for any number of input elements and output species the user chooses. We set all other y_i to a known, arbitrary number. Initially, the starting values for the known species are set to 0.1 moles, and the mass balance equation is calculated. If that does not produce all positive mole numbers, the code automatically sets known parameters to 10 times smaller and tries again. The initial iteration input is set when all mole numbers are positive, and the mass balance equation is satisfied.

To follow with the Lagrangian method, we denote two terms in Equation (9) as:

$$c_i = \frac{g_i^0(T)}{RT} + \ln P, \tag{18}$$

where P is the pressure in bar. Using c_i, we denote the right side of Equation (9) as the variable $f_i(Y)$:

$$f_i(Y) = y_i\left[c_i + \ln\frac{y_i}{\bar{y}}\right], \tag{19}$$

where $Y=(y_1,y_2,...,y_n)$ and $\bar{y}$ is the total initial number of moles. The left side of Equation (9), $G_{sys}(T)/RT$, we denote as function $F(Y)$:

$$F(Y) = \sum_{i=1}^{n} y_i\left[c_i + \ln\frac{y_i}{\bar{y}}\right]. \tag{20}$$

Then, we do a Taylor series expansion of the function F about Y. This yields a quadratic approximation $Q(X)$:

$$Q(X) = F(X)\Big|_{X=Y} + \sum_i \frac{\partial F}{\partial x_i}\Big|_{X=Y}\Delta_i + \frac{1}{2}\sum_i\sum_k \frac{\partial^2 F}{\partial x_i \partial x_k}\Big|_{X=Y}\Delta_i\,\Delta_k. \tag{21}$$

where $\Delta_i = x_i - y_i$, and x_i are the improved mole numbers. This function is minimized using the Lagrangian principle. We now introduce Lagrangian multipliers as π_j:

$$G(X) = Q(X) + \sum_j \pi_j(-\sum_i a_{ij}\,x_i + b_j), \tag{22}$$

and calculate the first derivatives, $\partial G/\partial x_i$, of the new function. We equate them to zero to find the minima, $\partial G/\partial x_i = 0$.

We solve for x_i from Equation (22) by combining Equation (17) and (19) with the fact that $\bar{x}$ is the sum of the improved mole numbers, $\bar{x} = \sum_{i=1}^{n} x_i$. The improved number of moles, x_i, are given as:

$$x_i = -f_i(Y) + (\frac{y_i}{\bar{y}})\bar{x} + (\sum_{j=1}^{m}\pi_j\, a_{ij})\, y_i, \tag{23}$$

while the Lagrangian multipliers, π_j, are expressed as:

$$\sum_{j=1}^{m}\pi_j\sum_{i=1}^{n} a_{ij}\,y_i = \sum_{i=1}^{n} y_i\left[\frac{g_i^0(T)}{RT} + \ln P + \ln\frac{y_i}{\bar{y}}\right], \tag{24}$$

where j iterates over the m elements and i iterates over the n species. $\bar{x}$ and $\bar{y}$ are the sums of improved and initial number of moles, respectively. Using Equation (19), we can now rewrite Equation (24) as:

$$\sum_{j=1}^{m}\pi_j\, b_j = \sum_{i=1}^{n} f_i(Y). \tag{25}$$

If we further denote the constants with:

$$r_{jk} = r_{kj} = \sum_{i=1}^{n}(a_{ij}\,a_{ik})\,y_i, \tag{26}$$

combining Equations (23), (25), and (26), we get the following system of $m+1$ equations that can easily be solved:

$$\begin{aligned} r_{11}\pi_1 + r_{12}\pi_2 + \ldots + r_{1m}\pi_m + b_1 u &= \sum_{i=1}^{n} a_{i1} f_i(Y), \\ r_{21}\pi_1 + r_{22}\pi_2 + \ldots + r_{2m}\pi_m + b_2 u &= \sum_{i=1}^{n} a_{i2} f_i(Y), \\ &\cdot \\ &\cdot \\ &\cdot \\ r_{m1}\pi_1 + r_{m2}\pi_2 + \ldots + r_{mm}\pi_m + b_m u &= \sum_{i=1}^{n} a_{im} f_i(Y), \\ b_1\pi_1 + b_2\pi_2 + \ldots + b_m\pi_m + 0\,u &= \sum_{i=1}^{n} f_i(Y), \end{aligned} \tag{27}$$

where:

$$u = -1 + \bar{x}/\bar{y}. \tag{28}$$

The solutions to Equations (27) and (28) will give π_j and u, and from them using Equation (23) we can calculate the next set of improved mole numbers, i.e., an improved direction of descent, $\Delta_i = x_i - y_i$.

4. LAMBDA CORRECTION ALGORITHM

Solving a system of linear equations (i.e., performing the Lagrangian calculation) can also lead to negative mole numbers for some species, so a short additional step is needed to eliminate this possibility and guarantee a valid result.

To do so, the difference between the initial and final values given by the Lagrangian calculation, $\Delta_i = x_i - y_i$, we will call the total distance for each species. To ensure that all improved mole numbers are positive, we introduce a new value, λ, that defines the fraction of the total distance as $\lambda\Delta_i$ (see Figure 2).

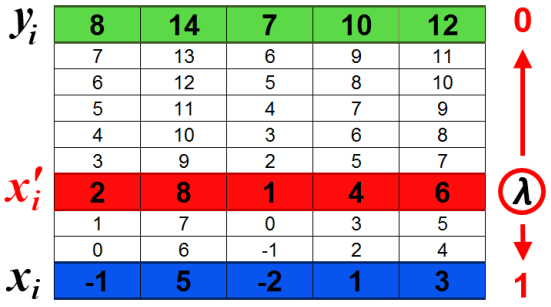

FIG. 2.— Simplified illustration of the lambda correction algorithm. Initial values for one hypothetical Lagrangian iteration cycle, y_i, are given in green. These values are all positive and satisfy the mass balance equation, Equation 17. The x_i values, given in blue, are the values produced by the Lagrangian calculation. These values can be negative, but they also satisfy the mass balance equation. The x_i' values, given in red, are produced by choosing the maximum value of lambda that ensures all positive and non-zero x_i'. These values become the new initial values of y_i for the next iteration cycle.

The computed changes, $\lambda\Delta_i$, are considered to be *directional numbers* indicating the preferred direction of descent the system moves to. Other than providing all positive mole numbers, we determine the value λ so that the Gibbs energy of the system must decrease, i.e., the minimum point is not passed (see Equation 34).

At each Lagrangian iteration cycle we start with the initial positive values, y_i and we get the next set of improved values x_i given as:

$$x_i = y_i + \Delta_i. \tag{29}$$

Since we do not want any x_i to be negative, the variable λ performs a small correction:

$$x_i' = y_i + \lambda\Delta_i. \tag{30}$$

λ takes values between 0 and 1, where value of *zero* implies no step is taken from the iteration's original input, y_i, and *one* implies that the full Lagrangian distance is travelled, Δ_i. We now rewrite Equation (19) using Equation (30) as:

$$f_i(X') = x_i'\left(\frac{g_i^0(T)}{RT} + \ln P + \ln\frac{x_i'}{\bar{x}'}\right), \tag{31}$$

which can be written in the form:

$$f_i(\lambda) = (y_i + \lambda\Delta_i)\left(\frac{g_i^0(T)}{RT} + \ln P + \ln\frac{y_i + \lambda\Delta_i}{\bar{y} + \lambda\bar{\Delta}}\right), \tag{32}$$

where $\bar{\Delta} = \bar{y} - \bar{x}$. Summing over i, we get a new function, $F(\lambda)$:

$$F(\lambda) = \sum_i (y_i + \lambda\Delta_i)\left(\frac{g_i^0(T)}{RT} + \ln P + \ln\frac{y_i + \lambda\Delta_i}{\bar{y} + \lambda\bar{\Delta}}\right). \tag{33}$$

Thus, to ensure that the new corrected values x_i' are all positive, the distance travelled will be limited to fractional amounts defined by $\lambda\Delta_i$, using the largest possible value of λ that satisfies the conditions:

1. The function called the *directional derivative* is defined and exists:

$$\frac{dF(\lambda)}{d\lambda} = \sum_{i=1}^{n} \Delta_i\left[\frac{g_i^0(T)}{RT} + \ln P + \ln\frac{y_i + \lambda\Delta_i}{\bar{y} + \lambda\bar{\Delta}}\right]. \tag{34}$$

2. The directional derivative does not become positive (the minimum point is not passed).

Every new iteration starts with a different set of y_i, thus changing the convex function $F(\lambda)$, Equation 33, and producing a new minimum. This yields to a new λ value. λ will be found to approach unity after some number of iterations. Unity in λ indicates the solution is near.

We repeat the Lagrangian method and the lambda correction until a pre-defined maximum number of iterations is met. The final abundances are given as fractional abundances (mole mixing fractions), i.e., the ratio of each species' mole numbers to the total sum of mole numbers of all species in the mixture.

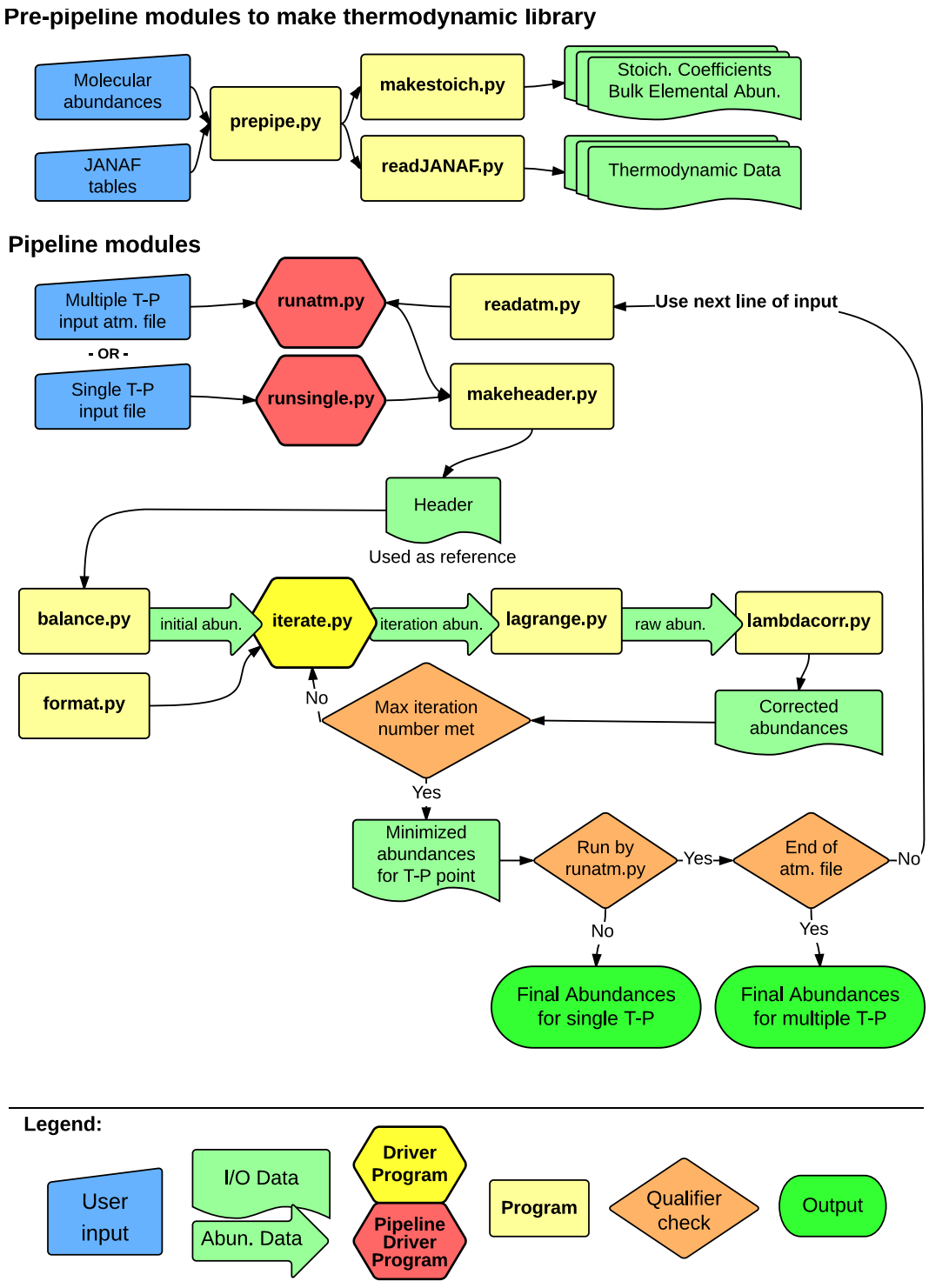

FIG. 3.— Layout of the TEA pre-pipeline and pipeline modules. The modules have one of three roles: scientific calculation, file or data structure support, or execution of the calculation programs over temperature and pressure points in an iterative manner. In addition to the modules shown, TEA has three supporting modules: readconfig.py, makeatm.py, and plotTEA.py. All modules are described in the text.

5. CODE STRUCTURE

The TEA code is written entirely in Python and uses the Python packages NumPy (`http://numpy.org/`) and (`http://www.scipy.org/`) along with SymPy, an external linear equation solver (`http://sympy.org/`).

The code is divided into two parts: the pre-pipeline that makes the thermochemical data library and stoichiometric tables, and the pipeline that performs abundance calculations. Given elemental abundances, TEA calculates molecular abundances for a particular temperature and pressure or a list of temperature-pressure pairs. Documentation is provided in the TEA User Manual (Bowman and Blecic) and the TEA Code Description (Blecic and Bowman) that accompany the code. Figure 3 shows the layout of the TEA program's flow. Its modules are:

1. **prepipe.py**: Runs the `readJANAF.py` and `makestoich.py` modules and provides their common setup.
2. **readJANAF.py**: Extracts relevant from all available NIST-JANAF Thermochemical Tables and writes ASCII files.
3. **makestoich.py**: Reads the chemical formula to obtain species names and their stoichiometric coefficients from each JANAF file, and elemental solar abundances from an ASCII file based on Asplund et al. (2009) Table 1. The code produces an output file containing species, stoichiometric coefficients, and abundances.
4. **runsingle.py**: Runs TEA for a single T, P pair.
5. **runatm.py**: Runs TEA over a pre-atmosphere file containing a list of T, P pairs.
6. **readatm.py**: Reads the pre-atmospheric file with multiple T, P pairs.
7. **makeheader.py**: Combines the stoichiometric information, Gibbs free energy per species at specific temperatures, and the user input to create a single file with relevant chemical informations further used by the pipeline.
8. **balance.py**: Uses species and stoichiometric information to establish viable, mass-balanced, initial mole numbers.
9. **format.py**: Auxiliary program that manages input/output operations in each piece of the pipeline.
10. **lagrange.py**: Uses data from the most recent iteration's corrected mole numbers and implements the Lagrangian method for minimization. Produces output with raw, non-corrected mole numbers for each species (values are temporarily allowed to be negative).
11. **lambdacorr.py**: Takes non-corrected mole numbers and implements lambda correction to obtain only valid, positive numbers of moles. Output is the corrected mole numbers for each species.
12. **iterate.py**: Driver program that repeats *lagrange.py* and *lambdacorr.py* until a pre-defined maximum number of iterations is met.
13. **readconfig.py**: Reads TEA configuration file.
14. **makeatm.py**: Makes pre-atmospheric file for a multiple T, P run.
15. **plotTEA.py**: Plots TEA output, the atmospheric file with final mole-fraction abundances.

6. APPLICATION TO HOT-JUPITER ATMOSPHERES

In this section, we illustrate several applications of the TEA code. We produced molecular abundances profiles for models of hot-Jupiter planetary atmospheres, given their temperature-pressure profiles.

The temperature and pressure ($T-P$) profiles adopted for our thermochemical calculations are shown in Figure 4. The left and middle panel show the $T-P$ profiles of WASP-12b from Stevenson et al. (2014a) with the C/O ratio of 0.5 and 1.2, respectively. The right panel shows the thermal profile of WASP-43b from Stevenson et al. (2014b) with solar metallicity. We chose atmospheric models with elemental-abundance profiles of C/O >1 and C/O <1 and three profiles with solar, 10 times solar, and 50 times solar elemental abundances to show the influence of the C/O ratio and metallicity on the chemistry and composition of extrasolar giant planets.

6.1. *Background of the Models Used*

WASP-12b is a first hot-Jupiter extrasolar planet that is found to have a C/O ratio larger than one (Madhusudhan et al. 2011b). Subsequent observations and analyses (Cowan et al. 2012, Bergfors et al. 2013, Crossfield et al. 2012, Swain et al. 2013) re-evaluated this result suggesting that stellar binary companions and thermal phase variations corrupted the original conclusion, leading to no evidence for a high C/O ratio. Using previously published results and a suite of inverse modeling approaches, Line et al. (2014) found high CO_2 and concluded that larger C/O ratios cannot be ruled out. Stevenson et al. (2014a) performed a uniform analysis of all available *Hubble* and *Spitzer Space Telescope* secondary eclipse data. They included additional opacity sources in the calculation, C_2H_2 and HCN, relevant and abundant in atmospheres with C/O larger than one. Their results reinstate the previous conclusion from Madhusudhan et al. (2011b), showing that a physically plausible carbon-rich solution achieves the best fit, being even 670 times more probable than the oxygen-rich solution. In our application, we used their two best-fit oxygen- and carbon-rich models (their Figure 7, left panel), with the respective C/O ratios of 0.5 and 1.2.

WASP-43b is a hot-Jupiter exoplanet, orbiting a cold K7-type star on a short-period orbit (only 19.5 hours). Due to the host star's cool temperature and small radius, as well as a small semi-major axis, the system produces deep eclipses both in transit and occultation, making WASP-43b one of the most observed and studied exoplanets today. Multiple photometric and spectroscopic observations of WASP-43b (Hellier et al. 2011, Gillon et al. 2012, Wang et al. 2013, Blecic et al. 2014, Chen et al. 2014, Zhou et al. 2014, Stevenson et al. 2014b) revealed no thermal inversion in the planetary atmosphere, low day-night energy redistribution, water abundance 1-10$\times$solar, and a slightly enhanced C/O ratio compared to solar (Benneke 2015, Blecic et al. 2015b). Stevenson et al. (2014b), in addition, constructed a longitudinally-resolved brightness temperature map as a function of the optical depth and compared their best-fit retrieved dayside temperature and pressure profile with three scenarios of self-consistent radiative equilibrium models. They showed that the best match is with the profile at the substellar point, supporting a low day-night heat redistribution. In our application, we used the

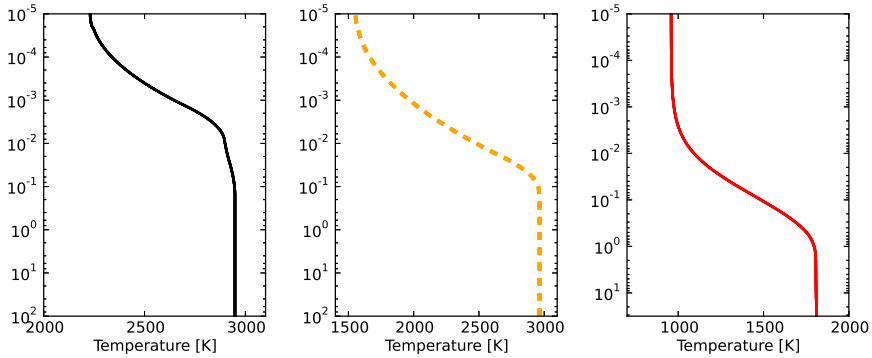

FIG. 4.— The left and middle panels show the O-rich and C-rich temperature and pressure ($T-P$) profile of WASP-12b from Stevenson et al. (2014a) with C/O = 0.5 and C/O = 1.2 respectively. The right panel shows the $T-P$ profile of WASP-43b from Stevenson et al. (2014b) with solar metallicity.

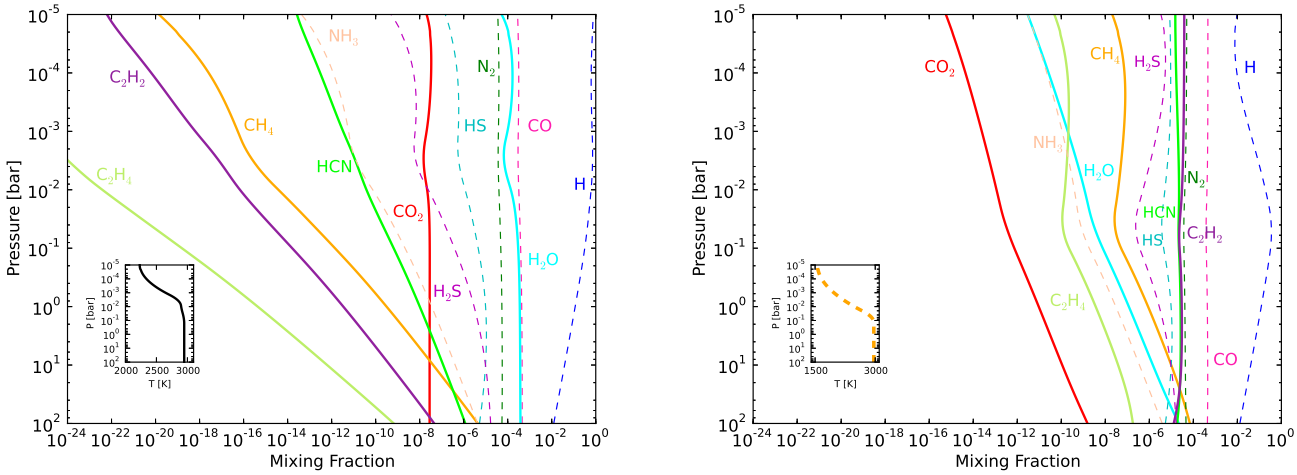

FIG. 5.— Comparison between vertical thermochemical equilibrium distributions for WASP-12b O-rich, left panel, and WASP-12b C-rich, right panel, elemental abundance profile. The inset plots are the $T-P$ profiles shown in Figure 4, left and middle panels.

retrieved dayside temperature and pressure profile from their Figure 5, Stevenson et al. (2014b), Supplemental Material.

6.2. *Results*

We adopt Asplund et al. (2009) photospheric solar abundances as our baseline. To change the elemental abundance profile, set them to a certain C/O ratio, or enhance metallicity, we use our Python routine, `makeAbun.py`. This routines is the part of the BART project and it is available to the community via `Github.com` under an open-source licence (`https://github.com/exosports/BART`). For different metallicities, the routine multiples the elemental abundances of all species except for hydrogen and helium, preserving the ratio of major atomic species like C, N, and O.

We chose to run the models for all plausible, spectroscopically active species in the infrared relevant for hot-Jupiter atmospheres: H_2, CO, CO_2, CH_4, H_2O, HCN, C_2H_2, C_2H_4, N_2, NH_3, HS, and H_2S. Our input species are: H, He, C, N, O, S.

Figure 5 shows results for WASP-12b. Each $T-P$ profile is sampled 100 times uniformly in log-pressure space. Figure 6 shows the TEA runs for WASP-43b with different metallicities. This $T-P$ profile is sampled 90 times in uniformly log-pressure space.

As expected, Figure 5 shows that H_2O, CH_4, CO, CO_2, C_2H_2, C_2H_4, and HCN are under the strong influence of the atmospheric C/O ratio in hot Jupiters (e.g., Lodders & Fegley 2002, Seager et al. 2005, Fortney et al. 2005, Madhusudhan et al. 2011b,a, Madhusudhan 2012, Madhusudhan & Seager 2011, Moses et al. 2013). These species are plotted in solid lines, while species with only small influence from the C/O ratio are plotted as dashed lines.

The results also show, as expected, that CO is a major atmospheric species on hot Jupiters for all C/O ratios and metallicities (Figures 5 and 6), because CO is chemically favored over H_2O. Other oxygen-bearing molecules like H_2O and CO_2 are more abundant when C/O<1, while CH_4, C_2H_2, and C_2H_4 become significant species when C/O>1. Species like N_2 and NH_3 that do not contain carbon or oxygen are much less affected by the C/O ratio.

H_2O is abundant in hot-Jupiter atmospheres (e.g., Burrows & Sharp 1999, Lodders & Fegley 2002, Hubeny & Burrows 2007, Sharp & Burrows 2007) due to the large solar abundances of oxygen and hydrogen. Even disequilibrium processes like photochemistry cannot deplete its abundance. Photochemical models by Moses et al. (2011) and Line et al. (2010, 2011) predict that water will be recycled in hot-Jupiter atmospheres, keeping H_2O abundances close to thermochemical equilibrium values. A low water abundance seems to occur only in atmospheres with a C/O>1.

CO_2, although present in hot-Jupiter atmospheres and spectroscopically important, is not a major constituent, and it becomes even less abundant when C/O>1. Although photochemistry can greatly enhance the HCN, C_2H_2, and C_2H_4 abundances (Moses et al. 2013), we also see that with C/O>1,

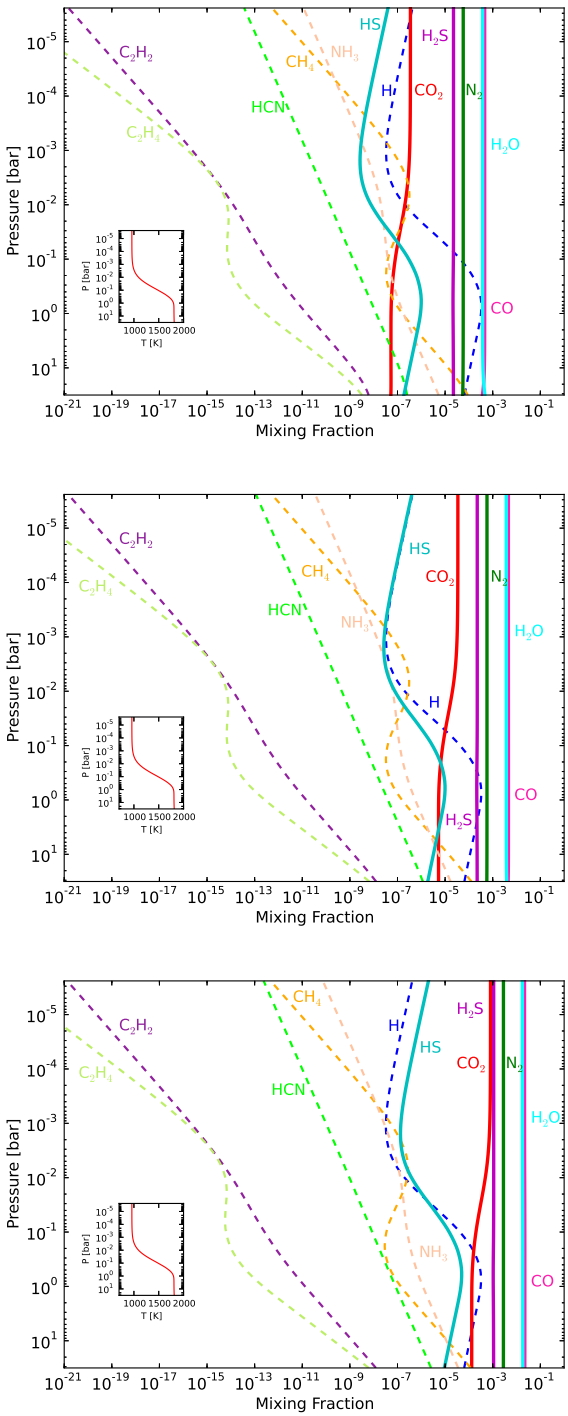

FIG. 6.— Thermochemical equilibrium vertical distributions for different metallicities of WASP-43b assuming the $T-P$ profile in Figure 4, right panel (profile given in inset). Three metallicity cases with ζ= 1, 10, and 50 are shown from the top to the bottom.

they are the most abundant constituents.

In Figure 6, the species strongly influenced by metallicity are again plotted as solid lines. In general, we see, as expected (e.g., Line et al. 2011, Lodders & Fegley 2002, Venot et al. 2014), that the shapes of the vertical distributions are mostly preserved for all metallicities. However, the thermochemical mixing ratio of CO_2, CO, H_2O, N_2, HS, and H_2S vary by several orders of magnitude over the range of metallicities, while CH_4 and hydrocarbons change very little.

When the metallicity changes from 1 to 50, the abundance of CO_2 experiences the most dramatic change. It increases by a factor of 1000, confirming it as the best probe of planetary metallicity (Lodders & Fegley 2002, Zahnle et al. 2009). CO_2 abundance is the quadratic function of metallicity (Venot et al. 2014), while CO, H_2O, HS, H_2S, and N_2 abundances, for species that either contain one metal atom or are the major reservoirs of carbon and nitrogen, increase linearly with metallicity (Visscher et al. 2006). For this metallicity range, the CO, H_2O, HS, H_2S, and N_2 abundances change by a factor of 100, while NH_3, CH_4, C_2H_2, C_2H_4, and HCN change by a factor of 10 or less.

7. COMPARISON TO OTHER METHODS

To test the validity of our code, we performed 4 different tests. We compared the output of TEA with the example from White et al. (1958) using their thermodynamic data. We also compared the TEA output with the output of our TEBS (Thermochemical Equilibrium by Burrows & Sharp) code that implements the Burrows & Sharp (1999) analytical method for calculating the abundances of five major molecular species present in hot-Jupiter atmospheres (CO, CH4, H2O, N2, NH3). As another comparison, we used the free thermochemical equilibrium code CEA (Chemical Equilibrium with Applications, available from NASA Glenn Research Center at `http://www.grc.nasa.gov/WWW/CEAWeb/`). This code uses the Newton-Raphson descent method within the Lagrange optimization scheme to solve for chemical abundances. Their approach is described by Gordon & McBride (1994), McBride & Gordon (1996), and Zeleznik & Gordon (1960, 1968). The thermodynamic data included in the CEA code are partially from the JANAF tables (Chase 1986) that we used in our TEA code, but also from numerous other sources (e.g., Cox et al. 1982, Gurvich et al. 1989, McBride et al. 1993). Lastly, we derived CEA free energies and used them as input to TEA, to compare the CEA and TEA outputs.

Our first comparison was done using the example from White et al. (1958). We determined the composition of the gaseous species arising from the combustion of a mixture of hydrazine, N_2H_4, and oxygen, O_2, at T = 3500 K and the pressure of 750 psi = 51.034 atm. We used the free-energy functions and b_j values (total number of moles of element j originally present in the mixture) from their Table 1. We reproduced their abundances, Table 1, with slightly higher precision probably due to our use of double precision.

TABLE 1
COMPARISON WHITE ET AL. *vs.* TEA

Species	$\frac{g_i^0(T)}{RT}$	White et al. abundances	TEA abundances	Difference
H	-10.021	0.040668	0.04065477	-0.00001323
H_2	-21.096	0.147730	0.14771009	-0.00001991
H_2O	-37.986	0.783153	0.78318741	0.00003441
N	-9.846	0.001414	0.00141385	-0.00000015
N_2	-28.653	0.485247	0.48524791	0.00000091
NH	-18.918	0.000693	0.00069312	0.00000012
NO	-28.032	0.027399	0.02739720	-0.00000180
O	-14.640	0.017947	0.01794123	-0.00000577
O_2	-30.594	0.037314	0.03730853	-0.00000547
OH	-26.111	0.096872	0.09685710	0.00001490

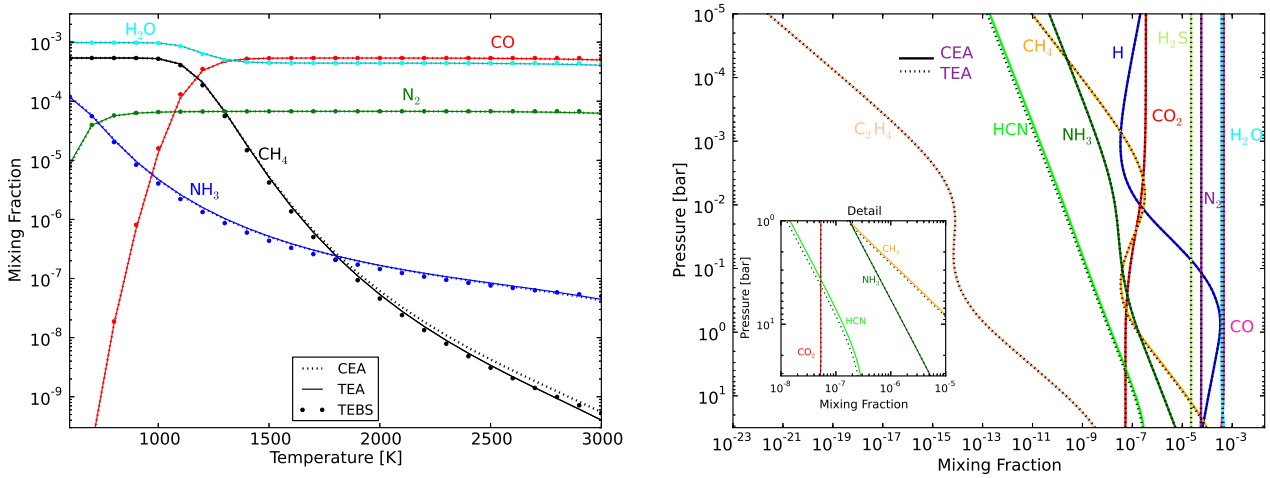

FIG. 7.— **Left:** Comparison TEA, CEA and TEBS. TEBS is an analytic method, while CEA and TEA are numerical methods. We show the major spectroscopically-active species in the infrared that can be produced by all three methods. We run the codes for the same range of temperatures and the pressure of $P = 1$ bar. Each species in each method is plotted with a different line style, but with the same color. The TEBS final abundances are plotted as dots, CEA as dashed lines, while TEA is plotted as solid lines. **Right:** Comparison of the TEA results with the results from CEA. CEA and TEA are both numerical methods that use Gibbs free energy minimization method with similar optimization scheme. We show the most plausible and most abundant spectroscopically-active species in the infrared expected to be present in hot-Jupiter atmospheres, that all codes can cover. In the inset plot, we show a detail (zoom-in part), pointing out species lines that do not overlap. The $T-P$ profile used for this run is given in the right panel of Figure 4. Tables 2 and 3 list differences between the final abundances for random three T, P points chosen from each run.

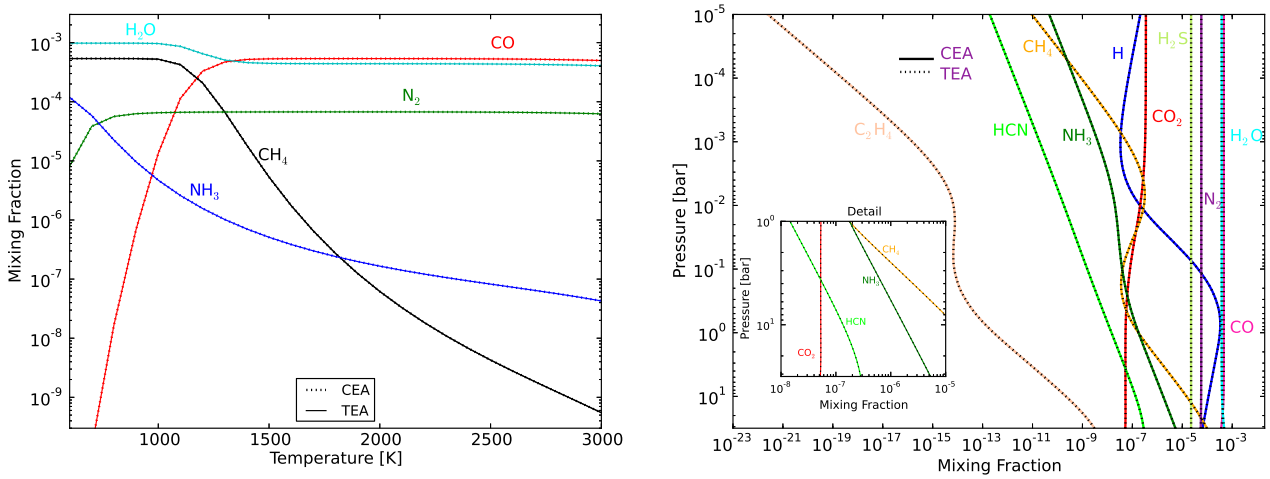

FIG. 8.— Comparison TEA and CEA using CEA thermodynamic data provided in their `thermo.inp` file. The comparison is done for the same conditions as in Figure 7. Tables 2 and 3 list differences between the final abundances for random three T, P points chosen from each run.

Figure 7, left panel, shows the CEA, TEA, and TEBS runs for the temperatures between 600 and 3000 K, pressure of 1 bar, and solar abundances. The runs were performed with the input and output species that all codes contain (H, C, O, N, H_2, CO, CH_4, H_2O, N_2, NH_3). We also run the comparison just between CEA and TEA, Figure 7, right panel, for the WASP-43b model atmosphere that we described in Section 6. We used the pressure and temperature profile shown in Figure 4, right panel, and solar elemental abundances. The temperatures and pressures range from 958.48 to 1811.89 K and 1.5×10^{-5} to 3.1623×10^{1} bar, respectively. We included the same species as in Section 6 with the exclusion of the C_2H_2 and HS species, because CEA does not carry the thermodynamical parameters for them.

In the left panel of Figure 7, we see that for the most species and temperatures CEA and TEA lines overlap (CEA result is plotted in dashed and TEA in solid lines). However, CH_4 species abundances above $T \sim 1700$ K do not overlap. TEBS colored dots do not overplot either CEA or TEA curves, but follow them closely. This method is derived for only five major molecular species and is based on a few simple analytic expressions.

In Figure 7, right panel, we again see that most species overlap, except HCN and CH_4. The HCN curves (for CEA and TEA runs) differ for the full temperature range (see the inset figure), while, as before, CH_4 curve differs slightly only for pressures above $\sim$0.1 bar and temperatures above $\sim$1700 K (see Figure 4 for the $T-P$ profile used for this run).

The differences seen in Figure 7 come from the different sources of thermodynamic data used for CEA and TEA (see Tables 2 and 3). When the CEA thermodynamic data are used as input to TEA, all species final abundances match, see Fig-

TABLE 2
DIFFERENCES CEA *vs.* TEA, FIGURES 7 AND 8, LEFT PANELS

Pressure	Temp	Species				
(bar)	(K)	CO	CH4	H2O	N2	NH3
CEA free energies						
1.0000e+00	2500.00	-33.80559930	-34.91655970	-40.12912409	-27.71757996	-32.70878374
1.0000e+00	2700.00	-33.69182758	-35.33414843	-39.64186320	-27.99507610	-33.05270703
1.0000e+00	2900.00	-33.61649725	-35.76252669	-39.25604592	-28.25692510	-33.39666924
TEA (JANAF) free energies						
1.0000e+00	2500.00	-33.80793700	-34.70780992	-40.12426098	-27.72037451	-32.73695542
1.0000e+00	2700.00	-33.69214791	-35.08662806	-39.63255004	-27.99496246	-33.08302621
1.0000e+00	2900.00	-33.61466712	-35.47533692	-39.24191944	-28.25439783	-33.42896561
CEA final abundances						
1.0000e+00	2500.00	5.3129e-04	4.2666e-09	4.3546e-04	6.6686e-05	8.2252e-08
1.0000e+00	2700.00	5.2311e-04	1.8387e-09	4.2855e-04	6.5661e-05	6.4332e-08
1.0000e+00	2900.00	5.0876e-04	8.2340e-10	4.1586e-04	6.3844e-05	4.9466e-08
TEA final abundances using CEA free energies						
1.0000e+00	2500.00	5.3129e-04	4.2665e-09	4.3547e-04	6.6685e-05	8.2251e-08
1.0000e+00	2700.00	5.2311e-04	1.8387e-09	4.2856e-04	6.5661e-05	6.4332e-08
1.0000e+00	2900.00	5.0876e-04	8.2339e-10	4.1586e-04	6.3844e-05	4.9466e-08
TEA final abundances using JANAF free energies						
1.0000e+00	2500.00	5.3129e-04	3.3976e-09	4.3547e-04	6.6685e-05	8.3987e-08
1.0000e+00	2700.00	5.2312e-04	1.4194e-09	4.2856e-04	6.5661e-05	6.6260e-08
1.0000e+00	2900.00	5.0878e-04	6.1471e-10	4.1586e-04	6.3845e-05	5.1339e-08

TABLE 3
DIFFERENCES CEA *vs.* TEA, FIGURES 7 AND 8, RIGHT PANELS

Pressure	Temp	Species						
(bar)	(K)	CO	CO2	CH4	H2O	HCN	NH3	H2S
CEA free energies								
3.8019e-01	1719.64	-34.9307244	-58.4124145	-33.595356	-43.7404858	-19.8124391	-31.4721776	-30.6054338
1.6596e+00	1805.28	-34.7237371	-57.3627896	-33.695587	-43.1403205	-20.4938031	-31.5886558	-30.7577826
2.1878e+01	1810.15	-34.7128505	-57.3065771	-33.701803	-43.1083097	-20.5310828	-31.5955222	-30.7664570
TEA (JANAF) free energies								
3.8019e-01	1719.64	-34.9288052	-58.4130468	-33.5180429	-43.7386156	-19.6678405	-31.4830322	-30.5760402
1.6596e+00	1805.28	-34.7231685	-57.3648328	-33.6061366	-43.1392058	-20.3573682	-31.6020990	-30.7285988
2.1878e+01	1810.15	-34.7123566	-57.3086978	-33.6116396	-43.1072342	-20.3950903	-31.6091097	-30.7372812
CEA final abundances								
3.8019e-01	1719.64	4.5960e-04	5.8035e-08	4.9221e-08	3.7681e-04	5.6243e-09	7.9716e-08	2.2498e-05
1.6596e+00	1805.28	4.5918e-04	5.2864e-08	4.4665e-07	3.7724e-04	2.4131e-08	2.8864e-07	2.2504e-05
2.1878e+01	1810.15	4.0264e-04	5.3052e-08	5.6873e-05	4.3396e-04	2.3851e-07	3.7082e-06	2.2520e-05
TEA final abundances using CEA free energies								
3.8019e-01	1719.64	4.5959e-04	5.8035e-08	4.9219e-08	3.7682e-04	5.6240e-09	7.9714e-08	2.2498e-05
1.6596e+00	1805.28	4.5918e-04	5.2865e-08	4.4667e-07	3.7724e-04	2.4131e-08	2.8864e-07	2.2504e-05
2.1878e+01	1810.15	4.0263e-04	5.3053e-08	5.6875e-05	4.3396e-04	2.3851e-07	3.7083e-06	2.2519e-05
TEA final abundances using JANAF free energies								
3.8019e-01	1719.64	4.5959e-04	5.8326e-08	4.5480e-08	3.7681e-04	4.8604e-09	8.0472e-08	2.2497e-05
1.6596e+00	1805.28	4.5922e-04	5.3200e-08	4.0512e-07	3.7719e-04	2.0937e-08	2.9102e-07	2.2504e-05
2.1878e+01	1810.15	4.0694e-04	5.3429e-08	5.2592e-05	4.2965e-04	2.1124e-07	3.7386e-06	2.2519e-05

ure 8. Section 7.1, below, elaborates on this and investigate the difference in free energy input values used for CEA and TEA.

7.1. *Comparison of free energy values in CEA and TEA*

The thermodynamic data used for CEA are in the form of polynomial coefficients, and are listed in the `termo.inp` file provided with the CEA code. The format of this library is explained in Appendix A of McBride & Gordon (1996). For each species, the file lists, among other data, the reference sources of the thermodynamic data, the values of the standard enthalpy of formation, $\Delta_f H^0_{298}$, at the reference temperature of 298.15 K and pressure of 1 bar, and coefficients of specific heat, C^0_p, with integration constants for enthalpy, H^o, and entropy, S^o, for temperature intervals of 200 to 1000 K, 1000 to 6000 K, and 6000 to 20000 K.

The JANAF tables list the reference sources of their thermodynamic parameters in Chase (1986, 1998). The data are also available at `http://kinetics.nist.gov/janaf/`.

The difference in thermodynamic parameters between CEA and TEA is noticeable even in their $\Delta_f H^0_{298}$ values. The source of standard enthalpies of formation in CEA for, e.g., HCN and CH_4 is Gurvich (1991), page 226 and 36, respectively, and their respective values are 133.08 and -74.60 kJ/mol. The source of standard enthalpies of formation in the JANAF tables is listed in Chase (1986) on page 600 and

615, respectively, and their respective values are 135.14 and -74.873 kJ/mol.

TEA uses JANAF tables to calculate the values of free energies for each species following Equation 10. To calculate the values of free energies used in CEA, we started from Chapter 4 in Gordon & McBride (1994). Our goal is to plug CEA free energies into TEA and test whether TEA will produce the same final abundances as CEA does.

As explained in Section 4.2, the thermodynamic functions specific heat, enthalpy, and entropy as function of temperatures are given as:

$$\frac{C_p^o}{R} = \sum a_i T^{q_i}\,, \tag{35}$$

$$\frac{H^o}{RT} = \frac{\int C_p^o\, dT}{RT}\,, \tag{36}$$

$$\frac{S^o}{R} = \int \frac{C_p^o}{RT}\, dT\,. \tag{37}$$

These functions are given in a form of seven polynomial coefficients for specific heat, C_p^o/R, and two integrations constants (a_8 and a_9) for enthalpy, H^o/RT, and entropy, S^o/R:

$$\frac{C_p^o}{R} = a_1T^{-2} + a_2T^{-1} + a_3 + a_4T + a_5T^2 + a_6T^3 + a_7T^4\,, \tag{38}$$

$$\frac{H^o}{RT} = -a_1T^{-2} + a_2T^{-1}\,lnT + a_3 + a_4\frac{T}{2} + a_5\frac{T^2}{3} + a_6\frac{T^3}{4} + a_7\frac{T^4}{5} + \frac{a_8}{T}\,, \tag{39}$$

$$\frac{S^o}{R} = -a_1\frac{T^{-2}}{2} - a_2T^{-1} + a_3\,lnT + a_4T + a_5\frac{T^2}{2} + a_6\frac{T^3}{3} + a_7\frac{T^4}{4} + a_9\,. \tag{40}$$

To derive free energies in the form that TEA uses them, we rewrite Equation 10 for one species as:

$$\frac{g^0(T)}{RT} = 1/R\left[\frac{G_T^0 - H_{298}^0}{T}\right] + \frac{\Delta_f H_{298}^0}{RT}\,, \tag{41}$$

The first term on the right side can be expressed in the following format (Chase et al. 1974, Page 3):

$$\frac{G_T^0(T) - H_{298}^0}{T} = -S_T^o + \frac{(H_T^o - H_{298}^0)}{T}\,. \tag{42}$$

Thus, we rewrite Equation 41 as:

$$\frac{g^0(T)}{RT} = 1/R\left[S_T^o + \frac{(H_T^o - H_{298}^0)}{T}\right] + \frac{\Delta_f H_{298}^0}{RT}\,, \tag{43}$$

$$\frac{g^0(T)}{RT} = 1/R\left[S_T^o + \frac{H_T^o}{T} - \frac{H_{298}^0}{T}\right] + \frac{\Delta_f H_{298}^0}{RT}\,, \tag{44}$$

To see Equations 39 and 40 inside Equation 44, we multiply and divide the first and second term on the right with R and get:

$$\frac{g^0(T)}{RT} = \frac{S_T^o}{R} + \frac{H_T^o}{RT} - \frac{H_{298}^0}{RT} + \frac{\Delta_f H_{298}^0}{RT}\,, \tag{45}$$

In the CEA analysis paper, Section 4.1, Gordon & McBride (1994) state that they have arbitrary assumed $H^o(298.15) = \Delta_f H^o(298.15)$. Adopting this assumption leads to:

$$\frac{g^0(T)}{RT} = \frac{S_T^o}{R} + \frac{H_T^o}{RT} - \frac{\Delta_f H_{298}^0}{RT} + \frac{\Delta_f H_{298}^0}{RT}\,. \tag{46}$$

The last two terms cancel leading to a simple expression for free energies:

$$\frac{g^0(T)}{RT} = \frac{S_T^o}{R} + \frac{H_T^o}{RT}\,, \tag{47}$$

The first term on the right side is Equation 40, while the second term is Equation 39; expressions with polynomial coefficients that are given in the CEA `thermo.inp` file.

Following the last conclusion, we calculated the free energies for each species of interest and used them as input to TEA. Figure 8 shows the comparison between CEA and TEA using CEA free energies. We see that all species overlap. Tables 2 and 3 give the exact values of free energies used and the final abundances for several (T, P) points that showed the largest differences between CEA and TEA runs in Figure 7. It also lists the free energies calculated using JANAF tables and the final abundances produced by TEA using JANAF thermodynamic data.

As seen in Figure 8, although CEA uses Newton-Raphson and TEA the Lagrangian method of steepest descent, both approaches, using the same inputs (free energies), find the same final abundances. Table 2, (groups *CEA final abundances* and *TEA final abundances using CEA free energies*), shows values identical for most species between the two tests. A few cases show that abundance ratios are inconsistent at the 10^{-5} level. Table 3 displays the same trend. The differences in the fifth decimal place may indicate that, somewhere in CEA, a calculation is carried out in 32-bit precision, possibly due to a literal single-precision number in the source code. Python floating literals are in 64-bit precision by default.

8. REPRODUCIBLE RESEARCH LICENSE

Reproducing a lengthy computation, such as that implemented in TEA, can be prohibitively time consuming (Stodden 2009). We have released TEA under an open-source license, but this is not enough, as even the most stringent of those licenses (e.g., the GNU General Public License) does not require disclosure of modifications if the researcher does not distribute the code. So that the process of science can proceed efficiently, there are several terms in our license to ensure reproducibility of all TEA results, including those from derivative codes. A key term requires that any reviewed scientific publication using TEA or a derived code must publish that code, the code input and output used in the paper (such as data in tables and figures, and data summarized in the text), and all the information used to initialize the code to produce those outputs. These items must appear in a Reproducible Research Compendium (RRC). The RRC must be published

with the paper, preferably in a permanent, free-of-charge (to the downloader), public internet archive, such as zenodo.org (note that zenodo.com is a different site). The American Astronomical Society recommends this site because it indexes its content for online searches, making it highly "discoverable". A permanent link to the archive must be published in the paper, and the archive must never be closed, altered, or charged for. Details and examples of how to do this appear in the license and documents accompanying TEA, along with additional discussion. The RRC for this paper, including the TEA package and documentation, is available at Blecic (2016, zenodo.org: `RRC-BlecicEtal-2016-ApJS-TEA`). See http://planets.ucf.edu/resources/reproducible for further discussion of the Reproducible Research Software License, its latest version, how to make a good RRC, and a discussion group.

An unreviewed version of this paper (Blecic et al. 2015a) also has an RRC, which was posted in May 2015 at `https://github.com/dzesmin/RRC-BlecicEtal-2015a-ApJS-TEA/`. This initial RRC carries a test version of the TEA code under a more restrictive, pre-publication license. We now understand that using github.com and other versioning sites is less than optimal, as they have poorer discoverability and are oriented toward making changes rather than historical archiving.

9. CONCLUSIONS

We have developed an open-source Thermochemical Equilibrium Abundances code for gaseous molecular species. Given elemental abundances and one or more temperature-pressure pairs, TEA produces final mixing fractions using the Gibbs-free-energy minimization method with an iterative Lagrangian optimization scheme.

We applied the TEA calculations to several hot-Jupiter $T-P$ models, with expected results. The code is tested against the original method developed by White et al. (1958), the analytic method developed by Burrows & Sharp (1999), and the Newton-Raphson method implemented in the free Chemical Equilibrium with Applications code. Using the free energies listed in White et al. (1958), their example, and derived free energies based on the thermodynamic data provided in CEA's `thermo.inp` file, TEA produces the same final abundances, but with higher precision.

Currently, TEA is specialized for gaseous species, with the implementation of condensates left for future work. In opacity calculations at low temperatures (below 1000 K), the inclusion of condensates is necessary as it reduces the gas phase contribution to opacity (e.g., Sharp & Huebner 1990, Lodders & Fegley 2002, Burrows & Sharp 1999).

The thermochemical equilibrium abundances obtained with TEA can be used in all static atmospheres, atmospheres with vertical transport and temperatures above 1200 K (except when ions are present), and as a starting point in models of gaseous chemical kinetics and abundance retrievals run on spectroscopic data. TEA is currently used to initialize the atmospheric retrial calculations in the open-source BART project (available at `https://github.com/exosports/BART`).

TEA is written in a modular way using the Python programming language. It is documented (the Start Guide, the User Manual, the Code Document, and this theory paper are provided with the code), actively maintained, and available to the community via the open-source development sites `https://github.com/dzesmin/TEA` and `https://github.com/dzesmin/TEA-Examples`.

This project was completed with the support of the NASA Earth and Space Science Fellowship Program, grant NNX12AL83H, held by Jasmina Blecic, PI Joseph Harrington, and through the Science Mission Directorate's Planetary Atmospheres Program, grant NNX12AI69G. We would like to thank Julianne Moses for useful discussions, and Kevin B. Stevenson and Michael R. Line for temperature and pressure profiles. We also thank contributors to SciPy, NumPy, Matplotlib, and the Python Programming Language; the open-source development website GitHub.com; and other contributors to the free and open-source community.

REFERENCES

Allard, F., & Hauschildt, P. H. 1995, ApJ, 445, 433
Asplund, M., Grevesse, N., Sauval, A. J., & Scott, P. 2009, ARA&A, 47, 481
Bahn, G. S., & Zukoski, E. E. 1960, Kinetics, Equilibria and Performance of High Temperature Systems: Proceedings of the First Conference, Los Angeles California 2-5 November 1959 (Butterworths)
Belford, R. L., & Strehlow, R. A. 1969, Annual Review of Physical Chemistry, 20, 247
Benneke, B. 2015, arXiv preprint arXiv:1504.07655
Bergfors, C., Brandner, W., Daemgen, S., et al. 2013, MNRAS, 428, 182
Blecic, J. 2016, RRC-BlecicEtal-2016-ApJS-TEA, v.1.0, Zenodo, doi:TBD
Blecic, J., Harrington, J., & Bowman, M. O. 2015a, ArXiv e-prints, arXiv:1505.06392
Blecic, J., Harrington, J., Cubillos, P., Maddison, S., & Foster, A. 2015b, in preparation
Blecic, J., Harrington, J., Madhusudhan, N., et al. 2014, ApJ, 781, 116
Burrows, A., & Sharp, C. M. 1999, ApJ, 512, 843
Chase, Jr., M. 1998, Journal of Physical and Chemical Reference Data, Monograph 9, update of 3rd Edition
Chase, M. W. 1986, JANAF thermochemical tables
Chase, M. W., Curnutt, J. L., Hu, A., et al. 1974, Journal of Physical and Chemical Reference Data, 3, 311
Chen, G., van Boekel, R., Wang, H., et al. 2014, A&A, 563, A40
Cowan, N. B., Machalek, P., Croll, B., et al. 2012, ApJ, 747, 82
Cox, J. D., Wagman, D. D., & Medvedev, V. A. 1982, Hemisphere Publishing Corp. New York
Crossfield, I. J. M., Barman, T., Hansen, B. M. S., Tanaka, I., & Kodama, T. 2012, ApJ, 760, 140
Eriksson, G. 1971, Acta Chem.Scand.
Fortney, J. J., Marley, M. S., Lodders, K., Saumon, D., & Freedman, R. 2005, ApJ, 627, L69
Gillon, M., Triaud, A. H. M. J., Fortney, J. J., et al. 2012, A&A, 542, A4
Gordon, S., & McBride, B. J. 1994, Computer Program for Calculation of Complex Chemical Equilibrium Compositions and Applications. I. Analysis, Reference Publication RP-1311, NASA, describes theory and numerical algorithms behind CEA computer program
Gurvich, L. V., Veyts, I. V., & Alcock, C. B. 1989, Hemisphere Publishing Corp. New York, 1, 1
Gurvich, V. 1991in , 721–726
Hellier, C., Anderson, D. R., Collier Cameron, A., et al. 2011, A&A, submitted
Hubeny, I., & Burrows, A. 2007, ApJ, 669, 1248
Kopparapu, R. K., Kasting, J. F., & Zahnle, K. J. 2012, ApJ, 745, 77
Lauretta, D. S., Lodders, K., & Fegley, Jr., B. 1997, Science, 277, 358
Line, M. R., Knutson, H., Wolf, A. S., & Yung, Y. L. 2014, ApJ, 783, 70
Line, M. R., Liang, M. C., & Yung, Y. L. 2010, ApJ, 717, 496
Line, M. R., Vasisht, G., Chen, P., Angerhausen, D., & Yung, Y. L. 2011, ApJ, 738, 32
Line, M. R., & Yung, Y. L. 2013, ApJ, 779, 3
Lodders, K., & Fegley, B. 2002, Icarus, 155, 393
Lodders, K., & Fegley, Jr., B. 1993, Meteoritics, 28, 387
Madhusudhan, N. 2012, ApJ, 758, 36
Madhusudhan, N., Mousis, O., Johnson, T. V., & Lunine, J. I. 2011a, ApJ, 743, 191
Madhusudhan, N., & Seager, S. 2011, ApJ, 729, 41
Madhusudhan, N., Harrington, J., Stevenson, K. B., et al. 2011b, Nature, 469, 64
Marley, M. S., & Robinson, T. D. 2015, ARA&A, 53, 279

McBride, B. J., & Gordon, S. 1996, Computer Program for Calculation of Complex Chemical Equilibrium Compositions and Applications. II. User's Manual and Program Description, Reference Publication RP-1311-P2, NASA
McBride, B. J., Gordon, S., & Reno, M. A. 1993, Coefficients for Calculating Thermodynamic and Transport Properties of Individual Species, Technical Memorandum TM-4513, NASA, this describes the pre-1994 7-coefficient fit, which is used in Cantera.
Miller, J. A., Kee, R. J., & Westbrook, C. K. 1990, Annual Review of Physical Chemistry, 41, 345
Moses, J. I., Madhusudhan, N., Visscher, C., & Freedman, R. S. 2013, ApJ, 763, 25
Moses, J. I., Visscher, C., Fortney, J. J., et al. 2011, ApJ, 737, 15
Seager, S., Richardson, L. J., Hansen, B. M. S., et al. 2005, ApJ, 632, 1122
Sharp, C. M., & Burrows, A. 2007, ApJS, 168, 140
Sharp, C. M., & Huebner, W. F. 1990, ApJS, 72, 417
Stevenson, K. B., Bean, J. L., Madhusudhan, N., & Harrington, J. 2014a, ApJ, 791, 36
Stevenson, K. B., Désert, J.-M., Line, M. R., et al. 2014b, Science, 346, 838
Stodden, V. 2009, Computing in Science & Engineering, 11, 35
Swain, M., Deroo, P., Tinetti, G., et al. 2013, Icarus, 225, 432
Tsuji, T., Ohnaka, K., Aoki, W., & Nakajima, T. 1996, A&A, 308, L29
Venot, O., Agúndez, M., Selsis, F., Tessenyi, M., & Iro, N. 2014, A&A, 562, A51
Venot, O., Hébrard, E., Agúndez, M., et al. 2012, A&A, 546, A43
Visscher, C., Lodders, K., & Fegley, Jr., B. 2006, ApJ, 648, 1181
—. 2010a, ApJ, 716, 1060
Visscher, C., Moses, J. I., & Saslow, S. A. 2010b, Icarus, 209, 602
Wang, W., van Boekel, R., Madhusudhan, N., et al. 2013, ApJ, 770, 70
White, W. B., Johnson, S. M., & Dantzig, G. B. 1958, J. Chem. Phys., 28, 751
Zahnle, K., Marley, M. S., Freedman, R. S., Lodders, K., & Fortney, J. J. 2009, ApJ, 701, L20
Zeleznik, F. J., & Gordon, S. 1960, An analytical investigation of three general methods for of calculating chemical equilibrium compositions, Technical Note TN-473, NASA
—. 1968, Calculation of complex chemical equilibria, Technical Memorandum TM X-52303, NASA
Zhou, G., Bayliss, D. D. R., Kedziora-Chudczer, L., et al. 2014, MNRAS, 445, 2746
Zolotov, M. Y., & Fegley, B. 1998, Icarus, 132, 431