APPLICATION OF THERMAL WIND EQUATIONS TO THE JOVIAN TROPOSPHERE & STRATOSPHERE

A thesis presented to the faculty of
San Francisco State University
In partial fulfillment of
The Requirements for
The Degree

Master of Science
In
Physics

by

Richard Caleb McWhirter

San Francisco, California

December 2018

CERTIFICATION OF APPROVAL

I certify that I have read *APPLICATION OF THERMAL WIND EQUATIONS TO THE JOVIAN TROPOSPHERE & STRATOSPHERE* by Richard Caleb McWhirter and that in my opinion this work meets the criteria for approving a thesis submitted in partial fulfillment of the requirements for the degree: Master of Science in Physics at San Francisco State University.

Joseph Barranco
Associate Professor of Physics & Astronomy

Kimberly Coble
Professor of Physics & Astronomy

Adrienne Cool
Professor of Physics & Astronomy

APPLICATION OF THERMAL WIND EQUATIONS TO THE JOVIAN TROPOSPHERE & STRATOSPHERE

Richard Caleb McWhirter
San Francisco State University
2018

In this thesis we use zonal velocity data at the "cloud layer," the visible atmospheric surface of Jupiter, and temperature data at 10 constant pressure surfaces above the cloud layer to numerically integrate two versions of the thermal wind equation to determine the mean zonal (east-west) velocity (u) on Jupiter as a function of latitude (λ) and pressure (p) in the region $-60° < \lambda < 60°$, $1 < p < 1000$ mbar. These calculations are performed in the context of the debate regarding the nature of Jupiter's jets. We determined that our results are valid far from the equator ($|\lambda| \gtrsim 30°$) and suggest that the jets in this region are as deep as the geostrophic approximation holds, and that near the equator the data is too noisy to quantitatively trust our results, let alone extrapolate below 1000 mbar. Nevertheless, our results at the equator are qualitatively consistent with observations of the quasiquadrennial oscillation in the upper stratosphere.

I certify that the Abstract is a correct representation of the content of this thesis.

Chair, Thesis Committee — 12/18/2018 Date

ACKNOWLEDGMENTS

Thanks to my advisor Dr. Joseph Barranco for opening the door of computational physics for me and for his guidance through every step on this journey, to Dr. Philip Marcus at UC Berkeley for his guidance, and to Andrew Sanville at UC Berkeley for his help with the nitty gritty math and MATLAB stuff. Special thanks to my dad Jim McWhirter for inspiring me to be a physicist from a young age, to my mom Carol McWhirter for teaching me everything I know, and to my wife Jodi McWhirter for inspiring and helping me to be better in every way.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

Introduction

The east-west bands on Jupiter are easily recognizable and have been observed since Giovanni Cassini's observations in the 1660s, roughly 50 years after Galileo's discovery of Jupiter's 4 largest moons. Through centuries of observations, the large scale structure of the bands and the Great Red Spot (GRS) have remained incredibly steady despite the highly turbulent nature of the flows. Scientists have long been able to observe Jupiter's bands by eye due to the sharp contrast between the bands' differing colors, but precision measurements of the latitudinal variation of the zonal (azimuthal) wind velocities were first made by the Voyager probes in the early 1980s.

Despite years of qualitative and quantitative observations, how the bands form and how deep they extend into the planet remains a topic of debate today. The debate is fueled by the use of simulations with different sets of approximations and assumptions, both physical and numerical. What started as a goal to perform our own simulations turned into a small set of calculations to assess more information

about the atmosphere of Jupiter before jumping into simulations.

To lay the groundwork, we derive the thermal wind equations and discuss their validity on Jupiter in Section 2.2. From these equations, we discuss two leading theories on the nature of the jets in Section 2.3, one that says the jets are deep and the other that they are shallow. We also highlight observations of equatorial temperature and zonal wind oscillations high in the stratosphere, which resemble oscillations in Earth's stratosphere, in Section 2.4.

Our contribution is to numerically integrate the thermal wind equations, obtaining the zonal velocities in the troposphere and stratosphere of Jupiter. Our integration technique is formulated in Section 3.1, and we discuss our results in Section 3.2. Finally, we turn a critical eye towards our work and discuss how our results should, and should not, be interpreted in Section 3.3.

Chapter 2

Theoretical Background

In this chapter, we derive the thermal wind equation from the momentum equation for a continuous fluid and explain the leading theories on the nature of Jupiter's zones and bands.

2.1 Coordinate System and Hydrodynamic Variables

Throughout this work, we use latitude-longitude (lat-lon) coordinates (r, λ, ϕ) corresponding to radial distance, latitude, and longitude, respectively, with unit vectors $(\hat{r}, \hat{\lambda}, \hat{\phi})$ and velocity field $\mathbf{v} = (w, v, u)$. At times, we also use Cartesian coordinates with the orientation dependent on the context. Flow variables are pressure p, density ρ, temperature T, and potential temperature $\theta = T\left(\frac{p_0}{p}\right)^{R/C_p}$, where p_0 is a reference pressure and C_p is the specific heat at constant pressure. We assume an an ideal gas, and thus our equation of state is the ideal gas law $p = \rho RT$, where $R \equiv C_p - C_V$ is the specific gas constant.

The momentum equation of motion in a rotating coordinate frame is

$$\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t} = \frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v}\cdot\nabla)\mathbf{v} = -2\boldsymbol{\Omega}\times\mathbf{v} - \frac{\boldsymbol{\nabla} p}{\rho} + \boldsymbol{\nabla}\Phi + \nu\nabla^2\mathbf{v}, \tag{2.1}$$

where $\boldsymbol{\Omega}$ is the planetary rotational angular frequency, Φ is a potential by which conservative forces can be written, and ν is the kinematic viscosity. For the purposes of our work, $\boldsymbol{\nabla}\Phi = \mathbf{g} = -g\hat{\boldsymbol{r}}$, the effective gravitational acceleration.

2.2 Geostrophic Motion & Thermal Wind Equations

Let L be a characteristic horizontal length scale, i.e. a characteristic density variation scale, and U be a characteristic horizontal velocity scale. The advective time scale is thus $\tau = L/U$. When considering large scale planetary flows, it is useful to compare the nonlinear terms to the Coriolis term in the momentum equation,

$$\frac{\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t}}{2\boldsymbol{\Omega}\times\mathbf{v}} \sim \frac{U/\tau}{2\Omega U\sin\lambda} = \frac{U}{2\Omega L\sin\lambda} \equiv \mathrm{Ro}. \tag{2.2}$$

This dimensionless scaling ratio is called the Rossby number, and the fluid is considered "large scale" if $\mathrm{Ro} \leq 1$. The smaller the Rossby number is, the more (less) the rotation (advection) influences the overall behavior of the flow. For Jupiter, $U \sim 100\ \mathrm{ms}^{-1}$, $L \sim 10^4$ km, and $2\Omega \sim 3.5\cdot 10^{-4}\ \mathrm{s}^{-1}$, which gives $\mathrm{Ro} \sim 0.03$,[1] so

[1]For reference, the bands on Jupiter have peak wind speeds from 50–150 ms^{-1} and have widths from 5–$20\cdot 10^3$ km. The Great Red Spot is $\sim 16{,}000$ km in diameter. The angular frequency Ω

the Coriolis acceleration considerably dominates the motion.

Another comparison often made in large scale planetary flows is the ratio of the viscous term to the Coriolis term, a dimensionless number called the Ekman number:

$$\frac{\nu\nabla^2\mathbf{v}}{2\boldsymbol{\Omega}\times\mathbf{v}} \sim \frac{\nu U/L^2}{2\Omega U} = \frac{\nu}{2\Omega L^2} \equiv \mathrm{E}. \tag{2.3}$$

The kinematic viscosity is often on the order of 10^{-6} $\mathrm{m^2 s^{-1}}$, and thus for Jupiter $E \sim 10^{-11}$; this suggests that, to a good approximation, the atmosphere is frictionless.

The geostrophic approximation is to take the limit as $\mathrm{Ro} \to 0$ and $\mathrm{E} \to 0$ and furthermore to assume that the vertical motion is much smaller than the horizontal motion, or, equivalently, that the aspect ratio of the flow is small. Dropping the nonlinear and viscous terms from (2.1),

$$-\rho\, 2\boldsymbol{\Omega}\times\mathbf{v} - \boldsymbol{\nabla}p + \rho\,\mathbf{g} = 0. \tag{2.4}$$

is calculated from the sidereal period of Jupiter's magnetosphere (referred to as the System III rotation period), which is 9.925 hr [14].

Rearranging (2.4) and putting it in component form gives us

$$\rho(-2\Omega v \sin\lambda + 2\Omega w \cos\lambda) = -\frac{1}{r\cos\lambda}\frac{\partial p}{\partial \phi} \tag{2.5}$$

$$\rho\, 2\Omega u \sin\lambda = -\frac{1}{r}\frac{\partial p}{\partial \lambda} \tag{2.6}$$

$$-\rho\, 2\Omega u \cos\lambda = -\frac{\partial p}{\partial r} - \rho g. \tag{2.7}$$

The ratio of the second and first terms on the left hand side of (2.5) is $\mathcal{O}(w/v) = \mathcal{O}(H/L)$, where H is the vertical scale height of the fluid, and thus the second term may be dropped under the assumption of small aspect ratio. Note that near the equator, this scaling does not work, but in that case, other terms we have already neglected must also be retained. We present an alternative thermal wind equation that is valid at the equator later in this section. In the absence of motion, i.e. in solid body rotation, the entire left hand side of each equation is zero, in which case the "static" pressure (p_s) and density (ρ_s) are functions of r only and are in hydrostatic balance:

$$\frac{\partial p_s}{\partial r} = -\rho_s g.$$

Thus we can write the pressure and density in terms of static and dynamic parts:

$$p = p_s(r) + \tilde{p}(r, \lambda, \phi)$$

$$\rho = \rho_s(r) + \tilde{\rho}(r, \lambda, \phi).$$

With this decomposition, the horizontal gradient $\nabla_\perp p$ reduces to $\nabla_\perp \tilde{p}$, and the ratio of the right and left hand sides for (2.6) and (2.5) scale as

$$\frac{\nabla_\perp \tilde{p}}{\rho 2(\mathbf{\Omega} \times \mathbf{v})_\perp} = \mathcal{O}\left(\frac{\tilde{p}/L}{\rho 2\Omega U}\right) \implies \tilde{p} = \mathcal{O}(\rho 2\Omega U L).$$

With this scaling for $\tilde{p}$, we turn to the ratio of the Coriolis acceleration and pressure gradient in (2.7),

$$\frac{\rho 2\Omega u \cos\lambda}{\partial\tilde{p}/\partial r} = \mathcal{O}\left(\frac{\rho 2\Omega U}{\rho 2\Omega U L/H}\right) = \mathcal{O}\left(\frac{H}{L}\right).$$

Since the aspect ratio is small, the Coriolis acceleration in the vertical direction is small and may be neglected, and the dynamic parts of the pressure and density are

also in hydrostatic balance. Our equations are now reduced to

$$\rho 2\Omega v \sin\lambda = \frac{1}{r\cos\lambda}\frac{\partial \tilde{p}}{\partial \phi} \tag{2.8}$$

$$\rho 2\Omega u \sin\lambda = -\frac{1}{r}\frac{\partial \tilde{p}}{\partial \lambda} \tag{2.9}$$

$$\frac{\partial p}{\partial r} = -\rho g. \tag{2.10}$$

The local normal component of the Coriolis acceleration is called the Coriolis parameter and is denoted $f = 2\Omega \sin\lambda$ for brevity. This set of equations is the geostrophic approximation to the full momentum equations, and the horizontal equations can be written succinctly as

$$\mathbf{v}_\perp = \frac{1}{f\rho}\hat{r} \times \nabla p.$$

Note that the geostrophic approximation does not say anything about the radial velocity aside from the assumption that it is small relative to the horizontal velocities.

To get the thermal wind equations, we make use of the chain rule relations:

$$(\nabla\rho)_p = \nabla\rho + \frac{\partial\rho}{\partial r}(\nabla r)_p$$

$$(\nabla p)_p = 0 = \nabla p + \frac{\partial p}{\partial r}(\nabla r)_p.$$

Taking the r derivative of the horizontal geostrophic equations,

$$\frac{\partial \mathbf{v}_\perp}{\partial r} = -\frac{1}{f\rho^2}\frac{\partial \rho}{\partial r}\hat{r} \times \boldsymbol{\nabla} p + \frac{1}{f\rho}\hat{r} \times \boldsymbol{\nabla}\left(\frac{\partial p}{\partial r}\right).$$

Using hydrostatic balance and the chain rule relations,

$$\begin{aligned}
\frac{\partial \mathbf{v}_\perp}{\partial r} &= +\frac{\rho g}{f\rho^2}\frac{\partial \rho}{\partial r}\hat{r} \times \left(\frac{\boldsymbol{\nabla} p}{\partial p/\partial r}\right) - \frac{g}{f\rho}\hat{r} \times \boldsymbol{\nabla}\rho \\
&= -\frac{g}{f\rho}\frac{\partial \rho}{\partial r}\hat{r} \times (\boldsymbol{\nabla} r)_p - \frac{g}{f\rho}\hat{r} \times \boldsymbol{\nabla} p \\
\frac{\partial \mathbf{v}_\perp}{\partial r} &= -\frac{g}{f\rho}\hat{r} \times (\boldsymbol{\nabla}\rho)_p.
\end{aligned}$$

These are the thermal wind equations in their most basic forms. For an ideal gas

$$\begin{aligned}
p = \rho R T \implies (\boldsymbol{\nabla} p)_p = 0 &= (\boldsymbol{\nabla}\rho)_p R T + (\boldsymbol{\nabla} T)_p R\rho \\
\implies \frac{(\boldsymbol{\nabla}\rho)_p}{\rho} &= -\frac{(\boldsymbol{\nabla} T)_p}{T},
\end{aligned}$$

and thus

$$\frac{\partial \mathbf{v}_\perp}{\partial r} = \frac{g}{fT}\hat{r} \times (\boldsymbol{\nabla} T)_p. \tag{2.11}$$

This is the form that inspired the title "thermal wind," and the profoundly simple physics that it implies is that in rapidly rotating flows with small aspect ratio, the latitudinal (longitudinal) change in temperature entirely determines the vertical

change in the longitudinal (latitudinal) wind speeds.

We have already stated that the scaling arguments presented do not apply at the equator, but in addition, we note that (2.11) is not valid at the equator because $1/f = 1/(2\Omega \sin\lambda) \to \infty$ as $\lambda \to 0$. However, a thermal wind equation for the longitudinal winds applicable near the equator has been derived in Marcus et. al [9], and we refer to this equation as the equatorial thermal wind equation (EQTWE),

$$\frac{\partial u}{\partial r} = -\frac{g}{f_0 r_0}\frac{1}{T}\left(\frac{\partial^2 T}{\partial \lambda^2}\right)_p,$$

where $f_0 = 2\Omega$ and r_0 is the radius of Jupiter at which the EQTWE is applied. The conditions under which the EQTWE holds are almost identical to those for the regular thermal wind equation; the important difference is that the modified Rossby number, which must be small for the EQTWE to hold, is $\overline{\mathrm{Ro}} = U/(f_0 L)$ and does not depend on $1/\sin\lambda$.

2.3 Deep or Shallow Jets?

How deep the zonal winds penetrate Jupiter is a topic of discussion to this day. Some claim that the jets penetrate deep into the atmosphere as a consequence of the Taylor–Proudman theorem [2]; others suggest that the jets are a consequence of shallow water dynamics and an inverse cascade of energy from small to large scales [8, 7].

2.3.1 Deep Jets and the Tangent Cylinder

The Taylor–Proudman theorem takes the geostrophic balance one step further by assuming that the flow is barotropic; that is, that the density can be written as a function of pressure only: $\rho = \rho(p)$. This theorem is simple enough to derive by working in a coordinate frame with $\mathbf{\Omega}$ along the $\hat{z}$ axis and taking the curl of (2.4),

$$\boldsymbol{\nabla} \times (2\mathbf{\Omega} \times \mathbf{v}) - \boldsymbol{\nabla} \times \left(\frac{1}{\rho}\boldsymbol{\nabla} p\right) + \boldsymbol{\nabla} \times (\boldsymbol{\nabla}\Phi) = 0. \tag{2.12}$$

The third term in (2.12) is identically zero, and the second term is

$$-\boldsymbol{\nabla} \times \left(\frac{1}{\rho}\boldsymbol{\nabla} p\right) = -\frac{1}{\rho}\boldsymbol{\nabla} \times \boldsymbol{\nabla} p + \frac{1}{\rho^2}\boldsymbol{\nabla}\rho \times \boldsymbol{\nabla} p. \tag{2.13}$$

Both of the terms in (2.13) are zero, the last one because $\boldsymbol{\nabla}\rho$ is in the same direction as $\boldsymbol{\nabla} p$ if $\rho = \rho(p)$. Finally, because $\mathbf{\Omega}$ is a constant, the velocity term is

$$\begin{aligned}\boldsymbol{\nabla} \times (2\mathbf{\Omega} \times \mathbf{v}) &= 2\mathbf{\Omega}(\boldsymbol{\nabla} \cdot \mathbf{v}) - 2(\mathbf{\Omega} \cdot \boldsymbol{\nabla})\mathbf{v} + 2(\mathbf{v} \cdot \boldsymbol{\nabla})\mathbf{\Omega} - 2\mathbf{v}(\boldsymbol{\nabla} \cdot \mathbf{\Omega}) \\ &= 2\mathbf{\Omega}(\boldsymbol{\nabla} \cdot \mathbf{v}) - 2(\mathbf{\Omega} \cdot \nabla)\mathbf{v} + 0 - 0.\end{aligned}$$

Thus geostrophic and barotropic motion leads to

$$2\Omega\frac{\partial v_x}{\partial z}\hat{x} + 2\Omega\frac{\partial v_y}{\partial z}\hat{y} + 2\Omega\left(\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y}\right)\hat{z} = 0$$

$$\implies \frac{\partial v_x}{\partial z} = 0, \quad \frac{\partial v_y}{\partial z} = 0, \quad \left(\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y}\right) = 0.$$

If the fluid is approximately incompressible, then

$$\nabla \cdot \mathbf{v} = 0 = \frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z}$$

$$\implies \frac{\partial v_z}{\partial z} = 0.$$

Thus the fluid cannot vary in the $z-$direction, i.e. in the $\mathbf{\Omega}-$direction, essentially reducing the flow to 2 dimensions. This has strange consequences, including that a solid object will project itself in the $z-$direction and the flow will move around the projection (called a Taylor column). This theorem was theoretically derived by Joseph Proudman [11], and Taylor columns were first demonstrated in a laboratory setting by Geoffrey Ingram Taylor [12].

If the Taylor–Proudman theorem holds on Jupiter, then the flow occurs in cylinders, and the zones we see are the intersection of the cylinders with the spherical surface. The metallic hydrogen core of Jupiter is theorized to create a Taylor column, called the tangent cylinder, inside of which zones cannot form, which would explain the lack of zones near the poles [2].

2.3.2 Shallow Jets and the Inverse Cascade

Another theory that attempts to describe the formation and size of the jets also approximates the flow as being 2 dimensional, this time with the radial direction suppressed. Under this model, the jets are shallow and driven by convection. Simulations model the deep convection by forcing the simulated domain with small scale vortices; under appropriate conditions, these vortices merge into larger and larger vortices, eventually forming large scale eddies or zonal flows. This "inverse cascade" of energy from small to large scales is eventually stopped by large scale dissipation, and the result is (again, under appropriate conditions) statistically steady zonal flows [8].

The question then is: what is the length scale at which the inverse cascade stops? This length scale would roughly determine the widths of the jets. One scaling [7] suggests that the jets are on the order of $2\pi L_R$, where L_R is the Rossby deformation radius and is a measure of the size at which rotational effects become as important as buoyancy. It is defined by

$$L_R = \frac{NH}{f}, \tag{2.14}$$

where H is the pressure scale height (approximately 30 km on Jupiter [14]) and N is the Brunt-Väisälä frequency, the frequency at which a displaced parcel of fluid

will oscillate in a stratified, statistically stable atmosphere, and is defined by

$$N^2 \equiv \frac{g}{\theta}\frac{\mathrm{d}\theta}{\mathrm{d}r}.$$

If N is taken to be a constant in the atmosphere, then $L_R \propto |\sin\lambda|^{-1}$. A least squares fit to zone width vs. $\frac{C}{|\sin\lambda|}$ qualitatively supports the relation between zone width and L_R, as can be seen in Figure (2.1), with $C = 6970$ km. If the zone widths are approximately $2\pi L_R$, then the fit parameter $C = 2\pi\frac{HN}{f_0} \implies \frac{f_0 C}{2\pi H} = 13$ mHz $= N$. Measurements of Jupiter's GRS at $|\lambda| = 23°$ estimate $L_R(23°) = 2000$ to 2200 km, which gives another calculation of $N = \frac{L_R(23°)\ f_0 \sin(23°)}{H} \approx 9.2$ to 10.1 mHz, which is the same order of magnitude as the least squares calculation.

These are somewhat rough scaling relations and calculations, but at the least they do not discredit the model of shallow jets generated by deep convection and an inverse cascade.

2.4 Quasiquadrennial Oscillation

We are concerned with and have explored theories about the extent of the jets below the cloud layer, but the content of this thesis has to do with what happens to the jets above the cloud layer, with the hope that our findings give insight into what happens below. The study of what happens above the cloud layer is not entirely without precedent; here, we discuss a relevant phenomenon called the quasiquadren-

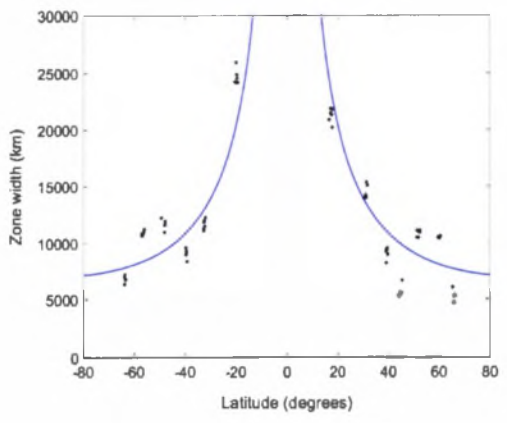

Figure 2.1: Least squares fit of zone width vs. latitude with the function $\frac{C}{|\sin\lambda|}$. The fit parameter $C = 6970$.

nial oscillation (QQO), which has been observed since 1980 [10, 5].

The QQO is the oscillation of the temperature at the equator in the stratosphere of Jupiter and is named as such because of its similarity to the Earth's semiannual and quasibiennial oscillations. Over a period of roughly 4–5 years, the temperature at the equator flips from being a local maxima to a local minima and back again; on Earth, this is observed to be associated with an oscillation in the wind shear from eastward to westward and back. On both Earth and Jupiter, the thermal wind equation is invoked as the relation between the temperature and wind shear oscillations, but the equation has not been numerically integrated to obtain the wind speeds in any previous studies of this phenomenon on Jupiter. It is important to note that the oscillations are confined to the equator, as can be seen in Figure (2.2).

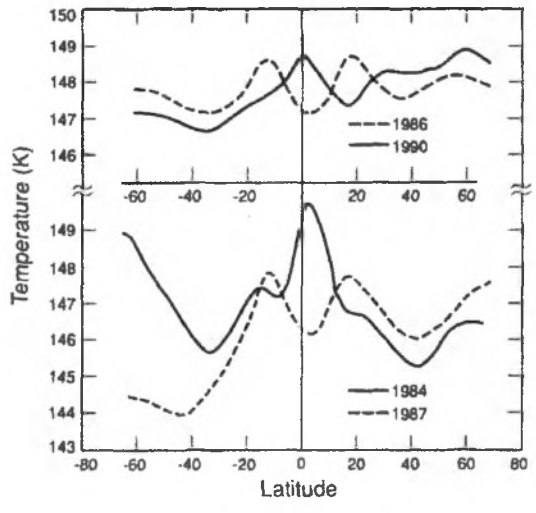

Figure 2.2: Temperature profiles from Figure 1 in Leovy, Friedson, & Orton [5]. These are attributed to pressures of 20 and 10 mbar for the upper and lower graphs, respectively.

Chapter 3

Methods and Results

The visible storms on Jupiter occur in what is often called the "cloud layer," a region generally accepted to have a pressure of $\sim$ 1 bar. The Galileo probe is the only instrument to directly measure properties of the atmosphere below the cloud layer and has measured wind speeds as a function of pressure down to $\sim$ 22 bars at its point of entry, approximately 7.3°N planetographic latitude [1]. This snapshot is helpful but limited, especially as the entry site was a "hot spot," a type of region with relatively few clouds whose dynamics are, as of yet, poorly understood. Though the hot spots are interesting in their own right, there is skepticism that measured properties in these regions can be extrapolated.

There have been plenty of cloud tracking measurements producing longitudinally averaged velocity vs. latitude profiles at the cloud level [6, 4, 13]. We use the average mean zonal flow data calculated from Hubble Space Telescope (HST) images from 1995–1998 by García-Melendo and Sánchez-Lavega [4] for our baseline velocities.

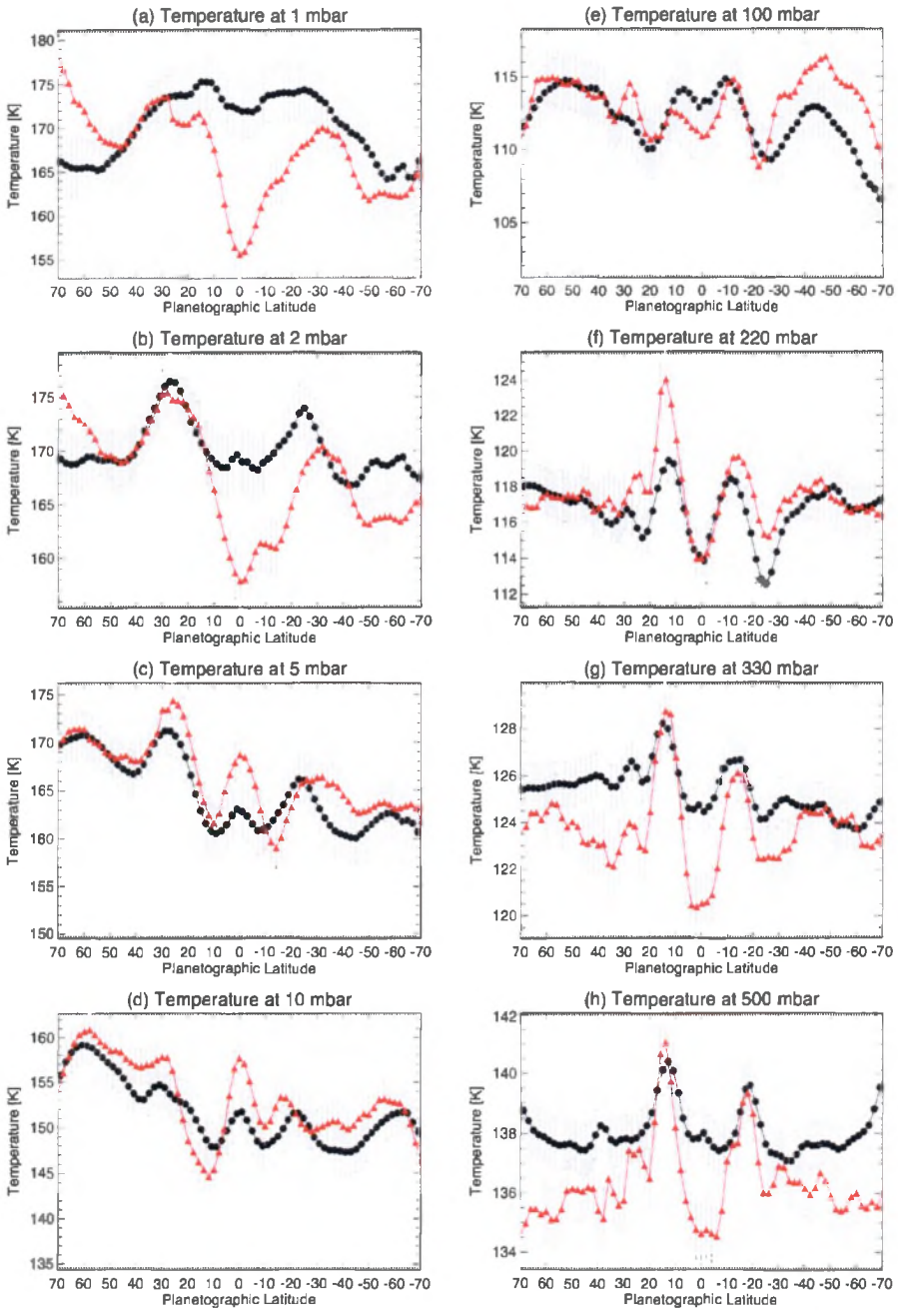

Figure 3.1: Temperature profiles from Figure 14 in Fletcher et. al [3]. Red triangles are from the CIRS instrument and black circles are from TEXES.

Leigh Fletcher et. al have produced longitudinally averaged temperature vs. latitude profiles of Jupiter at 10 pressure heights from 500 mbar to 1 mbar (shown in Figure (3.1)) by processing infrared data from the ground-based Texas Echelon Cross Echelle Spectrograph (TEXES) from 2014 and the satellite-based Cassini Composite Infrared Spectrometer (CIRS) from 2000 [3]. We use this data to integrate the thermal wind equation and the EQTWE.

3.1 Integration Technique

For Jupiter, we are only interested in the latitudinal velocities, and, for reasons discussed in Section (3.3), we work in $\ln p$ coordinates; continuing from (2.11), the thermal wind equation of interest is

$$\begin{aligned}
\frac{\partial u}{\partial \ln p}\frac{\partial \ln p}{\partial p}\frac{\partial p}{\partial r} &= -\frac{g}{fT}\frac{1}{r_0}\left(\frac{\partial T}{\partial \lambda}\right)_p \\
\frac{\partial u}{\partial \ln p}\left(\frac{1}{p}\right)(-\rho g) &= -\frac{g}{fTr_0}\left(\frac{\partial T}{\partial \lambda}\right)_p \\
\frac{\partial u}{\partial \ln p} &= \frac{p}{\rho T}\frac{1}{fr_0}\left(\frac{\partial T}{\partial \lambda}\right)_p \\
\frac{\partial u}{\partial \ln p} &= \frac{R}{f_0 r_0 \sin\lambda}\left(\frac{\partial T}{\partial \lambda}\right)_p,
\end{aligned} \tag{3.1}$$

where we have explicitly displayed the $1/\sin\lambda$ dependence. Similarly, the EQTWE in $\ln p$ coordinates is

$$\frac{\partial u}{\partial \ln p} = \frac{R}{f_0 r_0}\left(\frac{\partial^2 T}{\partial \lambda^2}\right)_p. \tag{3.2}$$

For reference, the constant $\frac{R}{f_0 r_0} \approx 0.15\ \mathrm{ms^{-1}K^{-1}}$.

To integrate the thermal wind equations, we first regularize the temperature data in latitude by applying a cubic spline interpolation to Fletcher's temperature data. The spline in MATLAB is described by "breaks" corresponding to the λ—coordinates of the input data and "coefs," which are the coefficients of the cubic polynomials on the intervals between the breaks. For instance, on the interval $[\lambda_1, \lambda_2]$ with coefficients $[a, b, c, d]$, the polynomial is

$$T(\lambda) = a(\lambda - \lambda_1)^3 + b(\lambda - \lambda_1)^2 + c(\lambda - \lambda_1) + d.$$

With the data splined, we could evaluate the spline at regular intervals and then use finite difference methods to numerically calculate the derivatives for use in the thermal wind equations; however, a simpler approach is to differentiate the spline first. The first and second derivatives of the spline are

$$\frac{\partial T}{\partial \lambda} = 0(\lambda - \lambda_1)^3 + 3a(\lambda - \lambda_1)^2 + 2b(\lambda - \lambda_1) + c$$
$$\frac{\partial^2 T}{\partial \lambda^2} = 0(\lambda - \lambda_1)^3 + 0(\lambda - \lambda_1)^2 + 6a(\lambda - \lambda_1) + 2b.$$

Thus we rearrange and scale the coefficients of the temperature spline in MATLAB, creating these first and second derivative splines, then evaluated these splines at regular intervals of $\Delta\lambda = 1°$.

With the derivatives of the temperature in hand, we use the thermal wind equation (3.1) and EQTWE to calculate the wind shear $\frac{\partial u}{\partial \ln p}$ at 10 pressure heights $p_1, \ldots, p_{10}$ from 500 mbar to 1 mbar and at regular intervals of $\Delta\lambda = 1°$, and from the HST data we have the value of the wind speed u at $p_0 = 950$ mbar,[1] which we spline and evaluate at the same λs. Our integration technique is one suggested by Marcus et. al [9], which is to fit parabolas in $x = \ln p$ by first solving

$$
\begin{aligned}
u(x_0) &= ax_0^2 + bx_0 + c \\
\left.\frac{\partial u}{\partial x}\right|_{x_1} &= 2ax_1 + b \\
\left.\frac{\partial u}{\partial x}\right|_{x_2} &= 2ax_2 + b
\end{aligned}
$$

and using the solution to evaluate $u(x_1)$, the velocity at $p_1 = 500$ mbar. After this

[1] 950 mbar is the pressure that Marcus et. al use for the cloud layer, which they infer by correlating the Galileo probe wind speeds vs. depth at 7.3°N with the HST wind speeds vs. latitude at cloud level. This correlation is warranted because the Galileo probe entry occurred near the same time as the HST observations.

first integration, we subsequently solve

$$\begin{aligned} u(x_i) &= ax_i^2 + bx_i + c \\ \left.\frac{\partial u}{\partial x}\right|_{x_i} &= 2ax_i + b \\ \left.\frac{\partial u}{\partial x}\right|_{x_{i+1}} &= 2ax_{i+1} + b \end{aligned}$$

for $i = 1, \ldots, 10$. As noted by Marcus et. al, fitting parabolas in $\ln p$ is equivalent to using the trapezoidal rule. Appendix A contains the MATLAB code used to perform these calculations.

When integrating the EQTWE we symmetrize the temperature data and skip 500 mbar as was done in Marcus et. al. They show that the velocity and thermodynamic variables at and near the equator are very nearly mirror-symmetric, that ignoring the anti-mirror-symmetric component is expected to introduce errors of less than 1%, and that symmetrizing the data can reduce numerical errors. Their reason for skipping 500 mbar is that it was considered to be too noisy and disproportionately affected the velocity profile, a problem we discovered as well. Marcus et. al determine the second derivative of the temperature by fitting parabolas of the form $T(\lambda) = T_0 + \frac{1}{2}T''\lambda^2$ to the symmetrized temperature data, but for consistency with our treatment of the regular thermal wind equation, we use the spline method with the addition of symmetrizing the splines.

3.2 Results and Discussion

The integrated velocity fields using the regular thermal wind equation can be seen in Figure (3.2), and the velocity fields calculated using the EQTWE are in Figure (3.4). When using the regular thermal wind equation and EQTWE, we cut out the regions $|\lambda| < 10°$ and $|\lambda| > 10°$, respectively, according to the approximate domains where they are applicable.

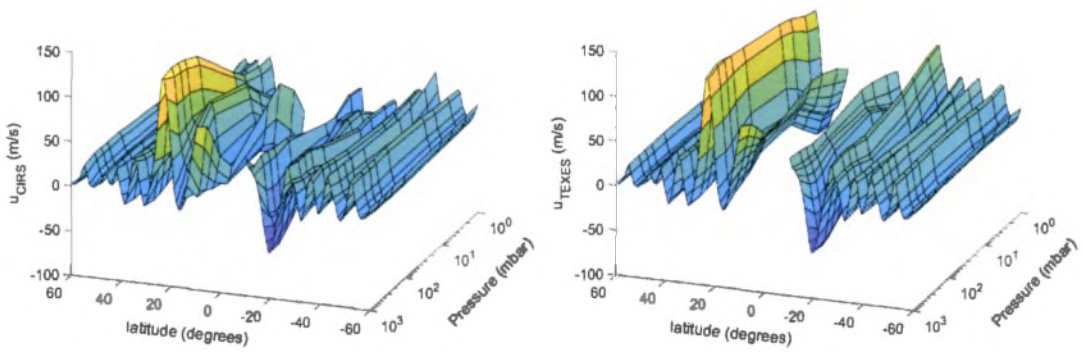

Figure 3.2: Velocity fields integrated using CIRS (left) and TEXES (right) temperature data and the regular thermal wind equation.

We note that in Figure (3.2) and (3.3) the velocities are very nearly constant far from the equator when both the height and data set are varied. This is because $\frac{R}{f_0 r_0}\left(\frac{\partial T}{\partial \lambda}\right)_p$ in the thermal wind equation is fairly small, and the remaining $1/\sin\lambda$ dependence is $\mathcal{O}(1)$ far from the equator but gets large near the equator. The temperature data in Figure (3.1) varies between TEXES and CIRS data sets, and

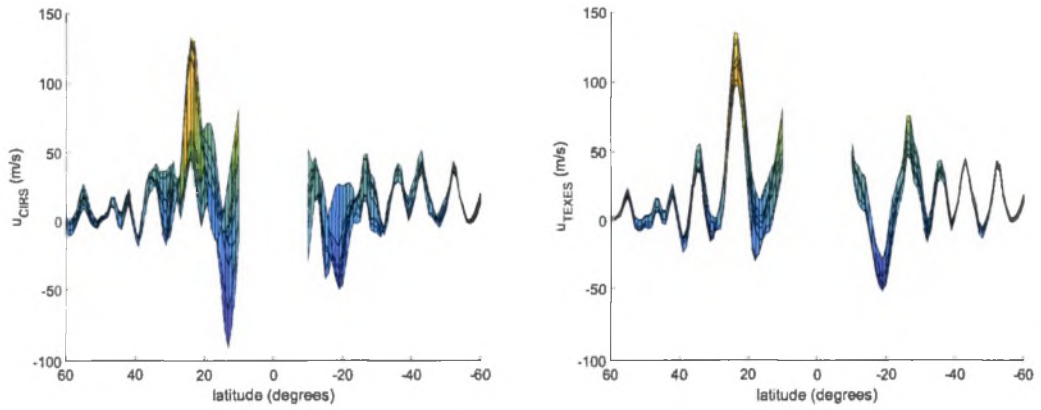

Figure 3.3: Same graph as Figure (3.2) viewed down the radial (pressure) axis. Note that the CIRS velocity field (right) varies more than the TEXES velocities (left) from $|\lambda| = 10°$ to $30°$, while both are fairly constant for $|\lambda| > 30°$

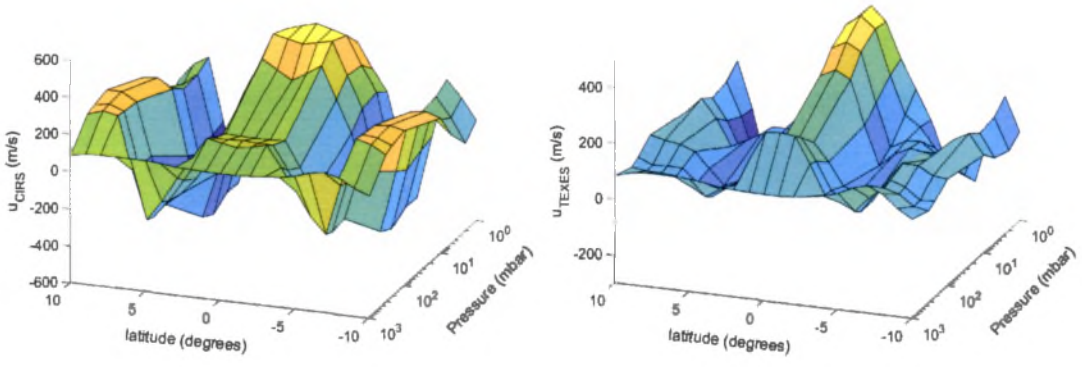

Figure 3.4: Velocity fields integrated using the CIRS (left) and TEXES (right) temperature data and the equatorial thermal wind equation.

thus presumably the temperatures vary in time, while the winds at large latitudes stay constant between the data sets, implying the wind is constant in time. This

is useful because it suggests that the wind speeds are insensitive to the changes in temperature at large latitudes, and thus the high latitude jets may be "deep," i.e. extend below the cloud layer as far as the geostrophic approximation holds.

Closer to the equator, the winds are more variable, but there does not appear to be a systematic trend; some zones get depressed, some get enhanced, and some appear to switch directions higher in the atmosphere. This is particularly pronounced in the case of the CIRS data, whereas the TEXES data does not show much variation in wind speeds at all, even up to 10° of the equator.

Integration of the EQTWE results in wildly varying velocity fields that reach up to ~ 500 ms^{-1}, in contrast to the integration performed by Marcus et. al, in which the velocities did not exceed 250 ms^{-1}, as seen in Figure (3.5). This discrepancy is due to our use of splines where Marcus et. al used parabolic least squares to determine the second derivative of the temperature, but the amount by which our results vary is troubling. Nonetheless, the variability between data sets in Figure (3.4) is not unexpected given the observations of the QQO.

We did not use temperature data from QQO observations, such as those in Figure (2.2), for integrating the EQTWE because the observations are at extremely low pressures (high altitudes), well above the cloud layer where we start our integration with known velocities. Fletcher's temperature data appears to be the only set with sufficient range, number, and quality of temperature profiles for use in numerical integration of the thermal wind equation with a baseline at 950 mbar.

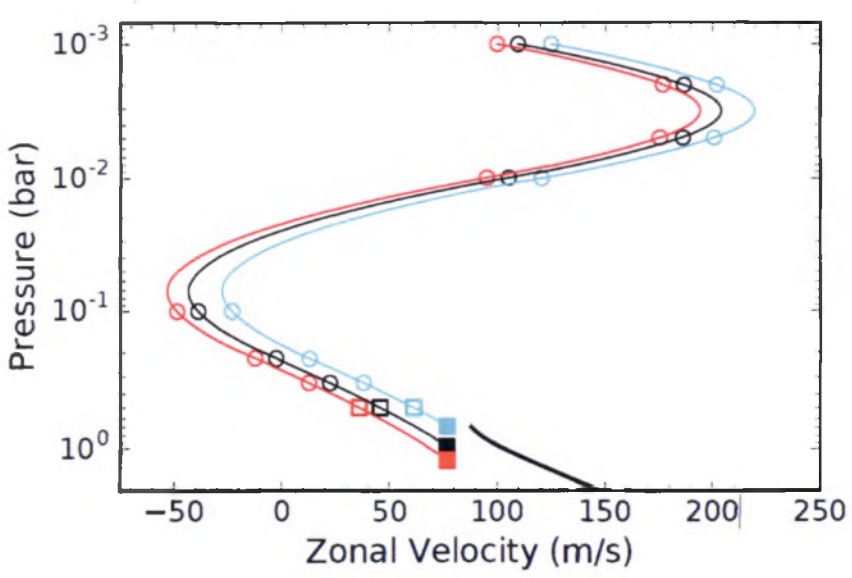

Figure 3.5: Velocity profile from Figure 7 in Marcus et. al [9]. The thin red, black, and cyan lines are the velocity fields integrated using values of $p = 1.2$ bar, 950 mbar, and 680 mbar, respectively, for the cloud layer. The thick black line that extends below 1 bar is the zonal velocity profile measured by the Galileo probe.

3.3 How Our Numerical Methods are Wrong, or: How I Learned to Stop Worrying and Love The Data

There are multiple ways to integrate the thermal wind equations, as well as many sources of error. In this section we briefly discuss why we use our method and consider the validity of our results.

3.3.1 Choice of Integration Coordinate

While the thermal wind equations written in terms of the derivative with respect to r vs. p vs. $\ln p$ are all mathematically equivalent, the numerical integration of the equations in these different forms is far from equivalent. First, we note that it would be hard to integrate in r because the derivative of the temperature is at surfaces of constant pressure, and the relation between radial extent and pressure is tied up in the hydrostatic balance equation $\frac{\partial p}{\partial r} = -\rho g$, another differential equation, the solutions of which would be approximate and thus would introduce errors. Between pressure and log pressure, the numerical integration is very different because the change in the numerical step size is nonlinear; equivalently, fitting a parabola in p is very different from fitting a parabola in $\ln p$.

We choose to integrate the equations in $\ln p$ because the scaling of the state variables in the atmosphere are logarithmic; specifically the scale height of a quantity q is given by

$$H_q = \left(\frac{1}{q_s}\frac{\partial q_s}{\partial r}\right)^{-1} = \left(\frac{\partial \ln q_s}{\partial r}\right)^{-1},$$

where q_s refers to the static component of the pressure, temperature, potential temperature, or density, and thus $\ln p$ is a "natural" coordinate choice.

3.3.2 Errors in Temperature Data

Until now, we have been sweeping perhaps the biggest concern under the rug: the error bars in Figure (3.1). The nature of the uncertainties in the temperature data has more of an effect than the size of the uncertainties, so let us consider the possibilities:

1. If the uncertainties are entirely systematic, then the temperatures themselves will be highly uncertain, but the *changes* in temperature, and thus the derivative and second derivative, will be precise.

2. If the uncertainties are standard deviations of the mean when longitudinally averaging the temperature (but each individual measurement is precise), then the uncertainties may not be of concern because the thermal wind equations describe the statistically steady flow, i.e. the mean, and are not concerned with the deviations from that. In other words the temperatures may vary due to the nonlinearity of the flow, but our analysis is concerned with the steady state around which the nonlinearities fluctuate. That said, if the standard deviations are very large, then the thermal wind approach may not be valid, as it assumes some level of statistical stability.

3. If the uncertainties in the temperature are entirely random for each measurement, i.e. "equipment" random, then the uncertainties in the derivative and second derivative of the temperature will only be worse.

It may be that (2) and (3) are functionally equivalent, but we entertain the difference for completeness.

In their caption for for the temperature profiles, Fletcher et. al state that the error bars "represent retrieval uncertainties, comprising random measurement error, smoothing error and degeneracies. Systematic calibration uncertainties are not included in these error bars." [3] The best case scenario of purely systematic uncertainties is thus not the case. The amounts that randomness, smoothing, and degeneracies contribute to the error bars is not discussed directly, although regarding randomness, Fletcher et. al note that they calculate the standard deviation in the measurements of the sky behind Jupiter as well as the standard deviation of the longitudinally averaged temperatures and use the more conservative of the two for their estimate of the random uncertainty for each spectral band. They also note that the standard deviation is smallest where the atmosphere is most transparent, and thus the 500 mbar measurements are the most error prone.

Assuming the worst about the uncertainties in the temperature, our integration of the regular thermal wind equation is suspect and our integration of the EQTWE is swimming in error. Numerically, second derivatives are always more uncertain than first derivatives and requires very high quality data to be trustworthy. We are not convinced that the temperature data is of high enough quality to quantitatively trust our results, but the method of parabolic least squares fitting of the symmetrized temperature data used by Marcus et. al in [9] may reduce random errors sufficiently

enough to trust their results. However, their method ignores real fluctuations in latitude and produces a single velocity vs. pressure curve at the equator by fitting data from $\pm 10^{\circ}$ around the equator.

Chapter 4

Conclusion and Future Work

In this thesis we numerically integrated the regular and equatorial thermal wind equations to obtain mean zonal velocity fields above the cloud layer near and far from the equator. We found that far from the equator the zonal velocity at a fixed latitude are relatively constant in height and time, suggesting that the velocity at the cloud layer extends radially as far as the geostrophic approximation holds. Near the equator we found that the temperature and velocity fields fluctuate in height and time, as has been observed in the case of the temperature viz. the QQO, although we believe our results are full of noise and thus should be treated qualitatively due to the presence of the error-exacerbating second derivative of the temperature in the EQTWE.

Our results suggest that simulations of the atmosphere at latitudes $\gtrsim 30°$ should expect relatively constant zonal wind speeds to extend above the cloud layer and possibly below it, which can be used as a rubric for simulations that explore various

types of convective forcing and radiative damping. Concerning the atmosphere near the equator, we would like to perform or see others perform high resolution equatorial simulations that extend into the stratosphere to explore the nature of the quasiquadrennial oscillation. Given that the winds at the cloud layer appear to be relatively stable, simulations exploring the QQO could use the cloud layer winds as boundary conditions.

What we would like more than simulations is more data below the cloud layer from in situ probes. To determine whether the jets are deep or shallow, we suggest a probe enter an average region of the atmosphere at mid-latitudes, as opposed to a hot spot near the equator. However, we recognize that this type of mission is extremely expensive. A more realistic endeavor is to perform more infrared measurements that probe the temperature as close to the cloud layer as possible, ideally at regular intervals, to decrease statistical errors and to better monitor the QQO profile from cloud layer to stratosphere.

Bibliography

[1] David H Atkinson, James B Pollack, and Alvin Seiff, *The galileo probe doppler wind experiment: Measurement of the deep zonal winds on jupiter*, Journal of Geophysical Research: Planets **103** (1998), no. E10, 22911–22928.

[2] FH Busse, *A simple model of convection in the jovian atmosphere*, Icarus **29** (1976), no. 2, 255–260.

[3] Leigh N Fletcher, TK Greathouse, GS Orton, JA Sinclair, RS Giles, PGJ Irwin, and T Encrenaz, *Mid-infrared mapping of jupiters temperatures, aerosol opacity and chemical distributions with irtf/texes*, Icarus **278** (2016), 128–161.

[4] E García-Melendo and A Sánchez-Lavega, *A study of the stability of jovian zonal winds from hst images: 1995–2000*, Icarus **152** (2001), no. 2, 316–330.

[5] Conway B Leovy, A James Friedson, and Glenn S Orton, *The quasiquadrennial oscillation of jupiter's equatorial stratosphere*, Nature **354** (1991), no. 6352, 380.

[6] Sanjay S Limaye, *Jupiter: New estimates of the mean zonal flow at the cloud level*, Icarus **65** (1986), no. 2-3, 335–352.

[7] Philip S. Marcus, Private Communication, 2018.

[8] Philip S. Marcus, T. Kundu, and Changhoon Lee, *Vortex dynamics and zonal flows*, Physics of Plasmas **7** (2000), no. 5, 1630–1640.

[9] Philip S Marcus, Joshua Tollefson, Michael H Wong, and Imke de Pater, *An equatorial thermal wind equation: applications to jupiter*, Icarus (2018).

[10] Glenn S Orton, A James Friedson, Kevin H Baines, Terry Z Martin, Robert A West, John Caldwell, Heidi B Hammel, Jay T Bergstralh, Michael E Malcom, William F Golisch, et al., *Thermal maps of jupiter: Spatial organization and time dependence of stratospheric temperatures, 1980 to 1990*, Science **252** (1991), no. 5005, 537–542.

[11] Joseph Proudman, *On the motion of solids in a liquid possessing vorticity*, Proc. R. Soc. Lond. A **92** (1916), no. 642, 408–424.

[12] Geoffrey Ingram Taylor, *The motion of a sphere in a rotating liquid*, Proc. R. Soc. Lond. A **102** (1922), no. 715, 180–189.

[13] Joshua Tollefson, Michael H Wong, Imke de Pater, Amy A Simon, Glenn S Orton, John H Rogers, Sushil K Atreya, Richard G Cosentino, William Januszewski, Raúl Morales-Juberías, et al., *Changes in jupiters zonal wind profile preceding and during the juno mission*, Icarus **296** (2017), 163–178.

[14] David R. Williams, *Jupiter fact sheet*, `https://nssdc.gsfc.nasa.gov/planetary/factsheet/jupiterfact.html`, June 2018, Accessed: 2018-12-10.

Appendix A: MATLAB code

The MATLAB script `makeTempSplines.m`:

```
% Temperature data in arrays of shape (N,2), the first column contains the
% latitudes and the second column contains the temperatures.

% spline the data
T1 = spline(TEXAS_1mbar(:,1)*pi/180, TEXAS_1mbar(:,2));
T2 = spline(TEXAS_2mbar(:,1)*pi/180, TEXAS_2mbar(:,2));
T5 = spline(TEXAS_5mbar(:,1)*pi/180, TEXAS_5mbar(:,2));
T10 = spline(TEXAS_10mbar(:,1)*pi/180, TEXAS_10mbar(:,2));
T100 = spline(TEXAS_100mbar(:,1)*pi/180, TEXAS_100mbar(:,2));
T220 = spline(TEXAS_220mbar(:,1)*pi/180, TEXAS_220mbar(:,2));
T330 = spline(TEXAS_330mbar(:,1)*pi/180, TEXAS_330mbar(:,2));
T500 = spline(TEXAS_500mbar(:,1)*pi/180, TEXAS_500mbar(:,2));

% make copies of splines
dT1=T1;
dT2=T2;
dT5=T5;
dT10=T10;
dT100=T100;
dT220=T220;
dT330=T330;
dT500=T500;
% rearrange coefficients to make first derivative splines
```

```
dT1.coefs=dT1.coefs(:,[4 1 2 3]).*[0 3 2 1];
dT2.coefs=dT2.coefs(:,[4 1 2 3]).*[0 3 2 1];
dT5.coefs=dT5.coefs(:,[4 1 2 3]).*[0 3 2 1];
dT10.coefs=dT10.coefs(:,[4 1 2 3]).*[0 3 2 1];
dT100.coefs=dT100.coefs(:,[4 1 2 3]).*[0 3 2 1];
dT220.coefs=dT220.coefs(:,[4 1 2 3]).*[0 3 2 1];
dT330.coefs=dT330.coefs(:,[4 1 2 3]).*[0 3 2 1];
dT500.coefs=dT500.coefs(:,[4 1 2 3]).*[0 3 2 1];

% make copies of splines
ddT1=T1;
ddT2=T2;
ddT5=T5;
ddT10=T10;
ddT100=T100;
ddT220=T220;
ddT330=T330;
ddT500=T500;
% rearrange coefficients to make second derivative splines
ddT1.coefs=ddT1.coefs(:,[4 3 1 2]).*[0 0 6 2];
ddT2.coefs=ddT2.coefs(:,[4 3 1 2]).*[0 0 6 2];
ddT5.coefs=ddT5.coefs(:,[4 3 1 2]).*[0 0 6 2];
ddT10.coefs=ddT10.coefs(:,[4 3 1 2]).*[0 0 6 2];
ddT100.coefs=ddT100.coefs(:,[4 3 1 2]).*[0 0 6 2];
ddT220.coefs=ddT220.coefs(:,[4 3 1 2]).*[0 0 6 2];
ddT330.coefs=ddT330.coefs(:,[4 3 1 2]).*[0 0 6 2];
ddT500.coefs=ddT500.coefs(:,[4 3 1 2]).*[0 0 6 2];
```

The MATLAB script `loopin.m`:

```
% The pressure domain, ln(pressure in Pa)
pres=log(100*[950 500 330 220 100 10 5 2 1]);
% Let's choose a theta domain... -60 to 60 degrees latitude
xx=-60:1:60;
xx=xx*pi/180;
```

```
% idealGasR: the idea gas constant R for jupiter, Pa m^3 / (kg K) (i.e. SI)
idealGasR = 100000 / (165 * 0.16);
% r0: Jupiter's equatorial radius at 1 bar
r0 = 71492 * 10^3;
% f0: 2 Omega_0, where Omega_0 is the rotational rate of Jupiter
f0 = 2 * (2 * pi / (9.925 * 60 * 60));

myConst = idealGasR./r0./f0;

% Initialize the array to hold the velocities
Vels=zeros(length(xx),length(pres));
% Velocity at ~950 mbar from Lavega (HST)
% Note the conversion from degrees to radians
cloudvels=load('HST_vels.txt');
cloudvels(:,1)=cloudvels(:,1)*pi/180;
Vels(:,1)=spline(cloudvels(:,1),cloudvels(:,2),xx);

load('T&dT.mat')
% note that we don't have deriv @ 950, so this list seems "shifted"
% compared to other lists; Vels(:,2) and dTs(1) corresond with velocities
% and derivatives at 500 mbar.
dTs=[dT500 dT330 dT220 dT100 dT10 dT5 dT2 dT1];

% Coeficients of the parabolic fits
aa=zeros(length(xx),length(pres)-1);
bb=zeros(length(xx),length(pres)-1);
cc=zeros(length(xx),length(pres)-1);

% Initial fit matches v at 950, deriv at 500, deriv at 330
myMat=[[pres(1).^2, pres(1), 1]; [2*pres(2), 1, 0]; [2*pres(3), 1, 0]];
deriv1=myConst./sin(xx).*ppval(dTs(1),xx);
deriv2=myConst./sin(xx).*ppval(dTs(2),xx);
for ii=1:length(xx)
temp = myMat \ [Vels(ii,1); deriv1(ii); deriv2(ii)];
```

```
aa(ii,1)=temp(1);
bb(ii,1)=temp(2);
cc(ii,1)=temp(3);
end
Vels(:,2) = aa(:,1).*pres(2).^2 + bb(:,1).*pres(2) + cc(:,1);

% Next one matches value at 500, deriv at 500, deriv at 330

for jj=2:8
myMat=[[pres(jj).^2, pres(jj), 1]; [2*pres(jj), 1, 0]; [2*pres(jj+1), 1, 0]];
% dTs at jj-1 because it's "shifted" to the left one index (i.e.
% because we don't have derivatives at 950)
deriv1=myConst./sin(xx).*ppval(dTs(jj-1),xx);
deriv2=myConst./sin(xx).*ppval(dTs(jj),xx);

for ii=1:length(xx)
temp = myMat \ [Vels(ii,jj); deriv1(ii); deriv2(ii)];
aa(ii,jj)=temp(1);
bb(ii,jj)=temp(2);
cc(ii,jj)=temp(3);
end

Vels(:,jj+1) = aa(:,jj).*pres(jj+1).^2 + bb(:,jj).*pres(jj+1) + cc(:,jj);
end

[Z,X]=meshgrid(pres,180/pi*xx);
CutVels=Vels;
CutVels(abs(X)<10)=CutVels(abs(X)<10)*NaN;
```

The MATLAB script `loopinEQTWE.m`:

```
% Using deriv at 500, or skipping that?
using500=false;

load('T&dT&ddT.mat')
```

```
% note that we don't have deriv @ 950, so this list seems "shifted"
% compared to other lists - EQVels(:,2) and ddTs(1) corresond with velocities
% and derivatives at 500 mbar.

% Set up the pressure domain, ln(pressure in Pa), and the derivs of T
if using500
pres=log(100*[950 500 330 220 100 10 5 2 1]);
ddTs=[ddT500 ddT330 ddT220 ddT100 ddT10 ddT5 ddT2 ddT1];
else
pres=log(100*[950 330 220 100 10 5 2 1]);
ddTs=[ddT330 ddT220 ddT100 ddT10 ddT5 ddT2 ddT1];
end

% Let's choose a theta domain... -60 to 60 degrees latitude
xx=-60:1:60;
xx=xx*pi/180;

% idealGasR - the idea gas constant R for jupiter, Pa m^3 / (kg K) (i.e. SI)
idealGasR = 100000 / (165 * 0.16);
% r0 - Jupiter's equatorial radius at 1 bar
r0 = 71492 * 10^3;
% f0 - 2 Omega_0, where Omega_0 is the rotational rate of Jupiter
f0 = 2 * (2 * pi / (9.925 * 60 * 60));

myConst = idealGasR./r0./f0;

% Initialize the array to hold the velocities
EQVels=zeros(length(xx),length(pres));
% Velocity at ~950 mbar from Lavega (HST)
% Note the conversion from degrees to radians
cloudvels=load('HST_vels.txt');
cloudvels(:,1)=cloudvels(:,1)*pi/180;
EQVels(:,1)=spline(cloudvels(:,1),cloudvels(:,2),xx);

% Coeficients of the parabo lic fits
```

```
aa=zeros(length(xx),length(pres)-1);
bb=zeros(length(xx),length(pres)-1);
cc=zeros(length(xx),length(pres)-1);

% Initial fit matches v at 950, deriv at 500, deriv at 330.
% (Unless we're skipping 500, naturally, then it does deriv at 330 and
% deriv at 220)
myMat=[[pres(1).^2, pres(1), 1]; [2*pres(2), 1, 0]; [2*pres(3), 1, 0]];
% derivatives are symmetrized at this step
deriv1=myConst.*(ppval(ddTs(1),xx)./2 + ppval(ddTs(1),-xx)./2 );
deriv2=myConst.*(ppval(ddTs(2),xx)./2 + ppval(ddTs(2),-xx)./2);
% deriv1=myConst.*ppval(ddTs(1),xx);
% deriv2=myConst.*ppval(ddTs(2),xx);
for ii=1:length(xx)
temp = myMat \ [EQVels(ii,1); deriv1(ii); deriv2(ii)];
aa(ii,1)=temp(1);
bb(ii,1)=temp(2);
cc(ii,1)=temp(3);
end
EQVels(:,2) = aa(:,1).*pres(2).^2 + bb(:,1).*pres(2) + cc(:,1);

% Next one matches value at 500, deriv at 500, deriv at 330
% (Unless we're skipping 500, naturally... You get the idea)

for jj=2:length(pres)-1
myMat=[[pres(jj).^2, pres(jj), 1]; [2*pres(jj), 1, 0]; [2*pres(jj+1), 1, 0]];
% ddTs at jj-1 because it's "shifted" to the left one index (i.e.
% because we don't have derivatives at 950)
% derivatives are symmetrized at this step
deriv1=myConst.*(ppval(ddTs(jj-1),xx)./2 + ppval(ddTs(jj-1),-xx)./2);
deriv2=myConst.*(ppval(ddTs(jj),xx)./2 + ppval(ddTs(jj),-xx)./2);
%     deriv1=myConst.*ppval(ddTs(jj-1),xx);
%     deriv2=myConst.*ppval(ddTs(jj),xx);

for ii=1:length(xx)
```

```
temp = myMat \ [EQVels(ii,jj); deriv1(ii); deriv2(ii)];
aa(ii,jj)=temp(1);
bb(ii,jj)=temp(2);
cc(ii,jj)=temp(3);
end

EQVels(:,jj+1) = aa(:,jj).*pres(jj+1).^2 + bb(:,jj).*pres(jj+1) + cc(:,jj);
end

[Z,X]=meshgrid(pres,180/pi*xx);
CutEQVels=EQVels;
CutEQVels(abs(X)>10)=CutEQVels(abs(X)>10)*NaN;
```