



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

**Escola Superior d'Enginyeries Industrial,
Aeroespacial i Audiovisual de Terrassa**

Implementation of a Shallow Water model for atmospheric applications in Python.

A thesis submitted in partial fulfillment of the requirements for the degree
of BSc. in Aerospace Engineering Technologies

Author

Sergio Coll Ibárs

Director

Enrique García Melendo

Co-director

Manel Soria Guerrero

ESCUELA SUPERIOR DE INGENIERÍAS INDUSTRIAL, AEREOESPACIAL I AUDIOVISUAL DE
TERRASSA (ESEIAAT)

Terrassa, June 22, 2021

Abstract

English

The simulation of large-scale atmospheric phenomena produced on the giant planets of the solar system is the subject of this work. For this purpose, a two-dimensional mathematical model is used which adapts the equations of *Navier Stokes*, making the necessary considerations for the atmospheres of the planets of the solar system, it is the "Shallow Water" model.

In addition to the mathematical development and numerical implementation of the model, a series of experiments are developed that corroborate "step by step" the theory explained, facilitating its understanding. All the results shown in this project are obtained using software developed in the programming language *Python*, one of the most widespread languages nowadays and which facilitates the understanding of the code thanks to the simplicity of its syntax and wide range of libraries.

The final result explains the procedure to computationally replicate one of the most famous atmospheric phenomena of the Solar System, the Great Red Spot of Jupiter (*GRS*), chosen because of the great availability of data and its great importance as a meteorological phenomenon within the solar system. This method is ideal for learning fluid dynamic computation.

Castellano

La simulación de fenómenos atmosféricos a gran escala producidos en los planetas gigantes del sistema solar son el tema de estudio de este trabajo. Para ello se hace uso de un modelo matemático bidimensional que adapta las ecuaciones de *Navier Stokes* realizando las consideraciones necesarias para las atmósferas de los planetas del sistema solar, se trata del modelo *Shallow Water*.

Además del desarrollo matemático e implementación numérica del modelo se desarrollan una serie de experimentos que corroboran "paso a paso" la teoría explicada, facilitando su comprensión. Todos los resultados mostrados en este proyecto se obtienen mediante un software desarrollado en el lenguaje de programación *Python*, uno de los lenguajes más extendidos hoy en día y que facilita la comprensión del código gracias a la sencillez de su sintaxis y amplia gama de librerías.

Como resultado final se explica el procedimiento para replicar computacionalmente uno de los fenómenos atmosféricos más famosos del sistema Solar, la Gran Mancha Roja de Júpiter (GRS) siendo esta elegida por la gran disponibilidad de datos y por su gran importancia como fenómeno metereológico dentro del sistema solar. Se trata de un método idóneo para el aprendizaje de la computación dinámica de fluidos.

I declare that,

the work in this Degree Thesis is completely my own work,

no part of this Degree Thesis is taken from other people's work without giving them credit,

all references have been clearly cited,

I'm authorised to make use of the research group related information I'm providing in this document.

I understand that an infringement of this declaration leaves me subject to the foreseen disciplinary actions by *The Universitat Politècnica de Catalunya - BarcelonaTECH*

Sergio Coll Ibáñez

June 22, 2021

Study: Implementation of a Shallow Water model for atmospheric applications in Python

Contents

1	Introduction	9
2	Initial approach	12
3	Fluid motion equations	13
4	Rotating coordinate frame	16
4.1	Rossby Number	17
5	Vorticity	18
5.1	Potential Vorticity	19
6	The Shallow Water model	21
6.1	Shallow Water equations	23
6.2	Potential vorticity applied to Shallow Water model	29
7	Shallow Water Implementation	30
7.1	Arakawa's grids	31
7.2	Finite Difference Method. FDM	36
7.3	Advection Terms computation	38
7.4	The Courant Number	48
7.5	Flux Limiter function	49
7.6	Time Integration	53
7.7	Coriolis Integration	55
8	MMS Validation method	58
9	Numerical Experiments	66
9.1	Gravity Wave Simulation	67
9.2	Cartesian vortex simulation	75
9.2.1	Geostrophic balance	75
9.2.2	Introduction zonal winds to the SW equations	76
9.2.3	Perturbation introduction	78

9.2.4	Simulation results	80
9.3	Elliptical vortex simulation	82
9.3.1	Derivation of ellipsoidal coordinates	82
9.3.2	Perturbation introduction	89
9.3.3	Simulation results	90
9.4	Great Red Spot	92
9.4.1	Perturbation introduction	94
9.4.2	Simulation results	96
9.5	Saturn storms	100
9.5.1	Simulation results	101
10	Conclusions and Future developments	102
11	Budget	103
12	Appendix	107
12.1	Jupiter Zonal Winds	108
12.2	Saturn Zonal Winds	124

List of Figures

1	Element of mass between two surfaces of constant λ	20
2	Ratio between horizontal plane and vertical ratio	21
3	Shallow Water scheme	23
4	Finite differences. Centred grid	32
5	Finite differences. Staggered grid	32
6	Saw-tooth function in a centred grid. Solid point represent surface perturbation η . Saw-tooth solid line represents the continuous η	33
7	Saw-tooth function in a staggered grid	33
8	C-grid variables distribution	34
9	Arakawa's grids. From Geophysical fluid dynamics (2010) [6]	35
10	Arakawa C-grid	35
11	Control Volume centered at u velocity	40
12	Comparison of Upwind schemes	50
13	Flux Limiter function $\Psi(r)$	51
14	Shallow Water Equations. Integration scheme used for each term	53
15	v^n velocity interpolation.	57
16	\tilde{P}_{1u} and \tilde{P}_{1v} advection error. Constant Δt	61
17	\tilde{P}_{2u} and \tilde{P}_{2v} advection error. Constant Δt	62
18	\tilde{P}_h continuity equation term validation. Constant Δt	63
19	\tilde{P}_p pressure equation term validation. Constant Δt	64
20	Adam Bashforth validation for the Shallow Water model. Constant Δt	65
21	Double Periodic Conditions	68
23	Wave simulations at different time steps. Cartesian domain (200 x 200 m) and Periodic Boundary Conditions (PBC)	70
25	Full Slip Conditions	73
27	Wave simulations at different time steps. FSC Boundary Conditions. Cartesian domain (200 x 200 m)	74
28	Surface elevation at different time steps. Cartesian domain. $\Delta t = 30\text{s}$. $t_f = 20$ days	78

29	Cartesian vortex parameters over the domain (2000 x 2000) km. Time simulation: 20 days	80
30	vortex at different time steps. Domain (2000 x 2000)km. Time simulation: 20 days	81
31	Elliptical coordinates	85
32	Elliptical coordinates. XZ plane projection	85
33	Definitions of latitude for an oblate planet. RE and RP are the equatorial and polar radii respectively. φ_c and φ_g the planetocentric and planetographic latitudes. The red dot is a surface point.	86
34	Canal Boundary conditions	88
35	vortex at different time steps. Elliptical domain. Time simulation: 20 days	90
36	Potential vorticity over time. Elliptical domain	90
37	Vortex colour at different latitudes. Image created by <i>Dr. Enrique Garcia Melendo</i>	91
38	Jupiter (left) and Saturn (right) zonal wind system superimposed on the visible cloud deck. From Kaspi et al. 2019, https://arxiv.org/pdf/1908.09613.pdf	92
39	Comparison of the four zonal wind systems on the giant planets of the Solar System. Sukoriansky et al. (2002).	93
41	Great Red Spot simulation using geostrophic balance. Jupiter planet parameters. $N_x = 100$, $N_y = 100$	96
42	GRS simulation using geostrophic balance. u and v velocity obtained with the Geostrophic Equilibrium (GE) and Shallow Water (SW) model	97
43	GRS u and v velocity value obtained by different missions. (Velocity and Vorticity Measurements of Jupiter's Great Red Spot Using Automated Cloud Feature Tracking [5])	97
44	GRS image from: https://hubblesite.org compared with SW simulation. $T = 55$ days, $\Delta t = 30s$, $N_x = 100$, $N_y = 100$	99
45	Saturn storm simulation. $\Delta t = 30s$, $D = 500m$	101
46	Real image of the Saturn storm in 2010. At this time the storm was still growing	101

List of Tables

1	Flux limiter Schemes. Arranged by Scheme's name, expression and accuracy	52
2	Wave simulation (PBC). Simulation parameters	69
3	Wave simulation (PBC). Surface elevation at different points and time steps	70
4	Wave simulation results. FSC Boundary Conditions	74
5	Initial parameters used to introduce surface perturbation into the Cartesian domain	79
6	Initial parameters used to introduce perturbation into the Elliptical domain	89
7	Elliptical domain, physical simulation parameters	89
8	Geometrical and physical variables used to implement GRS perturbation	95
9	Saturn storm simulation parameters.	100
10	Project budget	103
11	Jupiter Zonal Winds data	109
12	Continue I. Jupiter Zonal Winds data	110
13	Continue II. Jupiter Zonal Winds data	111
14	Continue III. Jupiter Zonal Wind data	112
15	Continue IV. Jupiter Zonal Wind data	113
16	Continue V. Jupiter Zonal Wind data	114
17	Continue VI. Jupiter Zonal Wind data	115
18	Continue VII. Jupiter Zonal Wind data	116
19	Continue VIII. Jupiter Zonal Wind data	117
20	Continue IX. Jupiter Zonal Wind data	118
21	Continue X. Jupiter Zonal Wind data	119
22	Continue XI. Jupiter Zonal Wind data	120
23	Continue XII. Jupiter Zonal Wind data	121
24	Continue XIII. Jupiter Zonal Wind	122
25	Continue XIV. Jupiter Zonal Wind data	123
26	Saturn Zonal Wind data	125
27	Continue I. Saturn Zonal Wind data	126
28	Continue II. Saturn Zonal Wind data	127
29	Continue III. Saturn Zonal Wind data	128

1 Introduction

Interest in the space industry has boomed in recent years, with a substantial increase in the number of missions dedicated to exploration and data collection on other planetary bodies. However, so far the availability of data on the atmospheric phenomena produced on other planets is scarce, and requires the development of software to understand their behaviour.

The model developed in this study allows us to understand the physics behind some phenomena by replicating different storms and vortex produced on planets such as Jupiter or Saturn.

This study is part of the APPLY European master's project that trains aspiring field and simulation engineers, R&D specialists and future industry managers in India, Thailand and Malaysia in the field of applied Computational Fluid Dynamics [4]. For this purpose, this Asian universities collaborates with several European universities, including the Polytechnic University of Catalonia (UPC). For this reason, this study focuses on the development of the theoretical framework and implementation of the two-dimensional Shallow Water model for educational purposes.

The Shallow Water model has been used for decades to study different atmospheric phenomena. Even today, where we can find more sophisticated models, it is still used as validation software, since it allows a certain simplicity in its implementation and a fast computational speed, which makes this study ideal to introduce the reader to the direct applications of computational fluid dynamics.

This thesis explains the implementation of the Shallow Water equations and how they should be applied computationally, detailing the numerical schemes and key points of the code: simulation parameters, algorithms, validations, etc, for students of computational geophysical fluid dynamics. The goal is to allow students to implement their own fully functional Shallow Water model in a open source language such as Python. In addition, a series of experiments are presented to facilitate the reader's understanding of the theoretical contents and to allow him to validate his own results.



Finally, instructions are added to recreate Jupiter's Great Red Spot (GRS), the final objective that the reader should reach by using the theory in this document. All the results are complemented with images of the simulations obtained, thus allowing visual as well as numerical comparisons; the data extracted from these simulations are compared with analytical data provided by space probes along different missions to test the validity of the model developed.

Shallow Water model



2 Initial approach

Geophysical fluids dynamics is the study of the fluid's movement caused by natural phenomena like winds, ocean currents at a large or planetary scale. The main objective of this study is to focus on the atmosphere of giant planets in the Solar system. To do so, the *Shallow Water model* is used to study large-scale motions.

The principles derived below are used to understand natural phenomena. Those principles are based on theoretical concepts and basic equations of physics. Being as the model is built using those principles, its validity is based on the comparison with experimental results obtained from other studies.

Before start with the mathematical principles, it is important to remark in the case of atmospheres, the importance of the stratification, which is the vertical variation of the atmosphere density due to the gravitational compression of the atmospheric gases, resulting in heavier gas underlying lighter gas. Heating received from the Sun may alter this equilibrium, causing vertical motion. But for large-scale motion in the atmosphere or oceans, the movement aspect ratio δ is usually small indicating that the phenomenon to be studied are produced in a thin sheet of fluid where the trajectories of the fluid particles are predominantly horizontal.

$$\delta = \frac{D}{L}$$

D refers to thickness of the fluid layer while L refers the characteristic horizontal scale of the movement. This is a reasonable assumption if we take into account that, for example, a weather system may have a size of thousands of kilometres while the weather phenomena is limited to the troposphere, which is $\approx 12\text{km}$ thick.



3 Fluid motion equations

As the starting point, basic conservation equations are needed to describe motions of the fluid. Those equations are: *mass conservation*, *momentum conservation equation* and *energy conservation equation*. These equations are the basis of the physics every fluid body must follow.

Every fluid particle is characterised by dynamics variables such as its velocity (\vec{u}), and thermodynamic variables such as density (ρ), temperature (T), pressure (p), entropy (s) etc. Moreover, an Eulerian¹ kinematic frame will be used which implies that all those variables are function of time (t) and position vector (\vec{r}).

Starting with the mass conservation principle in the absence of sources or sink of mass the expression results in:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \vec{u} = 0 \quad (3.0.1)$$

In the Shallow Water model density will be considered constant. This is an accurate assumption in oceans and is, perhaps, the weakest assumption for atmospheres, where there is strong stratification, i.e. a strong dependence of density (ρ) with altitude ($\rho = \rho(z)$)

Momentum conservation, also called Second law of Newton, states the relation between pressure forces (P), body forces described as a potential (ϕ) and non conservative forces as friction ($\vec{\mathcal{F}}$) with fluid particle acceleration, which per unit mass is:

$$\frac{D\vec{u}}{Dt} = -\frac{1}{\rho} \nabla p + \nabla \phi + \vec{\mathcal{F}}(\vec{u}) \quad (3.0.2)$$

D/Dt is called total derivative which states the relation between Langrangian and Eulerian frames:

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + (\vec{u} \cdot \nabla)$$

¹In the Eulerian kinematic description the observer is fixed in the space while particles flow through the domain

It is important to notice that molecular friction should be considered even in large-scale motions. Interaction between particles can influence motion at a large scale so an initial model must be considered.

$$\mathcal{F} = \mu \nabla^2 \vec{u} + \frac{\mu}{3} \nabla (\nabla \cdot \vec{u}) \quad (3.0.3)$$

In order to close the dynamical system, the first law of thermodynamic must be used. This equation states that the internal energy per unit mass gained ($e = [\frac{J}{Kg}]$) by an element is equal to its work and heat received.

$$\frac{de}{dt} = Q - W$$

Where Q is heat, per unit mass and time, and W is work per unit mass and time.

Heat gained is based on the contact between fluid particles, so it can be specified by the Fourier equation, and other sources. The heat received by the particle is *positive*, and *negative* when it is expelled by the particle. Work done by the particle is *negative*, and work done on the particle is *positive*.

Positiveness indicates that heat or work may serve to increase particle's internal energy, while negativeness indicates that it diminishes particle's internal energy. Besides, work done by the fluid depends on pressure force. Expressions for the heat and work are given by:

$$Q = \frac{\kappa_T}{\rho} \nabla^2 T \quad \text{and} \quad W = -\frac{p}{\rho^2} \frac{D\rho}{Dt}$$

Where heat is considered only by conduction. κ_T is known as the thermal conductivity. After those considerations, the energy equation can be expressed as:

$$\rho \frac{de}{dt} = -p \rho \frac{d}{dt} \rho^{-1} + \kappa \nabla^2 T \quad (3.0.4)$$



By considering constant density equation (3.0.1) can simplify to the incompressible fluid condition:

$$\nabla \cdot \vec{u} = 0$$

Since:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = \frac{D\rho}{Dt} + \rho \nabla \cdot \vec{u} = 0,$$

Incompressibility by itself does not imply that $D\rho/Dt$ vanishes, it is still present in the energy equation and only is neglected if conductive and internal heating are neglected too. (i.e. adiabatic motion). But constant density does.

4 Rotating coordinate frame

In the last section, fluid motion equations were introduced in a non rotating and Cartesian frame. However, the main phenomena of the study (currents and winds) are on a spinning planet and therefore on a rotating coordinate frame. This angular motion produces non-inertial forces such as Coriolis, which may be dominant in the balance of forces.

Consider a spherical homogeneous planet with constant angular motion $\vec{\Omega}$ and the coordinate system positioned on the planet's center² rotating at the same angular speed. In this case, equations of motion described in section (3) must be corrected. Rotating to inertial system relations can be used in order to obtain fictitious accelerations on the rotating reference frame. Given an arbitrary vector \vec{B} a relation for the variation of \vec{B} in the inertial and rotating frame can be established.

$$\left(\frac{d\vec{B}}{dt}\right)_I = \left(\frac{d\vec{B}}{dt}\right)_R + \vec{\Omega} \times \vec{B}$$

Where \vec{B} is an arbitrary vector and sub-index I implies inertial frame while sub-index R implies rotating frame. Applying this concept to the position vector (\vec{r}) a relation between velocities in both frames can be established.

$$\left(\frac{d\vec{r}}{dt}\right)_I = \left(\frac{d\vec{r}}{dt}\right)_R + \vec{\Omega} \times \vec{r} \rightarrow \vec{u}_I = \vec{u}_R + \vec{\Omega} \times \vec{r}$$

\vec{u}_R is defined as the velocity relative to the rotating frame and \vec{u}_I is the velocity in the inertial frame. The same process is repeated to obtain the acceleration.

$$\left(\frac{d\vec{u}_I}{dt}\right)_I = \left(\frac{d\vec{u}_R}{dt}\right)_R + \vec{\Omega} \times \left(\frac{d\vec{r}}{dt}\right)_R + \frac{d\vec{\Omega}}{dt} \times \vec{r} + \vec{\Omega} \times (\vec{u}_R + \vec{\Omega} \times \vec{r}) = \vec{a}_I \quad (4.0.1)$$

In the last expression the *Coriolis acceleration* is obtained as : $2\vec{\Omega} \times \vec{u}_R$ and the *Centripetal acceleration* as: $\vec{\Omega} \times (\vec{\Omega} \times \vec{r})$. The resulting Coriolis force is perpendicular to the fluid velocity, so it does no work.

²As an approximation, we will consider a planet with a homogeneous interior with a constant mean density

Centripetal acceleration can be written as the gradient of a potential (ϕ_c) which can be included with the potential gravity (ϕ) in the momentum equation (3.0.3). It can be shown that $\phi_c = -\frac{|\vec{\Omega} \times \vec{r}|^2}{2}$

$$-\nabla \phi_c = \vec{\Omega} \times (\vec{\Omega} \times \vec{r}) \rightarrow \phi_T = \phi + \phi_c$$

Finally, the Coriolis force appears into the momentum equation when acceleration \vec{a}_I is substituted. The angular velocity $\vec{\Omega}$ of the planet is considered constant, and therefore $\frac{d\vec{\Omega}}{dt} = 0$. The momentum equation results:

$$\rho \left[\frac{d\vec{u}}{dt} + 2\vec{\Omega} \times \vec{u}_r \right] = -\nabla p + \rho \nabla \vec{\phi} + \mathcal{F} \quad (4.0.2)$$

It is important to notice that spatial gradients of pressure and potential will be independent of the frame of reference. The same happens with scalar variables such as density and temperature, that's why equations (3.0.1) and (3.0.4) are coordinate frame independent. Nevertheless, \mathcal{F} will depend on its relation with velocity. For a Newtonian fluid with the expression adopted in (3.0.3) can be stated that:

$$\mathcal{F}(\vec{u}_I) = \mathcal{F}(\vec{u}_R)$$

Generally, fluid motion will be described in this rotating frame and consequently dynamic variables will be measured in a no inertial frame. In subsequent sections we will drop the subindex r in the velocity so that $\vec{u} = \vec{u}_r$.

4.1 Rossby Number

As it is said before, a rotating frame coordinate produces non-inertial Coriolis forces which may be relevant at large scale motion. This concept must be constrained specifying what *large motion scale* are for atmospheres (or oceans) and when the Coriolis effect starts to be relevant. In the Shallow Water model applied to atmospheres large-scale motions are those which are influenced by planet's rotation as in the case of fast rotating planets such as Earth.

Comparing the importance of the movement to the planet's angular motion gives a non-dimensional number called the *Rossby number*. Let's characterise the horizontal spatial variation of the dynamic field as L . In the same way, horizontal velocity of fluid motion can be

characterised by the magnitude U . The period for an element fluid to displace a distance L at velocity U is: $T = \frac{L}{U}$.

Rossby number can be defined as the comparison of T with the planet's rotation. It will be denoted by R_o

$$R_o = \frac{U}{2\Omega L} \leq 1 \quad (4.1.1)$$

The factor 2 is included in the denominator to be consistent with other derivations of the *Rossby* number. When the time scale of fluid motion exceeds the planet's rotation period, the Coriolis effect beats relative acceleration obtaining a Rossby number less than unit ($R_o \leq 1$). This is caused by the tendency of the fluid to reduce its relative acceleration at large-scale while Coriolis effect remains constant.

5 Vorticity

Vorticity and potential vorticity play a central role in geophysical fluid dynamics and its evolution governs important phenomenon. So before introducing the *Shallow Water* model, it is important to look at the concept.

Vorticity is defined as the curl of the velocity field.

$$\vec{\omega} = \nabla \times \vec{u} \quad (5.0.1)$$

In a Cartesian frame, vorticity can be expressed as a function of velocity as.

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} \frac{\partial w}{\partial y} - \frac{\partial v}{\partial z} \\ \frac{\partial u}{\partial z} - \frac{\partial w}{\partial x} \\ \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \end{bmatrix}$$

Vorticity also depends on the frame of reference. $\vec{\omega}_a$ is the called *absolute vorticity* which is defined as as the curl of the inertial velocity when observed in the non rotating frame:

$$\vec{\omega}_a = \nabla \times [\vec{u} + \vec{\Omega} \times \vec{r}] = \vec{\omega} + 2\vec{\Omega} \quad (5.0.2)$$

Last equation splits absolute vorticity into two distinct terms. $\vec{\omega}$ is the relative vorticity while $2\vec{\Omega}$ is the planetary vorticity. The component of the planetary vorticity normal to the planet's surface is the *Coriolis parameter* (f):

$$f = 2\Omega \sin(\varphi)$$

Where φ is the latitude of the planet's surface.

5.1 Potential Vorticity

From the time derivation of the absolute vorticity and the use of the mass conservation equation a new theorem can be established (see derivation in Peldosky 1983 [13])

$$\frac{d}{dt} \left(\frac{\vec{\omega}_a}{\rho} \right) = \left(\frac{\vec{\omega}_a}{\rho} \cdot \nabla \right) \vec{u} + \nabla \rho \times \frac{\nabla \rho}{\rho^3} + \left(\nabla \times \frac{\mathcal{F}}{\rho} \right) \frac{1}{\rho} \quad (5.1.1)$$

Consider λ a scalar property like temperature and Ψ an unspecific source of the property λ . So that:

$$\frac{d\lambda}{dt} = \Psi$$

If the dot product of $\nabla \lambda$ and (5.1.1) is taken, it can be shown (Peldosky, 1983) that:

$$\frac{d}{dt} \left(\frac{\vec{\omega}_a}{\rho} \cdot \nabla \lambda \right) = \left(\frac{\vec{\omega}_a}{\rho} \cdot \nabla \Psi \right) + \nabla \lambda \cdot \left(\frac{\nabla \rho \times \nabla p}{\rho^3} \right) + \frac{\nabla \lambda}{\rho} \cdot \left(\nabla \times \frac{\mathcal{F}}{\rho} \right) \quad (5.1.2)$$

If the fluid fulfills the following conditions:

- λ is a conserved quantity for each fluid element ($\Psi=0$) → for example adiabatic motion when $\lambda = \theta$ (θ = potential temperature)
- The frictional force is negligible $\mathcal{F} = 0$
- The fluid is barotropic ($\nabla \rho \times \nabla p = 0$) or λ can be considered function only of p and ρ

then:

$$\frac{d\Pi}{dt} = 0$$

Where

$$\Pi = \frac{\vec{\omega} + 2\vec{\Omega}}{\rho} \cdot \nabla \lambda \quad (5.1.3)$$

is called the *potential vorticity*

Consider two surfaces (1 and 2) separated by the small distance δl , where the property λ has the respective constant value λ_0 on surface 1, and $(\lambda_0 - \Delta\lambda)$ on surface 2. If Π is conserved in a fluid element, represented by the cylinder in Figure (1), when the distance δl between the two surfaces increases, the gradient $\nabla \lambda = \frac{\Delta\lambda}{\delta l}$ decreases and absolute vorticity must increase to keep potential vorticity constant (assuming constant density).

Potential vorticity is a relevant quantity in geophysical fluid dynamics, and it will reveal as a very useful in the *Shallow Water* model.

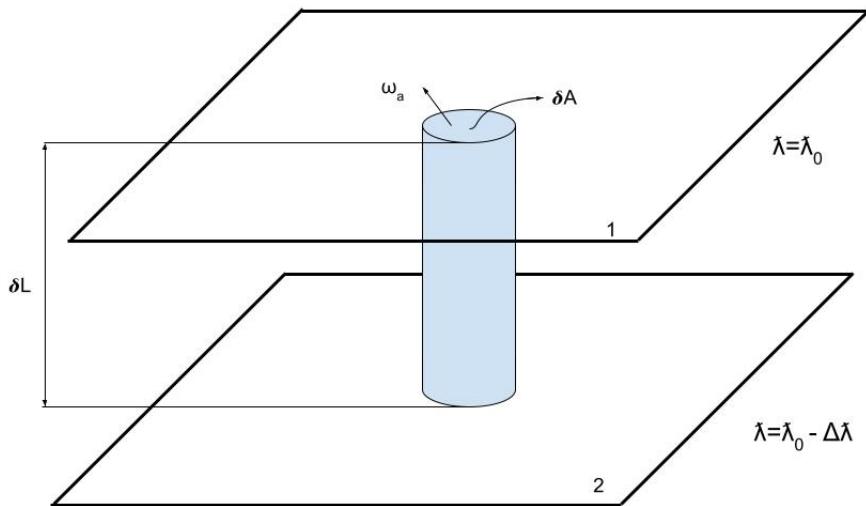


FIGURE 1: Element of mass between two surfaces of constant λ

6 The Shallow Water model

The *Shallow Water* model (SW) implemented on this report is reduced to a two-dimensional domain in a cartesian frame of coordinates. Later, the same equations will be modified to an ellipsoidal frame of coordinates in order to apply it to the giant planets' geometry.

In the *SW* model the atmosphere is reduced to an homogeneous, incompressible, inviscid and thin layer of fluid where $\delta \ll 1$. These simplifications still keep much physics of large-scale phenomena.

The inviscid hypothesis can be assumed if large scale is considered, where friction \mathcal{F} is described by equation (3.0.3). The ratio between frictional force per unit mass to Coriolis acceleration is the called *Ekman number*.

$$E = \frac{\nu}{2\Omega L^2}$$

At large-scale, *Ekman number* is very small, which implies that frictional forces are negligible in front of Coriolis force.

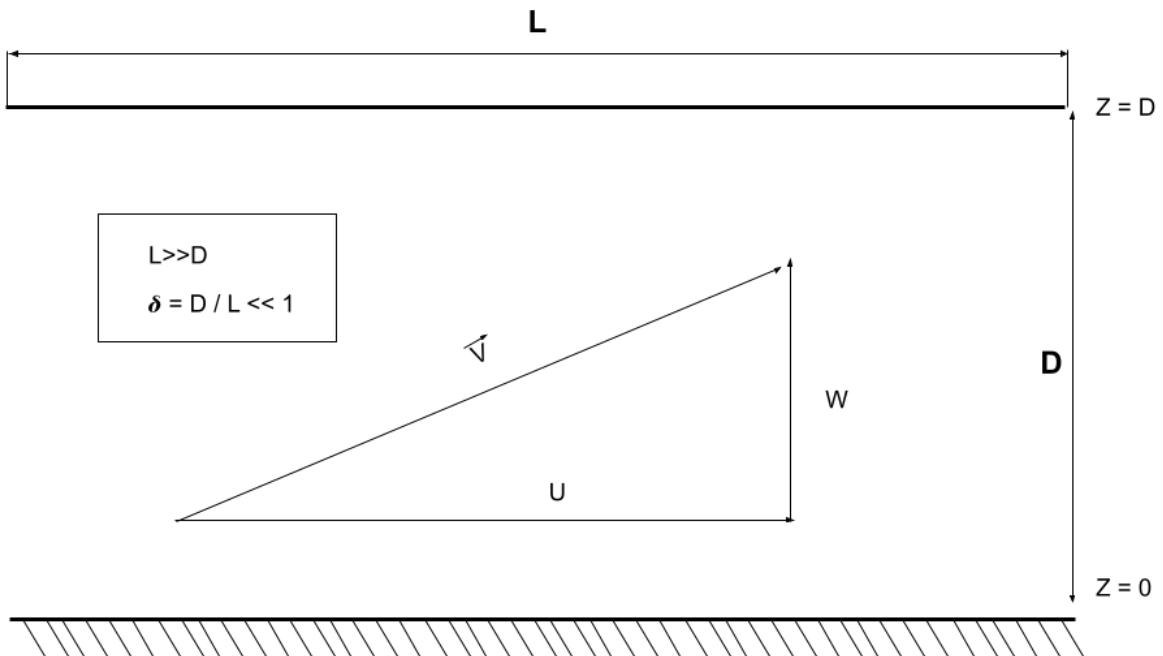


FIGURE 2: Ratio between horizontal plane and vertical ratio

The planetary fluid's layer can be described as a thin sheet of fluid from a reference level ($z=0$) at the bottom. In order to establish its height, some variables are defined: $h(x, y, t)$ represents the fluid's free surface height from the reference ($z = h_b$) while $h_b(x, y)$ is the rigid bottom (see Figure(3)). The topography fluid's free surface changes with x and y coordinates, so a characteristic length is defined: D . η describes the free surface perturbation added on the unperturbed free surface at $z = D$

Finally, some additional considerations are made. First, the body force is modeled by the gravity vector ($\vec{g} = -g\vec{k}$) perpendicular to the fluid's layer. The rotation axis of the fluid is parallel to the z axis. Since the fluid is homogeneous, density is constant ($\rho = \rho_0$).

One of the consequences of considering $\delta \ll 1$ is that the number of dynamical equations is reduced by one. In the same way, dependent variables are reduced by one (The vertical component of velocity, w , can be readily calculated from the horizontal components u and v) and the number of independent variables are reduced in the same order (z will not appear in the dynamical equations). Figure (2) represents the geometric relation between the horizontal and vertical velocities of the atmospheric layer.

Orders of magnitude for each velocity component can be described using the geometric dimensions D and L . This is, for the movement in the fluid layer to be consistent $D = WT$ and $L = UT$, where T refers to the time order of magnitude. These relations lead to:

$$\frac{D}{L} = \frac{W}{U}$$

$$W = \frac{D}{L}U = \delta U \quad (6.0.1)$$

6.1 Shallow Water equations

The derivation of the Shallow Water equations is performed following the studies of: *Pedlosky (1987) [13]*, *Zeitlin (2018) [16]*, *Holton (2004) [7]* and *Arnaud Sabatés (2018) [14]*

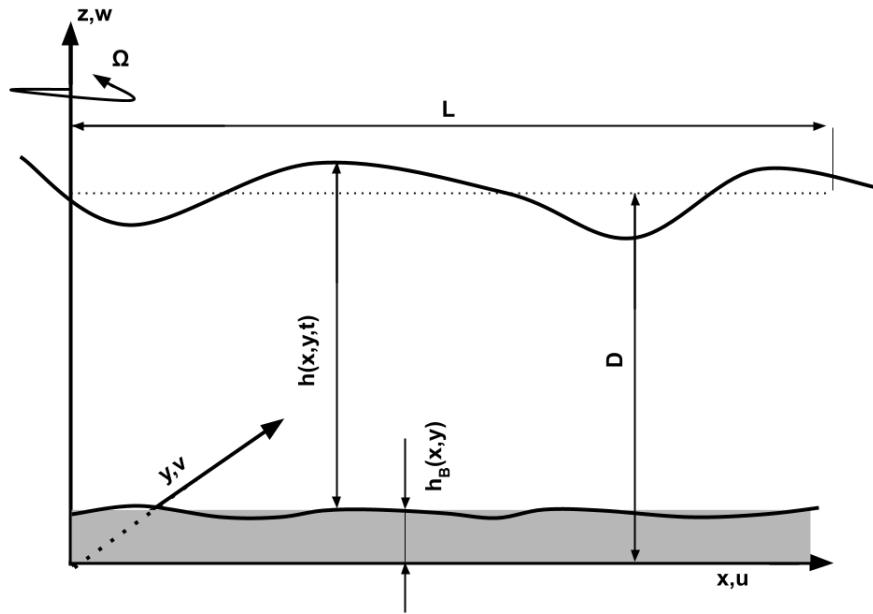


FIGURE 3: Shallow Water scheme

The geometry and rotation of the problem is depicted in Figure (3), representing a thin layer of fluid. For illustrative reasons, D and L are not scaled, as the figure would not be readable. From previous discussions the reader should bear in mind that on $D \ll L$. According to Figure(3), the following geometric relation can be established:

$$h(x, y, t) = D + \eta(x, y, t) - h_B(x, y, t)$$

Assuming the hypothesis of inviscid and incompressible fluid the energy equation is not needed. Equation (4.0.2) in component form including the Coriolis parameter f is written like.

$$\frac{\partial u}{\partial t} + [u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z}] - fv = -\frac{1}{\rho} \frac{\partial p}{\partial x} \quad (6.1.1a)$$

$$\frac{\partial v}{\partial t} + [u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z}] + fu = -\frac{1}{\rho} \frac{\partial p}{\partial y} \quad (6.1.1b)$$

$$\frac{\partial w}{\partial t} + [u \frac{\partial w}{\partial x} + v \frac{\partial w}{\partial y} + w \frac{\partial w}{\partial z}] = -\frac{1}{\rho} \frac{\partial p}{\partial z} - g \quad (6.1.1c)$$

Total pressure in the fluid's layer is expressed as:

$$p(x, y, z, t) = -\rho g z + \tilde{p}(x, y, z, t) \quad (6.1.2)$$

Where it is considered that the fluid at rest is in hydrostatic equilibrium and $\tilde{p}(x, y, z, t)$ is a pressure perturbation.

Introducing this expression into (6.1.1) and deriving it, the only term left is the perturbation filed for the three components, cancelling the gravity in (6.1.3c)

$$\frac{\partial u}{\partial t} + [u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z}] - fv = -\frac{1}{\rho} \frac{\partial \tilde{p}}{\partial x} \quad (6.1.3a)$$

$$\frac{\partial v}{\partial t} + [u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z}] + fu = -\frac{1}{\rho} \frac{\partial \tilde{p}}{\partial y} \quad (6.1.3b)$$

$$\frac{\partial w}{\partial t} + [u \frac{\partial w}{\partial x} + v \frac{\partial w}{\partial y} + w \frac{\partial w}{\partial z}] = -\frac{1}{\rho} \frac{\partial \tilde{p}}{\partial z} \quad (6.1.3c)$$

The relevance of each of the terms in (6.1.3) may be determined with an order of magnitude analysis, using the same notation as in (6.0.1) adding \tilde{P} for the pressure field and T for the reference time scale.

$$\frac{U}{T} + \frac{U^2}{L} + \frac{U^2}{L} + \frac{UW}{D} - fU \approx -\frac{1}{\rho} \frac{\tilde{P}}{L} \quad (6.1.4a)$$

$$\frac{U}{T} + \frac{U^2}{L} + \frac{U^2}{L} + \frac{UW}{D} + fU \approx -\frac{1}{\rho} \frac{\tilde{P}}{L} \quad (6.1.4b)$$

$$\frac{W}{T} + \frac{UW}{L} + \frac{UW}{L} + \frac{W^2}{D} \approx -\frac{1}{\rho} \frac{\tilde{P}}{D} \quad (6.1.4c)$$

if equations (6.1.4) are to be satisfied, the orders of magnitude of the pressure gradients should be similar to that of the accelerations. As the reader may have deduced, acceleration terms have equivalent expressions using the following relations: $W = \frac{D}{L}U$, $U = \frac{L}{T}$. Then, it is verified that

$$\frac{U}{T} = \frac{U^2}{L} = \frac{UW}{D} \quad (6.1.5a)$$

$$\frac{W}{T} = \frac{UW}{L} = \frac{W^2}{D} \quad (6.1.5b)$$

If $T = L/U$ is assumed, which is reasonable because time scale T must be of the order of the time needed to span L at velocity U . From the (6.1.4a) and (6.1.4b), deduced that

$$\mathcal{O}(\tilde{P}) = \mathcal{O}\left[\rho L \max\left(\left\{\frac{U^2}{L}, fU\right\}\right)\right] \quad (6.1.6)$$

Where $\mathcal{O}()$ means 'order of'. Equation (6.1.6) is expressed for the x and y components, but similar interpretation can be done for the remaining component z , as it is divided by D instead of L , ensuring that it has the same order of magnitude as the acceleration terms. Looking at the ratio between the vertical pressure gradient and vertical acceleration, the following relation is obtained:

$$\rho \frac{\frac{dw}{dt}}{\frac{\partial p}{\partial z}} \approx \rho \frac{\frac{WU}{L}}{\frac{\tilde{P}}{D}} = \delta^2 \frac{\frac{U}{L}}{\max\left(\left\{\frac{U}{L}, f\right\}\right)} = \delta^2 \min(\{1, R_0\}) \quad (6.1.7)$$

being δ the ratio $\frac{D}{L}$. Here $R_0 = \frac{U}{fL}$. As the Rossby number R_0 for atmospheric phenomena, in a fast rotating planets, is always less than 1 —see equation (4.1.1)— and, of course, $\delta^2 \ll 1$, vertical accelerations are far less relevant than pressure gradients in the z direction. Moreover, this also indicates that $\frac{\partial \tilde{P}}{\partial z}$ must be δ^2 times smaller than the other (horizontal) pressure gradients, as is the case of $\frac{Dw}{Dt}$. The hydrostatic model is then verified in (6.1.8)

$$\frac{\partial p}{\partial z} = -\rho g + \frac{\partial \tilde{P}}{\partial z} = -\rho g + \mathcal{O}(\delta^2) \approx -\rho g \quad (6.1.8)$$

Integration of the pressure gradient (6.1.8) between $z = 0$ and z is performed.

$$p = -\rho g z + A(x, y, t) \quad (6.1.9)$$

in which A is the integration constant. Note that A does not depend on the vertical position.

Assuming that at the fluid's free surface $z = D + \eta$ the pressure is constant and equal to p_0 on the surface,

$$\begin{cases} p(x, y, z = D + \eta) = p_0 \\ -\rho g(D + \eta) + A(x, y, t) = p_0 \end{cases}$$

$$A(x, y, t) = p_0 + \rho g(D + \eta) \quad (6.1.10)$$



Incorporating this result to the pressure expression in (6.1.9):

$$p(x, y, z, t) = -\rho g z + p_0 + \rho g D + \rho g \eta = \rho g (\eta - z) + \rho g D + p_0$$

It is important to notice that p_0 and $\rho g D$ are constants. Pressure gradients can be rewritten,

$$\frac{\partial p}{\partial x} = \rho g \frac{\partial \eta}{\partial x} \quad (6.1.11a)$$

$$\frac{\partial p}{\partial y} = \rho g \frac{\partial \eta}{\partial y} \quad (6.1.11b)$$

$$\frac{\partial p}{\partial z} = -\rho g \quad (6.1.11c)$$

With all the performed simplifications, the expressions for the conservation of linear momentum of the *Shallow Water* model are reduced to two components, these being written in (6.1.12), showing its independence from z since η is independent of z .

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} - fv = -g \frac{\partial \eta}{\partial x} \quad (6.1.12a)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + fu = -g \frac{\partial \eta}{\partial y} \quad (6.1.12b)$$

For homogeneous ($\rho = cnst$) flow, the first equation form (4.0.2) corresponding to the conservation of mass or continuity equation may be simplified, as variation of density is cancelled from the expression. Thus, conservation of mass is reduced to the divergence of the velocity equal to zero (6.1.13), which means that the volume of fluid elements is conserved.

$$\nabla \cdot \vec{u} = 0$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \quad (6.1.13)$$

A similar treatment as in the pressure term is applied to the continuity equation. Integration of the vertical velocity component through z is straightforward because horizontal velocities are z independent, and an integration constant, say $B(x, y, t)$, arises — see Equation (6.1.14)—



$$\frac{\partial w}{\partial z} = -\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y}$$

$$w(x, y, z, t) = -\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)z + B(x, y, t) \quad (6.1.14)$$

This integration constant can be obtained at the bottom of the fluid layer, that is, $w(h_B) = w_0$. Note that, by definition, the vertical velocity is found through the material derivative in the z direction when $z = h_B$, this being $w = \frac{D(h_B)}{Dt}$.

$$w_0 = \frac{D(h_B)}{Dt} = \frac{\partial h_B}{\partial t} + u \frac{\partial h_B}{\partial x} + v \frac{\partial h_B}{\partial y}$$

$$-\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)h_B + B(x, y, t) = \frac{\partial h_B}{\partial t} + u \frac{\partial h_B}{\partial x} + v \frac{\partial h_B}{\partial y}$$

$$B(x, y, t) = \frac{\partial h_B}{\partial t} + u \frac{\partial h_B}{\partial x} + v \frac{\partial h_B}{\partial y} + \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)h_B \quad (6.1.15)$$

In (6.1.15). the topography is deemed to also vary with time, as will be the case if several layers are simulated simultaneously. Equation (6.1.16) regroups terms for future manipulation.

$$w(x, y, z, t) = -\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)z + \frac{D(h_B)}{Dt} + \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)h_b = \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)(h_B - z) + \frac{D(h_B)}{Dt} \quad (6.1.16)$$

Following the same process as in (6.1.15), the derivative of $z = D + \eta(x, y, t)$, showed in (6.1.17), is used for the calculation of the vertical velocity on the surface.

$$w(x, y, z, t) = \frac{D\eta}{Dt} = \frac{\partial \eta}{\partial t} + u \frac{\partial \eta}{\partial x} + v \frac{\partial \eta}{\partial y} \quad (6.1.17)$$

The derivatives of constant D are zero, so they may be introduced arbitrarily to perform the substitution $h = \eta + D - h_B$

$$\frac{D\eta}{Dt} = \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)(h_B - D - \eta) + \frac{D(h_B)}{Dt}$$

$$\frac{D}{Dt}(\eta - h_B + D) - \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)(h_B - D - \eta) = 0$$

$$\frac{\partial(h)}{\partial t} + u \frac{\partial(h)}{\partial x} + v \frac{\partial h}{\partial y} + \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)h = 0 \quad (6.1.18)$$

In order to simplify (6.1.18), the product rule for derivatives is applied as follows.

$$\frac{\partial(hu)}{\partial x} + \frac{\partial(hv)}{\partial y} = u \frac{\partial(h)}{\partial x} + v \frac{\partial h}{\partial y} + \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) h \quad (6.1.19)$$

So (6.1.18) is now (6.1.21)

$$\frac{\partial h}{\partial t} + \frac{\partial(hu)}{\partial x} + \frac{\partial(hv)}{\partial y} = 0 \quad (6.1.20)$$

$$\frac{\partial h}{\partial t} + \nabla \cdot (h\vec{u}) = 0 \quad (6.1.21)$$

However, in the case of this study, the topography h_B , if existing, will be constant with time, that is, $h_B = h_B(x, y)$ as only one layer of the atmosphere will be simulated. This fact enables the substitution of h by η in the partial time derivative as in (6.1.22). **The continuity equation** for the Shallow Water model finally is.

$$\frac{\partial \eta}{\partial t} + \nabla \cdot (h\vec{u}) = 0 \quad (6.1.22)$$

Eventually, **the Shallow Water model** for homogeneous inviscid, incompressible, one-layer flow is defined using three partial differential equations and a geometrical constraint, as showed in the set of expressions (6.1.23).

$$\left. \begin{aligned} \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} - fv &= -g \frac{\partial \eta}{\partial x} \\ \frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + fu &= -g \frac{\partial \eta}{\partial y} \\ \frac{\partial \eta}{\partial t} + \nabla \cdot (h\vec{u}) &= 0 \\ h - D - \eta + h_B &= 0 \end{aligned} \right\} \quad (6.1.23)$$

6.2 Potential vorticity applied to Shallow Water model

Conservation of potential vorticity is introduced in the model as a passive tracer reminiscent to a cloud for the visualisation of the results. Its expression can be readjusted to the model variables. In the same way as the *SW* equations, it is expressed in a Cartesian frame of coordinates.

The components of vorticity are:

$$\omega_x = \frac{\partial w}{\partial y} - \frac{\partial v}{\partial z} \quad (6.2.1)$$

$$\omega_y = \frac{\partial u}{\partial z} - \frac{\partial w}{\partial x} \quad (6.2.2)$$

$$\omega_z = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \quad (6.2.3)$$

The order of magnitude of the first two components of vorticity are δ times smaller than vertical component.

$$O(\omega_x) = O(\omega_y) = \delta O(\omega_z)$$

This fact allows to consider relative vorticity as $\vec{\omega} \approx \vec{\omega}_z$. Potential vorticity is obtained from the conservative fluid's quantity λ (*see section(5.1)*). For the *SW* model is taken:

$$\lambda = \frac{z - h_B}{h}$$

So potential vorticity:

$$\Pi = \frac{\omega}{\rho} \cdot \nabla \lambda = \frac{\omega + f}{\rho} \frac{\partial}{\partial z} \left(\frac{z - h_B}{h} \right) = \frac{\omega + f}{\rho h}$$

Since in the *SW* model density is an irrelevant constant, potential vorticity is written independently of density.

$$\Pi_S = \frac{\omega + f}{h} \quad (6.2.4)$$

In the model implemented in this project, as depicted in Figure(3), $f = 2\Omega$ but in general if the *SW* is applied on a sphere, f is the Coriolis parameter, which correspond to the vertical component of the planetary vorticity.



7 Shallow Water Implementation

Once *Shallow Water* equations are explained, the next step is to implement them in a computational model. To do so, different computational methods and algorithms will be described in the following sections. As it is said before, this model can be used for both, planetary and oceanic simulations.

SW implementation must start with the collocation of the u , v , η variables in the domain and the computational methods used to solve them. It is important to notice that, every algorithm must be correctly validated with the so called *MMS* method explained in section(8), this will avoid errors in the final main code.

Once the domain is correctly divided, next step is to start with some numerical experiments where some new numerical concepts will be established such as the boundary conditions (periodic or full slip as will be seen later) The choice will depend on what physical event is being represented: ocean portion, planetary strip, closed tank etc.

In the second part of this section, the initial code will be extended in order to simulate more real situations like planetary atmospheres. This will be done thought the addition of some characteristics effects such as Coriolis, zonal winds, specific planetary perturbations or vortices. Discretization of the equations must be changed too, being adapted to the planetary shape. Shallow Water equations (6.1.23) will be transformed and the code will have to be re-adapted.

7.1 Arakawa's grids

In the first simulation, the fluid's domain will be an $(L \times L)m^2$ flat square divided in cells of dimensions $(\Delta_x \times \Delta_y)$ where $\Delta_x = \Delta_y$. Later, those dimensions and geometry is transported to the atmosphere on a spherical (Earth) or ellipsoidal (Jupiter, Saturn) planet. Next step is to wonder how variables (u, v and η) are distributed along the grid, distribution will influence fundamental properties of the model such as the accuracy order of the equations, the frequency of propagation waves, etc. Let's consider a one-dimension model where variables u and η are positioned in the same point of a linear mesh as can be seen in Figure(4). Time derivatives of those variables are evaluated at point j , as well as $\frac{\partial u}{\partial x}$ and $\frac{\partial \eta}{\partial x}$ which are expressed like:

$$\begin{aligned}\frac{\partial u_j}{\partial x} &= \frac{u_{j+1} - u_{j-1}}{2\Delta_x} \\ \frac{\partial \eta_j}{\partial x} &= \frac{\eta_{j+1} - \eta_{j-1}}{2\Delta_x}\end{aligned}$$

Space increment is $2\Delta_x$, while in the mesh, cell size is only Δ_x . Also, with this type of discretization, high frequency modes are not well simulated producing artifacts. To mitigate this problem, staggered grids are applied, it means that velocity field components and scalar values (η) are not collocated at the same point, as can be seen in Figure(5). In this case, $\frac{\partial u}{\partial t}$ is evaluated at position $j + 1/2$ and $\frac{\partial \eta}{\partial t}$ is evaluated at position j achieving a second order precision with the Δ_x spatial increment:

$$\begin{aligned}\frac{\partial u_j}{\partial x} &= \frac{u_{j+1/2} - u_{j-1/2}}{\Delta_x} \\ \frac{\partial \eta_{j+1/2}}{\partial x} &= \frac{\eta_{j+1} - \eta_j}{\Delta_x}\end{aligned}$$

Staggered grids offer a computational advantage since the numerical discretization is of second order without increasing spatial increments, and they can better represent wave propagation through a better evaluation of pressure gradients; (see for instance *Cushman-Roisin and Beckers (2011)* [6])

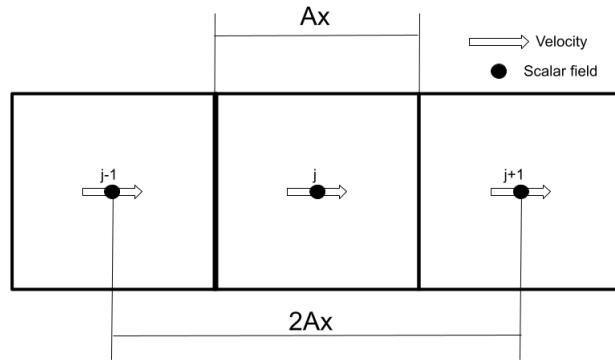


FIGURE 4: Finite differences. Centred grid

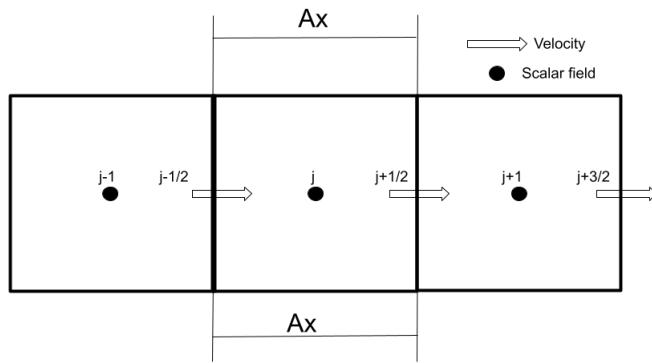


FIGURE 5: Finite differences. Staggered grid

In addition, high frequency modes ($\lambda = 2\Delta_x$) are not captured by the mesh represented in Figure(4) as will be shown below. Let's assume an oscillating saw-tooth perturbation $\eta(x)$ in a 1D Shallow Water linear system with no rotation in the non-staggered grid as represented in Figure(4):

$$\frac{\partial u}{\partial t} = -g \frac{\partial \eta}{\partial x}$$

$$\frac{\partial \eta}{\partial t} = -H \frac{\partial u}{\partial x}$$

Where $\eta \ll H = constant$

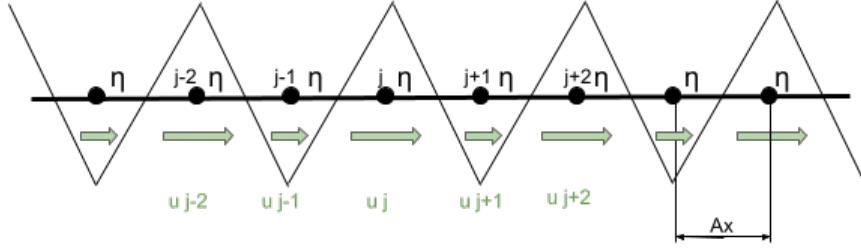


FIGURE 6: Saw-tooth function in a centred grid. Solid point represent surface perturbation η . Saw-tooth solid line represents the continuous η

If pressure gradient for u_j is computed, then the most intuitive approach, by using finite differences, is making (where \tilde{u}_j is the numerical solution):

$$\frac{\partial \tilde{u}_j}{\partial t} = -g \frac{\eta_{j+1} - \eta_{j-1}}{2\Delta_x}$$

But $\eta_{j+1} - \eta_{j-1} = 0$, so the high frequency wave can not be seen by the model.

If instead a staggered grid is used, see Figure(7), now can be observed that high frequency dynamic pressure differences are captured. As matter of fact, the 1D staggered grid represents

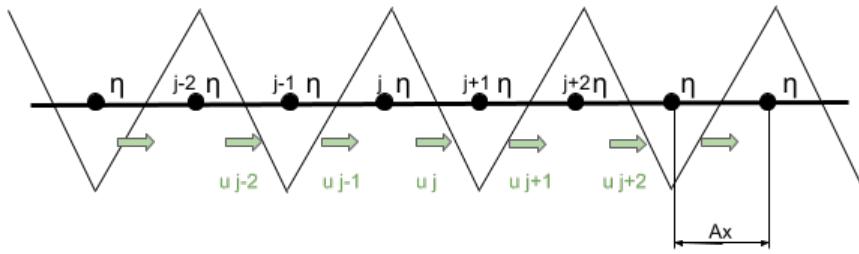


FIGURE 7: Saw-tooth function in a staggered grid

better wave phenomena than the non-staggered collocation for velocity and pressure.

Once outlined why the 1D staggered grid is superior, *Arakawa's 2D grids* can be introduced. Its name is given in honor of Akio Arakawa ³, who defined five types of staggered meshes in a two dimensional domain (Figure (9)). They are named with letters (A,B,C,D,E) and variables u , v and η are differently positioned depending on the mesh type.

³Born in 1927, is a climate scientist famous for his studies about atmospheres and oceans at the University of California, Los Angeles

For the Shallow Water model, the *C – grid*, Figure(8) and Figure(9), is one of the better choices, due to the fact that the pressure gradient is compute in a natural way without averaging, and advection and pressure terms can be computed with a second order approximation. However, Coriolis effect implementation is not trivial because velocity components u and v are not equally positioned, and an interpolation needs to be done. For example, in the x momentum equation, the v component appears at the Coriolis term at the u position so it is computed such as:

$$v_{j+1/2,i} = \frac{\frac{v_{i+1/2,j} + v_{j+1,i+1/2}}{2} + \frac{v_{i-1/2,j} + v_{j+1,i-1/2}}{2}}{2}$$

but in atmospheric applications this is not serious because the Rossby radius of deformation is usually much large than grid resolution, so the interpolation of the Coriolis force does not introduce an interpolation error. Therefore the *C-grid* is a good choice for atmospheric simulations.

On the other hand, the *B – grid* has the velocity components centred at the same point, so Coriolis does not need any interpolation. The choice between both meshes depends on the most predominantly phenomenon in the equations, this is pressure gradient or Coriolis effect. *C-grid* presents less error for a big number of elements when the Rossby radius of deformation is well resolved (see *Geophysical fluids dynamics, 2010* [6]) and pressure gradient is better represented at those types of mesh. Although the Coriolis effect is better represented at *B-grid*, its computation is accurate enough for *C-grid* if mesh discretization is fine. The *A-grid* introduces artefacts such as the *checker board mode*. The *E-grid* is a rotated *B-grid*.

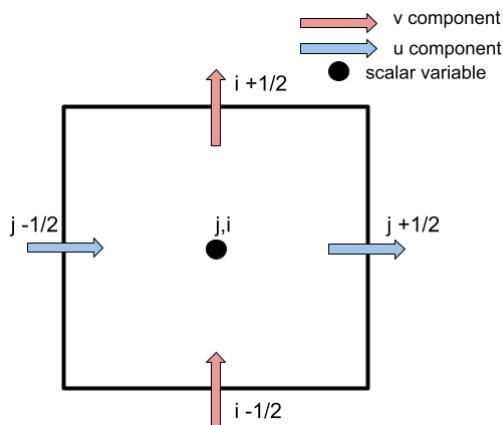


FIGURE 8: C-grid variables distribution

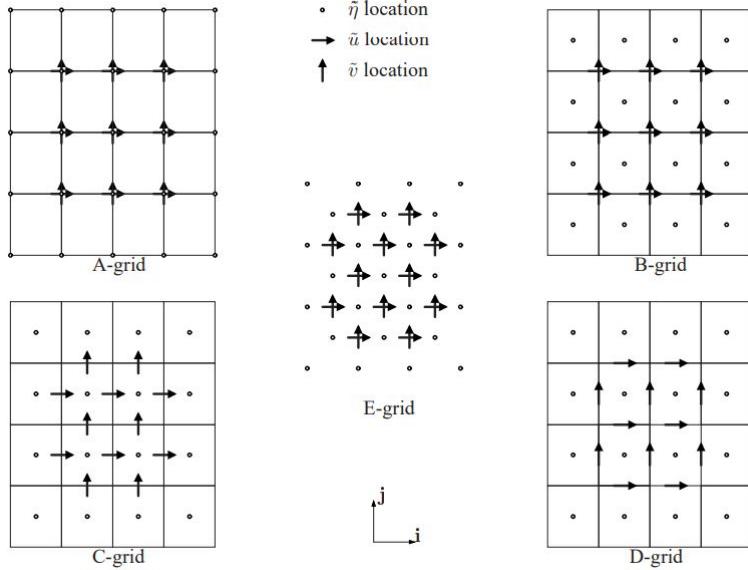


FIGURE 9: Arakawa's grids. From Geophysical fluid dynamics (2010) [6]

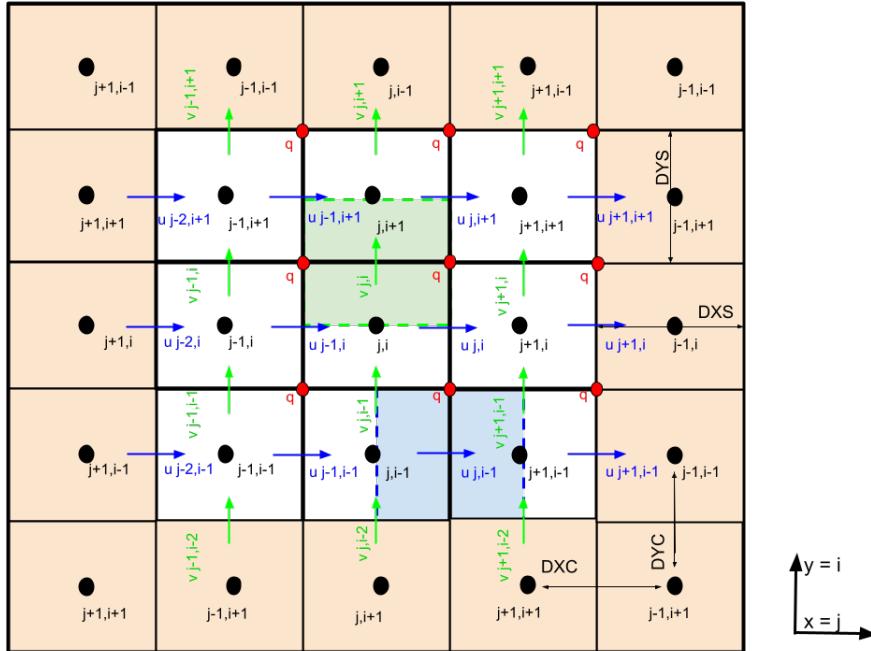


FIGURE 10: Arakawa C-grid

In Figure(10), black points are the cells' centres where variable η is computed and q stands for the potential vorticity (to be discussed later). Cells with different colour around white cells constitute the 'halo' around the domain. It is used to apply boundary conditions to the model.

Distances between staggered nodes are denominated as DXS and DYS respectively, while distance between centred nodes are DXC and DYC

7.2 Finite Difference Method. FDM

The finite Difference Method (*FDM*) is used in numerical computation in order to approximate derivatives by finite-differences. To do so Taylor series are used.

A Taylor series of a function $f(x_0 - h)$ is expressed as:

$$f(x_0 + h) = f(x_0) + \frac{f'(x_0)}{1!}h + \frac{f''(x_0)}{2!}h^2 + \frac{f^n(x_0)}{n!}h^n + R_n(x) \quad (7.2.1)$$

Where $R_n(x)$ is the difference between the real function and the Taylor approximation value (this is the truncation error). Truncating the polynomial expression at the first value and dividing each term by h , an approximation to the first derivative is obtained.

$$\frac{f(x_0 + h)}{h} = \frac{f(x_0)}{h} + f'(x_0) + \frac{R_1(x)}{h} \quad (7.2.2)$$

$$f'(x_0) = \frac{f(x_0 + h) - f(x_0)}{h} - \frac{R_1(x)}{h} \quad (7.2.3)$$

Taking into account the staggering of the Arakawa-C grid, an approximation for $\partial\eta/\partial x$ in the momentum equation at the $u_{j,i}$ position is computed below. $u_{j,i}$ is between $\eta_{j,i}$ and $\eta_{j+1,i}$ at a distance $\Delta x/2$ from $\eta_{j,i}$ and $\eta_{j+1,i}$. Now let's estimate $\partial\eta/\partial x$ at the position $u_{j,i}$, by using Taylor series.

x_0 is considered the horizontal position of the $u_{j,i}$ at the staggered grid. Therefore, $\eta_{j,i}$ is expressed as:

$$\eta_{j,i} = \eta(x_0 - \frac{\Delta x}{2}) = \eta(x_0) - \frac{\partial\eta}{\partial x}\Big|_{x_0} \frac{\Delta x}{2} + \frac{1}{2!} \frac{\partial^2\eta}{\partial x^2}\Big|_{x_0} (\frac{\Delta x}{2})^2 - \frac{1}{3!} \frac{\partial^3\eta}{\partial x^3}\Big|_{x_0} (\frac{\Delta x}{2})^3 + \dots \quad (7.2.4)$$

and for $\eta_{j+1,i}$:

$$\eta_{j+1,i} = \eta(x_0 + \frac{\Delta x}{2}) = \eta(x_0) + \frac{\partial\eta}{\partial x}\Big|_{x_0} \frac{\Delta x}{2} + \frac{1}{2!} \frac{\partial^2\eta}{\partial x^2}\Big|_{x_0} (\frac{\Delta x}{2})^2 + \frac{1}{3!} \frac{\partial^3\eta}{\partial x^3}\Big|_{x_0} (\frac{\Delta x}{2})^3 + \dots \quad (7.2.5)$$

If Equation(7.2.4) is subtracted from (7.2.5) the following expression is obtained:

$$\eta_{j+1,i} - \eta_{j,i} = + \frac{\partial\eta}{\partial x}\Big|_{x_0} \Delta x + \frac{2}{3!} \frac{\partial^3\eta}{\partial x^3}\Big|_{x_0} (\frac{\Delta x}{2})^3 + \dots \quad (7.2.6)$$

Therefore, $+\partial\eta/\partial x$ at x_0 position is estimated as:

$$+\frac{\partial\eta}{\partial x} \approx \frac{\eta_{j+1,i} - \eta_{j,i}}{\Delta x} + \frac{1}{24} \frac{\partial^3\eta}{\partial x^3} \Big|_{x_0} \Delta x^2 = \frac{\eta_{j+1,i} - \eta_{j,i}}{\Delta x} + \mathcal{O}(\Delta x^2) \quad (7.2.7)$$

Which indicates that the error, if the rest of terms of the Taylor series is omitted, is of second order. So finally, at $u_{j,i}$, $\partial\eta/\partial x$ can be estimated with a second order error as:

$$\frac{\partial\eta}{\partial x} \approx \frac{\eta_{j+1,i} - \eta_{j,i}}{\Delta x}$$

It can be seen that the staggered grid yields to a second order error at $u_{j,i}$ for the finite differences computed gradients of η . the same procedure can be followed for $\partial\eta/\partial y$ at the $v_{j,i}$ positions.

7.3 Advection Terms computation

After explaining how the domain is going to be discretized and how this affects to the order of accuracy, it is mandatory to talk about advective terms in the *Shallow Water* equations and how they are computed.

Two different schemes are followed in order to compute the different terms. Momentum equation advective's terms are evaluated using Finite Volume Method (*FVM*) and the Finite Differences Method (*FDM*) while for dynamic pressure terms Finite Differences Method (*FDM*) is used.

The Finite Volume Method is applied in CFD applications to determine partial differential equations. It consist on the use of the divergence theorem to convert volume integrals (which contain advection flux terms) into surface integrals and then, these terms are evaluated as fluxes at the walls of the control volume. Using this method the conservative property of equations is ensured and a flux limiter such as Total Variation Dismissing (*TVD*) can be applied.

Advective terms find in the *Shallow Water* momentum equations can be split in two terms, this leads to expression where *FVM* and *FDM* can be applied:

$$\begin{aligned} u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} &= \underbrace{\frac{\partial(uu)}{\partial x}}_{P_{1u}} + \underbrace{\frac{\partial(uv)}{\partial y}}_{P_{2u}} - u \left(\underbrace{\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}}_{\text{Flux}} \right) \\ u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} &= \underbrace{\frac{\partial(uv)}{\partial x}}_{P_{1v}} + \underbrace{\frac{\partial(vv)}{\partial y}}_{P_{2v}} - v \left(\underbrace{\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}}_{\text{Flux}} \right) \end{aligned}$$

The continuity equation is evaluated using the same *FVM* method.

$$\nabla \cdot (h \vec{u}) = \underbrace{\frac{\partial(hu)}{\partial x} + \frac{\partial(hv)}{\partial y}}_{P_h}$$

On the other hand, dynamic pressure terms is evaluated using *FDM* method, this is:

$$P_{pu} = -g \frac{\partial \eta}{\partial x} = -\frac{\eta_{j+1,i} - \eta_{j,i}}{\Delta x} \quad (7.3.1)$$

$$P_{pv} = -g \frac{\partial \eta}{\partial y} = -\frac{\eta_{j,i+1} - \eta_{j,i}}{\Delta y} \quad (7.3.2)$$

Derivatives approximated by using Finite Difference Method (*FDM*) are second order. It is demonstrated using the Taylor series approximation. (see *Section (7.2)*)

It is also important to notice that equations are discretized for a regular domain with constant cell size, if the model is adapted to planetary coordinates equations must be rewritten.

P1 for x-axis momentum equation

P_1 terms are computed applying *FVM* to the following expression related to component x of the momentum equation:

$$P_1 = \nabla \cdot (u\vec{u})$$

In order to do it, the control volume is assumed to be small enough to use the average value of $\nabla \cdot (u\vec{u})$, so P_1 is integrated over the fluid's particle volume, then, the divergence theorem is applied obtaining a surface integral:

$$\frac{1}{\Omega} \int_{\Omega} \nabla \cdot (u\vec{u}) d\Omega = \frac{1}{\Omega} \int_S u\vec{u} \cdot d\vec{S}$$

Surface integral is evaluated at each frontier of the finite volume.

$$\int_S u\vec{u} \cdot d\vec{S} = \frac{1}{\Omega} (u_e F_e - u_w F_w + u_n F_n - u_s F_s)$$

Where Ω is the volume of the control volume and F is the flux quantity that goes through a boundary surface of the control volume ($F = \vec{u} \cdot d\vec{S}$). Negative terms indicate that the fluid fluxes out of the control volume, while a positive sign means that flux goes into the control volume. Since velocities are not z dependent and considering that volume elements are very small, it can be shown that volume elements can be treated as two-dimensional, and therefore the flux is limited through Δx or Δy dimensions. So far example $F_e/\Omega = u/\Delta x$

Multiplying this expression by Δt , $\Delta t F$ results into the Courant number (C) and the P_{1u}

term is defined.

$$\tilde{P}_{1u} = -\Delta t \nabla \cdot (u \vec{u}) = \mathcal{C}_w u_w - \mathcal{C}_e u_e + \mathcal{C}_s u_s - \mathcal{C}_n u_n \quad (7.3.3)$$

The negative sign of Equation(7.3.3) appears as the advective term P_1 passes to the right-hand side of Equation(6.1.23). Obtaining the following expression (leaving aside the Coriolis force).

$$\frac{\partial u}{\partial t} = -g \frac{\partial n}{\partial x} - \left(\frac{\partial(uu)}{\partial x} + \frac{\partial(uv)}{\partial y} \right) + \left(u \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) \right) \quad (7.3.4)$$

The same happens to the rest of the components (v and η)

Equation (7.3.3) can be interpreted in the following way: Figure (11), centred at $u_{i,j}$ is a modification of Figure(8), where the left and right sides are re-labeled as West (w) and East (e) respectively, and the upper and lower boundaries of the control volume as North (n) and south (s) respectively.

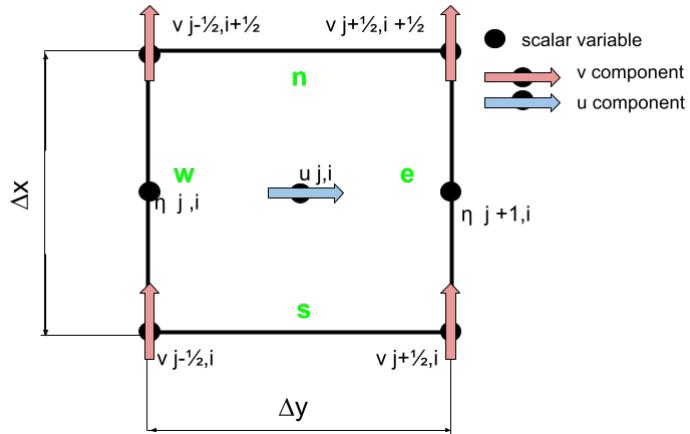


FIGURE 11: Control Volume centered at u velocity

In Equation (7.3.3) $C_w u_w$ is the flux of u velocity through the west side of the control volume multiplied by $-\Delta t$, etc.

Flow through a surface can go in the same direction of the axis or in the opposite one, for this reason quantities in equation (7.3.3) must be split depending on its direction relative to the axis. If the flux goes in the same direction of axis the quantity is positive otherwise it is negative. It may seem obvious that, if the flux is positive all the quantities should remain

zero except positive ones, etc.

$$\tilde{P}_{1u} = -\Delta t \nabla \cdot (u \vec{u}) = \mathcal{C}_w^+ u_w^+ + \mathcal{C}_w^- u_w^- - \mathcal{C}_e^+ u_e^+ - \mathcal{C}_e^- u_e^- + \mathcal{C}_s^+ u_s^+ + \mathcal{C}_s^- u_s^- - \mathcal{C}_n^+ u_n^+ - \mathcal{C}_n^- u_n^- \quad (7.3.5)$$

Due to the staggered mesh, variables are not positioned at the cell's centre, as it is explained in section (7.1), so an interpolation in velocities components is needed. This leads to:

$$\left\{ \begin{array}{l} \mathcal{C}_w^+ = \frac{1}{2}(\frac{1}{2}(u_{j-1,i} + |u_{j-1,i}|) + \frac{1}{2}(u_{j,i} + |u_{j,i}|)) \frac{\Delta t}{\Delta x} \\ \mathcal{C}_w^- = \frac{1}{2}(\frac{1}{2}(u_{j-1,i} - |u_{j-1,i}|) + \frac{1}{2}(u_{j,i} - |u_{j,i}|)) \frac{\Delta t}{\Delta x} \\ \mathcal{C}_e^+ = \frac{1}{2}(\frac{1}{2}(u_{j,i} + |u_{j,i}|) + \frac{1}{2}(u_{j+1,i} + |u_{j+1,i}|)) \frac{\Delta t}{\Delta x} \\ \mathcal{C}_e^- = \frac{1}{2}(\frac{1}{2}(u_{j,i} - |u_{j,i}|) + \frac{1}{2}(u_{j+1,i} - |u_{j+1,i}|)) \frac{\Delta t}{\Delta x} \\ \mathcal{C}_s^+ = \frac{1}{2}(\frac{1}{2}(v_{j,i-1} + |v_{j,i-1}|) + \frac{1}{2}(v_{j+1,i-1} + |v_{j+1,i-1}|)) \frac{\Delta t}{\Delta y} \\ \mathcal{C}_s^- = \frac{1}{2}(\frac{1}{2}(v_{j,i-1} - |v_{j,i-1}|) + \frac{1}{2}(v_{j+1,i-1} - |v_{j+1,i-1}|)) \frac{\Delta t}{\Delta y} \\ \mathcal{C}_n^+ = \frac{1}{2}(\frac{1}{2}(v_{j,i} + |v_{j,i}|) + \frac{1}{2}(v_{j+1,i} + |v_{j+1,i}|)) \frac{\Delta t}{\Delta y} \\ \mathcal{C}_n^- = \frac{1}{2}(\frac{1}{2}(v_{j,i} - |v_{j,i}|) + \frac{1}{2}(v_{j+1,i} - |v_{j+1,i}|)) \frac{\Delta t}{\Delta y} \end{array} \right. \quad (7.3.6)$$

The terms $1/\Delta x$ and $1/\Delta y$ in Equation(7.3.6) come from the fact that for 2D control volumes $\Omega = \Delta x \Delta y$, in addition the flux through each side is Δx or Δy , depending on the whether the velocity is v or u respectively. For instance, the average flux trough the West side of any control volume in the positive x direction times Δy is:

$$\frac{u_w^+ \Delta y}{\Delta x \Delta y} \Delta t = \frac{1}{2} \left[\frac{1}{2}(u_{j-1,i} + |u_{j-1,i}|) + \frac{1}{2}(u_{j,i} + |u_{j,i}|) \right] \frac{\Delta t}{\Delta x} = C_w^+$$

The absolute value of the velocity is the way to control the sign of velocities, if the flow goes in the negative direction on the x-axis through (for example) west face of the control volume, then the term \mathcal{C}_w^+ is zero.

Once the Courant numbers are computed, fluid velocity u and v must be evaluated at each face of the volume domain (ex. east face) using the following interpolation:

$$u_e = u_{j,i} + \frac{1}{2} \Psi(r)(u_{j+1,i} - u_{j,i})$$

where r is the up-wind down-wind gradient and Ψ is the flux limiter function (see-Section

(7.5)-), and ϕ is the u or v velocities

$$r = \frac{\phi_{j,i} - \phi_{j-1,i}}{\phi_{j+1,i} - \phi_{j,i}}$$

Since flow is divided into the positive and negative component, lasts expression must be rewritten using the Courant number to establish flow direction.

$$\begin{cases} \phi_e^+ = \phi_{j,i} + \frac{1}{2}\Psi(r^+)(1 - C_e^+)(\phi_{j+1,i} - \phi_{j,i}) \\ \phi_e^- = \phi_{j+1,i} - \frac{1}{2}\Psi(r^-)(1 + C_e^-)(\phi_{j+1,i} - \phi_{j,i}) \end{cases} \quad (7.3.7)$$

$$r^+ = \frac{\phi_{j,i} - \phi_{j-1,i}}{\phi_{j+1,i} - \phi_{j,i}} \quad (7.3.8)$$

$$r^- = \frac{\phi_{j+2,i} - \phi_{j+1,i}}{\phi_{j+1,i} - \phi_{j,i}} \quad (7.3.9)$$

Using the interpolation shown in equation (7.3.7) the u components at each face of volume domain are defined:

$$\begin{cases} u_w^+ = u_{j-1,i} + \frac{1}{2}\Psi(r_w^+)(1 - C_w^+)(u_{j,i} - u_{j-1,i}) \\ u_w^- = u_{j,i} - \frac{1}{2}\Psi(r_w^-)(1 + C_w^-)(u_{j,i} - u_{j-1,i}) \\ u_e^+ = u_{j,i} + \frac{1}{2}\Psi(r_e^+)(1 - C_e^+)(u_{j+1,i} - u_{j,i}) \\ u_e^- = u_{j+1,i} - \frac{1}{2}\Psi(r_e^-)(1 + C_e^-)(u_{j+1,i} - u_{j,i}) \\ u_s^+ = u_{j,i-1} + \frac{1}{2}\Psi(r_s^+)(1 - C_s^+)(u_{j,i} - u_{j,i-1}) \\ u_s^- = u_{j,i} - \frac{1}{2}\Psi(r_s^-)(1 + C_s^-)(u_{j,i} - u_{j,i-1}) \\ u_n^+ = u_{j,i} + \frac{1}{2}\Psi(r_n^+)(1 - C_n^+)(u_{j,i+1} - u_{j,i}) \\ u_n^- = u_{j,i+1} - \frac{1}{2}\Psi(r_n^-)(1 + C_n^-)(u_{j,i+1} - u_{j,i}) \end{cases} \quad (7.3.10)$$

$$\left\{ \begin{array}{l} r_w^+ = \frac{u_{j-1,i}-u_{j-2,i}}{u_{j,i}-u_{j-1,i}} \\ r_w^- = \frac{u_{j+1,i}-u_{j,i}}{u_{j,i}-u_{j-1,i}} \\ r_e^+ = \frac{u_{j,i}-u_{j-1,i}}{u_{j+1,i}-u_{j,i}} \\ r_e^- = \frac{u_{j+2,i}-u_{j+1,i}}{u_{j+1,i}-u_{j,i}} \\ r_s^+ = \frac{u_{j,i-1}-u_{j,i-2}}{u_{j,i}-u_{j,i-1}} \\ r_s^- = \frac{u_{j,i+1}-u_{j,i}}{u_{j,i}-u_{j,i-1}} \\ r_n^+ = \frac{u_{j,i}-u_{j,i-1}}{u_{j,i+1}-u_{j,i}} \\ r_n^- = \frac{u_{j,i+2}-u_{j,i+1}}{u_{j,i+1}-u_{j,i}} \end{array} \right. \quad (7.3.11)$$

P1 for y-axis momentum equation

Looking at the y-axis component in the momentum equation (6.1.23), the P_{1v} is defined as:

$$P_{1v} = \nabla \cdot (v \vec{u}) \quad (7.3.12)$$

The same procedure of last section is followed in order to obtain the computational expression for \tilde{P}_{1v} . This is:

$$\tilde{P}_{1v} = -\Delta t \nabla \cdot (v \vec{u}) = \mathcal{C}_w v_w - \mathcal{C}_e v_e + \mathcal{C}_s v_s - \mathcal{C}_n v_n$$

Splitting each velocity component, a similar expression is obtained discretized according to the flux direction.

$$\tilde{P}_{1v} = \mathcal{C}_w^+ v_w^+ + \mathcal{C}_w^- v_w^- - \mathcal{C}_e^+ v_e^+ - \mathcal{C}_e^- v_e^- + \mathcal{C}_s^+ v_s^+ + \mathcal{C}_s^- v_s^- - \mathcal{C}_n^+ v_n^+ - \mathcal{C}_n^- v_n^- \quad (7.3.13)$$

$$\left\{
 \begin{array}{l}
 C_w^+ = \frac{1}{2}(\frac{1}{2}(u_{j-1,i+1} + |u_{j-1,i+1}|) + \frac{1}{2}(u_{j-1,i} + |u_{j-1,i}|)) \frac{\Delta t}{\Delta x} \\
 C_w^- = \frac{1}{2}(\frac{1}{2}(u_{j-1,i+1} - |u_{j-1,i+1}|) + \frac{1}{2}(u_{j-1,i} - |u_{j-1,i}|)) \frac{\Delta t}{\Delta x} \\
 C_e^+ = \frac{1}{2}(\frac{1}{2}(u_{j,i+1} + |u_{j,i+1}|) + \frac{1}{2}(u_{j,i} + |u_{j,i}|)) \frac{\Delta t}{\Delta x} \\
 C_e^- = \frac{1}{2}(\frac{1}{2}(u_{j,i+1} - |u_{j,i+1}|) + \frac{1}{2}(u_{j,i} - |u_{j,i}|)) \frac{\Delta t}{\Delta x} \\
 C_s^+ = \frac{1}{2}(\frac{1}{2}(v_{j,i-1} + |v_{j,i-1}|) + \frac{1}{2}(v_{j,i} + |v_{j,i}|)) \frac{\Delta t}{\Delta y} \\
 C_s^- = \frac{1}{2}(\frac{1}{2}(v_{j,i-1} - |v_{j,i-1}|) + \frac{1}{2}(v_{j,i} - |v_{j,i}|)) \frac{\Delta t}{\Delta y} \\
 C_n^+ = \frac{1}{2}(\frac{1}{2}(v_{j,i} + |v_{j,i}|) + \frac{1}{2}(v_{j,i+1} + |v_{j,i+1}|)) \frac{\Delta t}{\Delta y} \\
 C_n^- = \frac{1}{2}(\frac{1}{2}(v_{j,i} - |v_{j,i}|) + \frac{1}{2}(v_{j,i+1} - |v_{j,i+1}|)) \frac{\Delta t}{\Delta y}
 \end{array}
 \right.$$

$$\left\{
 \begin{array}{l}
 v_w^+ = v_{j-1,i} + \frac{1}{2}\Psi(r_w^+) (1 - C_w^+) (v_{j,i} - v_{j-1,i}) \\
 v_w^- = v_{j,i} - \frac{1}{2}\Psi(r_w^-) (1 + C_w^-) (v_{j,i} - v_{j-1,i}) \\
 v_e^+ = v_{j,i} + \frac{1}{2}\Psi(r_e^+) (1 - C_e^+) (v_{j+1,i} - v_{j,i}) \\
 v_e^- = v_{j+1,i} - \frac{1}{2}\Psi(r_e^-) (1 + C_e^-) (v_{j+1,i} - v_{j,i}) \\
 v_s^+ = v_{j,i-1} + \frac{1}{2}\Psi(r_s^+) (1 - C_s^+) (v_{j,i} - v_{j,i-1}) \\
 v_s^- = v_{j,i} - \frac{1}{2}\Psi(r_s^-) (1 + C_s^-) (v_{j,i} - v_{j,i-1}) \\
 v_n^+ = v_{j,i} + \frac{1}{2}\Psi(r_n^+) (1 - C_n^+) (v_{j,i+1} - v_{j,i}) \\
 v_n^- = v_{j,i+1} - \frac{1}{2}\Psi(r_n^-) (1 + C_n^-) (v_{j,i+1} - v_{j,i})
 \end{array}
 \right.
 \quad (7.3.14)$$

$$\left\{
 \begin{array}{l}
 r_w^+ = \frac{v_{j-1,i} - v_{j-2,i}}{v_{j,i} - v_{j-1,i}} \\
 r_w^- = \frac{v_{j+1,i} - v_{j,i}}{v_{j,i} - v_{j-1,i}} \\
 r_e^+ = \frac{v_{j,i} - v_{j-1,i}}{v_{j+1,i} - v_{j,i}} \\
 r_e^- = \frac{v_{j+2,i} - v_{j+1,i}}{v_{j+1,i} - v_{j,i}} \\
 r_s^+ = \frac{v_{j,i-1} - v_{j,i-2}}{v_{j,i} - v_{j,i-1}} \\
 r_s^- = \frac{v_{j,i+1} - v_{j,i}}{v_{j,i} - v_{j,i-1}} \\
 r_n^+ = \frac{v_{j,i} - v_{j,i-1}}{v_{j,i+1} - v_{j,i}} \\
 r_n^- = \frac{v_{j,i+2} - v_{j,i+1}}{v_{j,i+1} - v_{j,i}}
 \end{array}
 \right.
 \quad (7.3.15)$$

As said before, Ψ function is a flux limiter. Different types of flux limiters and its advantages are explained deeply at the *Section (7.5)*. This terms is very important since determines the stability of the solution, some schemes are more stable than others, also the accuracy of the solution depends on this flux function.

Ph for the continuity equation

Finite Volume Method (*FVM*) is also applied at the continuity equation computation. In order to do so, the same procedure from P_1 is followed.

Firstly, the divergence theorem averaging $\nabla \cdot (h\vec{u})$ at the control volume, will be applied and F coefficients are obtained for each flow direction. After that, the whole term is multiplied by $-\Delta t$ obtaining the computational expression \tilde{P}_h .

$$P_h = \nabla \cdot (h\vec{u}) = \frac{1}{\Omega} \int_{\Omega} \nabla \cdot (h\vec{u}) = \frac{1}{\Omega} (h_e F_e - h_w F_w + h_n F_n - h_s F_s)$$

$$\tilde{P}_h = -\Delta t \nabla \cdot (h\vec{u}) = C_w^+ h_w^+ + C_w^- h_w^- - C_e^+ h_e^+ - C_e^- h_e^- + C_s^+ h_s^+ + C_s^- h_s^- - C_n^+ h_n^+ - C_n^- h_n^- \quad (7.3.16)$$

Into the equation (7.3.16), variable h is already split. In addition, the staggered mesh has to be taken into account; h and η are computed at the cell's centre while velocity components are moved to the cell's boundary. This fact simplifies the way on how Courant number is computed.

$$\left\{ \begin{array}{l} C_w^+ = \frac{1}{2} (u_{j-1,i} + |u_{j-1,i}|) \frac{\Delta t}{\Delta x} \\ C_w^- = \frac{1}{2} (u_{j-1,i} - |u_{j-1,i}|) \frac{\Delta t}{\Delta x} \\ C_e^+ = \frac{1}{2} (u_{j,i} + |u_{j,i}|) \frac{\Delta t}{\Delta x} \\ C_e^- = \frac{1}{2} (u_{j,i} - |u_{j,i}|) \frac{\Delta t}{\Delta x} \\ C_s^+ = \frac{1}{2} (u_{j,i-1} + |u_{j,i-1}|) \frac{\Delta t}{\Delta y} \\ C_s^- = \frac{1}{2} (u_{j,i-1} - |u_{j,i-1}|) \frac{\Delta t}{\Delta y} \\ C_n^+ = \frac{1}{2} (u_{j,i} + |u_{j,i}|) \frac{\Delta t}{\Delta y} \\ C_n^- = \frac{1}{2} (u_{j,i} - |u_{j,i}|) \frac{\Delta t}{\Delta y} \end{array} \right.$$

Once the Courant number is computed, h is expressed in the same way as the velocity is interpolated in the P_1 advection term.

$$\left\{ \begin{array}{l} h_w^+ = h_{j-1,i} + \frac{1}{2}\Psi(r_w^+) (1 - C_w^+) (h_{j,i} - h_{j-1,i}) \\ h_w^- = h_{j,i} - \frac{1}{2}\Psi(r_w^-) (1 + C_w^-) (h_{j,i} - h_{j-1,i}) \\ h_e^+ = h_{j,i} + \frac{1}{2}\Psi(r_e^+) (1 - C_e^+) (h_{j+1,i} - h_{j,i}) \\ h_e^- = h_{j+1,i} - \frac{1}{2}\Psi(r_e^-) (1 + C_e^-) (h_{j+1,i} - h_{j,i}) \\ h_s^+ = h_{j,i-1} + \frac{1}{2}\Psi(r_s^+) (1 - C_s^+) (h_{j,i} - h_{j,i-1}) \\ h_s^- = h_{j,i} - \frac{1}{2}\Psi(r_s^-) (1 + C_s^-) (h_{j,i} - h_{j,i-1}) \\ h_n^+ = h_{j,i} + \frac{1}{2}\Psi(r_n^+) (1 - C_n^+) (h_{j,i+1} - h_{j,i}) \\ h_n^- = h_{j,i+1} - \frac{1}{2}\Psi(r_n^-) (1 + C_n^-) (h_{j,i+1} - h_{j,i}) \end{array} \right. \quad (7.3.17)$$

$$\left\{ \begin{array}{l} r_w^+ = \frac{h_{j-1,i} - h_{j-2,i}}{h_{j,i} - h_{j-1,i}} \\ r_w^- = \frac{h_{j+1,i} - h_{j,i}}{h_{j,i} - h_{j-1,i}} \\ r_e^+ = \frac{h_{j,i} - h_{j-1,i}}{h_{j+1,i} - h_{j,i}} \\ r_e^- = \frac{h_{j+2,i} - h_{j+1,i}}{h_{j+1,i} - h_{j,i}} \\ r_s^+ = \frac{h_{j,i-1} - h_{j,i-2}}{h_{j,i} - h_{j,i-1}} \\ r_s^- = \frac{h_{j,i+1} - h_{j,i}}{h_{j,i} - h_{j,i-1}} \\ r_n^+ = \frac{h_{j,i} - h_{j,i-1}}{h_{j,i+1} - h_{j,i}} \\ r_n^- = \frac{h_{j,i+2} - h_{j,i+1}}{h_{j,i+1} - h_{j,i}} \end{array} \right. \quad (7.3.18)$$

With all of this, \tilde{P}_h is computed.

Finally, P_2 is computed.

P2 for the x-axis momentum equation

P_2 term is easier to compute than the terms explained above. Here, finite differences will be used. It is split depending on which axis components is being computed in the same way P_1 is divided (P_{2u} and P_{2v}). Analytical expression for this advection term is:

$$P_{2u} = u \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right)$$



Numerical expression can be obtained by finite differences, however, the staggered mesh requires velocity interpolation depending on the component.

$$\frac{\partial u}{\partial x} = \frac{u_{j+1,i} - u_{j-1,i}}{2\Delta x}$$

But on the other hand, v velocity component interpolation results in a more complicated expression. Since velocity v is not at the same place of the u velocity, two interpolations needs to be done. Firstly, the average value at the top and bottom of the control volume is done.

$$v_{top} = \frac{1}{2}(v_{j,i} + v_{j+1,i})$$
$$v_{bottom} = \frac{1}{2}(v_{j,i-1} + v_{j+1,i-1})$$

Once those values are computed, velocity derivative is expressed as:

$$\frac{\partial v}{\partial y} = \frac{v_{top} - v_{bottom}}{\Delta y}$$

Which is similar to an averaging of the v velocity.

Grouping the above equations together the following term is obtained:

$$\tilde{P}_{2u} = u_{j,i} \left(\frac{u_{j+1,i} - u_{j-1,i}}{2\Delta x} + \frac{(v_{j,i} + v_{j+1,i}) - (v_{j,i-1} + v_{j+1,i-1})}{2\Delta y} \right)$$

P2 for the y-axis momentum equation

Analytical expression for this terms is:

$$P_{2v} = v \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right)$$

In order to compute y-axis component of the P_2 term, the same procedure is followed. Nevertheless in this case, the u component is not placed in the same position as the v component so the most complicated interpolation is done for u velocity.

$$\frac{\partial v}{\partial y} = \frac{v_{j,i+1} - v_{j,i-1}}{2\Delta y}$$



$$u_{top} = \frac{1}{2}(u_{j,i} + u_{j,i+1})$$
$$u_{bottom} = \frac{1}{2}(u_{j-1,i+1} + u_{j-1,i})$$

where velocity derivative is expressed as:

$$\frac{\partial u}{\partial x} = \frac{u_{top} - u_{bottom}}{\Delta x}$$

Taking this interpolation, \tilde{P}_{2y} terms is defined.

$$P_{2v} = v_{j,i} \left(\frac{(u_{j,i} + u_{j,i+1}) - (u_{j-1,i+1} + u_{j-1,i})}{2\Delta x} + \frac{v_{j,i+1} - v_{j,i-1}}{2\Delta y} \right) \quad (7.3.19)$$

7.4 The Courant Number

Convergence in equations is achieved if the difference between exact and computational solution tends to zero when the number of points in the mesh increases and the time step decreases. However, this convergence is not mathematically trivial and the knowledge needed to prove converge in the discretised advective terms is out of the scope of this text[15].

In *CFD* problems there is a condition that must be satisfied in order to obtain the convergence, this is the Courant-Friedrichs-Lowy number, discovered by Richard Courant, Kurt Friedrichs and Hans Lewy in 1928 [1]. It is usually used in advection schemes without diffusive terms. This Courant condition always must meet: $\mathcal{C} \leq 1$ and it's expressed as:

$$\mathcal{C} = \frac{U\Delta t}{\Delta x} \quad (7.4.1)$$

Where U is the characteristic velocity in the mesh, Δt is the time step and Δx is the spatial increment. This expression states the maximum Δt allowed int the *SW* model for a given values of Δx and U . Nevertheless, the code implementing the numerical model can monitor the Courant number and fix a maximum to ensure security. For instance the Courant number can be fixed to $\mathcal{C} = 0.5$.

7.5 Flux Limiter function

Flux trough the domain may present strong variations in its magnitude. These variations are expressed mathematically as gradients which are called "fronts". In order to capture fronts presented in the mesh with size ($L \times L$), the following condition must be fulfilled:

$$\begin{cases} \Delta y \ll L \\ \Delta x \ll L \end{cases} \quad (7.5.1)$$

Two options can be adopted: the first one consist on increasing mesh size and vary it where the fronts appear. This will ensure that constraint (7.5.1) is fulfilled. The second one is to keep cell size as a constant and capture high gradients (fronts) using a variable numerical scheme capable of adapt to local solution. Those methods are called Total Variation Dismissing (*TVD*) schemes which are used in section (7.3) to compute advection terms.

Upwind schemes are used in computational physics to solve hyperbolic equations as advection terms. Its name comes from the fact that for a function u advected by a positive velocity c in the positive x direction, 'downwind' satisfies the linear advection equation.

$$\frac{\partial u}{\partial t} + c \frac{\partial u}{\partial x} = 0 \quad (7.5.2)$$

Which is numerically solved for u_j using the immediate u_{j-1} 'upwind' value for the lowest order scheme or several 'upwind' values, u_{j-1}, u_{j-2}, \dots , if the scheme is not superior order.

Some of basic upwind schemes characteristics are its first order of accuracy and the monotonicity of the function, this means that artificial extrema are avoided. Despite of its advantages, it rapidly smears out strong variations (fronts). Higher order upwind schemes support fronts although some oscillations called wiggles appear close to the fronts [10]. Figure(12).

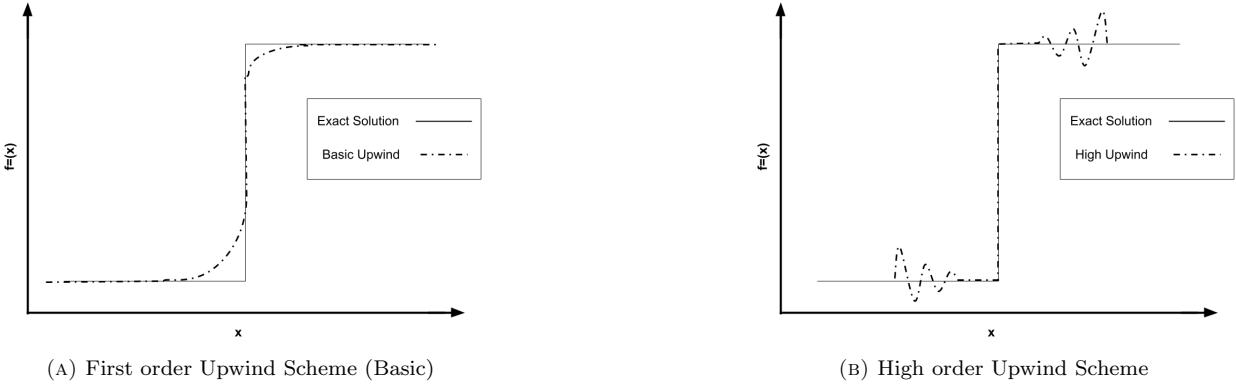


FIGURE 12: Comparison of Upwind schemes

TVD method combines both upwind schemes allowing to obtain the main advantage of each method. When high gradients appear, the scheme changes to a first order scheme (basic upwind) applying diffusion into the flux, whereas when the flux do not have oscillations, the order of accuracy increases by using higher order upwind schemes. Total Variation at time step n (TV^n) is defined as:

$$TV^n = \sum_i |\tilde{\phi}_{j+1}^n - \tilde{\phi}_j^n| \quad (7.5.3)$$

Where ϕ is a fluid's advected magnitude.

The dismissing property implies that: $TV^{n+1} < TV^n$. Notice that Equation (7.5.3) quantifies the magnitude of wiggles generated in the mesh.

Non-linear schemes can be defined combining explicit Euler methods in order to apply successfully the *TVD* method. This expression is used to obtain the velocities at the different faces of the control volume. (see for example Equation(7.3.10))

$$\phi_e = \phi_{j,i}^L + \frac{1}{2}\Psi(r)(\phi_{j+1,i}^H - \phi_{j,i}^L) \quad (7.5.4)$$

Where $\phi_{j,i}$ is the flux transported variable (like u or v), and ϕ^L , ϕ^H denote a low order or high order upwind advector variable.

The Ψ function is the flux limiter which is responsible for the local control flux solution. If local extrema are present in the solution, this function tends to zero while when the solution is smooth, flux limiter grows to 1 to increase order of precision.

The r parameter, which is called upwind/downwind gradient is very important in order to define flux limiter value.

$$r = \frac{\phi_{j,i} - \phi_{j-1,i}}{\phi_{j+1,i} - \phi_{j,i}}$$

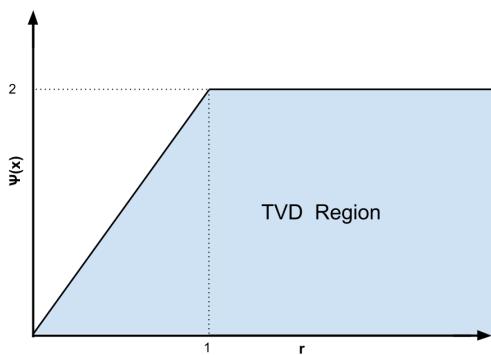
If r_j is close or equal to one, $\tilde{\phi}$ has a linear response while for $r_j < 0$, a local extrema appears. So depending on the value of r , Ψ will change its value $\rightarrow \Psi(r)$. This fact allows to establish the *TVD* condition whose derivation may be obtained from (*Cushman-Roisin and Beckers (2011)*-[6]):

$$0 \leq \mathcal{C} + \frac{\mathcal{C}(1-\mathcal{C})}{2} \left[\frac{\Psi_j}{r} - \Psi_{j-1} \right] \leq 1 \quad (7.5.5)$$

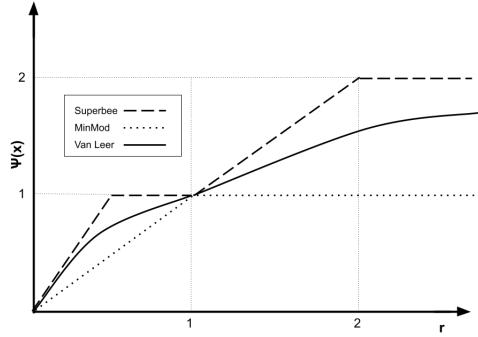
Where \mathcal{C} is the *Courant* number (see Section (7.4)).

Looking at the most critical values for ϕ at points (j and j-1) the *TVD* condition (7.5.5) is ensured whenever the following conditions are fulfilled.

$$\Psi \leq 2 \quad \text{and} \quad \frac{\Psi}{r} \leq 2 \quad (7.5.6)$$



(A) Flux Limiter. TVD Zone allowed



(B) 2nd order TVD flux limiter schemes

FIGURE 13: Flux Limiter function $\Psi(r)$

Figure(13a) plots $\Psi(r)$ and the region where the *TVD* region is fulfilled. Figure(13b) plots the limiter values for the different *TVD* methods.

Once Flux limiter function region is defined where the *TVD* property is ensured (as seen in Figure (13a)), some schemes can be presented:

NAME	EXPRESSION	ACCURACY
Van Leer	$\Psi(r) = \frac{r+ r }{1+ r }$	2nd Order, symmetric
Van Albada	$\Psi(r) = \frac{r+r^2}{1+r^2}$	2n Order, symmetric
MINMOD	$\Psi(r) = \max [0, \min(r, 1)]$	2nd Order, symmetric
Bounded QUICK	$\Psi(r) = \max [0, \min (2r, \frac{3+4}{4}, 2)]$	Quick scheme bounded to keep in the TVD zone.
Superbee	$\Psi(r) = \max [0, \min(2r, 1), \min(r, 2)]$	2nd Order, symmetric
Koren	$\Psi(r) = \max [0, \min (2r, \frac{2r+1}{3}, 2)]$	3rd Order, non symmetric
MUSL	$\Psi(r) = \max [0, \min (2, 2r, \frac{1+r}{2})]$	2nd Order, symmetric

TABLE 1: Flux limiter Schemes. Arranged by Scheme's name, expression and accuracy

Those schemes are obtained from [2]

Each of these schemes present different advantages and disadvantages, hence its use will depend on the purpose of the calculation to be performed. In the Shallow Water model, the *Superbee* scheme is used for low to moderate resolution, whereas *MUSCL* is used for every high resolution simulations. Some properties of the *Superbee* ans *MUSCL* limiters can be seen at [2]

7.6 Time Integration

In the previous section (7.3) terms from the right side of Shallow Water equations were computed. These are: P_1 , P_2 , P_p and P_h . It is now necessary to temporarily integrate the variables: u , v , η to predict this evolution. Figure (14) shows each term of SW equations and the computation scheme used to solve it.

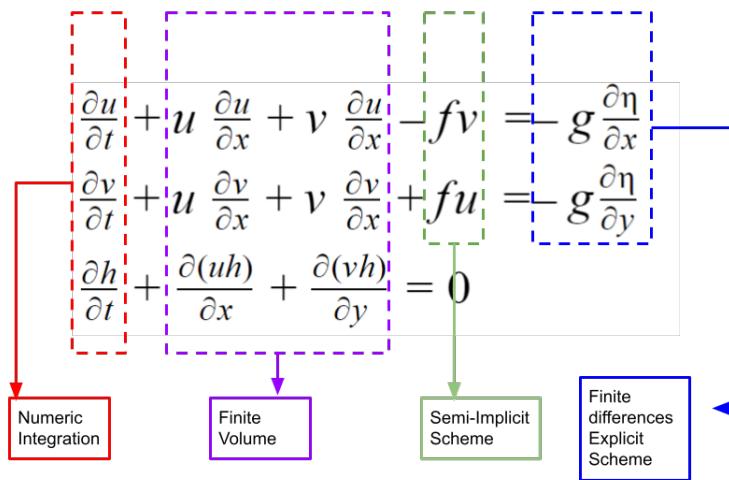


FIGURE 14: Shallow Water Equations. Integration scheme used for each term

In order to integrate expression (7.6.1), some schemes can be used: Euler, Range-Kutta, Adams Bashforth etc. Each scheme has different properties and its use depends on which physic phenomenon is being represented.

$$\frac{\partial u}{\partial t} = \frac{P_{1u}}{\Delta t} + P_{2u} + P_{pu} \quad (7.6.1)$$

It is important to notice that Coriolis parameter has been taken away from this expression because a special treatment is needed for this term after updated velocities are obtained.

Once the advection term and the pressure gradients are computed, time integration will be performed before applying Coriolis.

To integrate velocities, an Adams-Bashforth thir order scheme is used. The advantages of this scheme can be found in section (2.4) and in *Dowling et al. (1998)*.

Implementation of time integration

The Shallow Water equations (6.1.23) can be split into two different terms: time derivative terms are in the left side of the expression while the advective an other terms are in the right side.

$$\frac{\partial u}{\partial t} = f_1(u, v, \eta, x, y, t) \quad (7.6.2)$$

$$\frac{\partial v}{\partial t} = f_2(u, v, \eta, x, y, t) \quad (7.6.3)$$

$$\frac{\partial \eta}{\partial t} = f_3(u, v, \eta, x, y, t) \quad (7.6.4)$$

Where f terms, not to be mistaken with the Coriolis acceleration, are the advection and pressure gradient terms of the Shallow Water equations. i.e ($f_1 = P_{1u}/\Delta t + P_{2u} + P_{pu}$)

As it is said before, *Adams Bashforth* time integration method is used. This is:

$$\phi_{n+1} = \phi_n + \Delta t \left(\frac{23}{12} f^n - \frac{4}{3} f^{n-1} + \frac{5}{12} f^{n-2} \right) \quad (7.6.5)$$

$$\begin{cases} u^{n+1} = u^n + \Delta t \left(\frac{23}{12} f_1^n - \frac{4}{3} f_1^{n-1} + \frac{5}{12} f_1^{n-2} \right) \\ v^{n+1} = v^n + \Delta t \left(\frac{23}{12} f_2^n - \frac{4}{3} f_2^{n-1} + \frac{5}{12} f_2^{n-2} \right) \\ \eta^{n+1} = \eta^n + \Delta t \left(\frac{23}{12} f_3^n - \frac{4}{3} f_3^{n-1} + \frac{5}{12} f_3^{n-2} \right) \end{cases} \quad (7.6.6)$$

Where u^{n+1} , v^{n+1} and η^{n+1} are the time update horizontal velocities and free surface perturbation u^n , v^n , η^n are their values a time step before, and the f^n , f^{n-1} and f^{n-2} are the (7.6.2) to (7.6.4) f functions already computed three time steps before (n , $n - 1$ and $n - 2$).

Expression (7.6.6) involves f terms up to two time steps backward which leads to use the Euler method for the first two time iterations, $n = 0$ and $n = 1$.

$$\begin{aligned} u^{n+1} &= u^n + \Delta t f_1^n \\ v^{n+1} &= v^n + \Delta t f_2^n \\ \eta^{n+1} &= \eta^n + \Delta t f_3^n \end{aligned} \quad (7.6.7)$$

It is important to notice that time step (Δt) is considered to be constant and is defined at the beginning of the code. Despite of the fact that velocity (U) and space increment (Δx) change over the grid's domain its implementation turns much more difficult if Δt is not constant. Moreover, this hypothesis is well established for atmospheric phenomenon where the velocity is maintained for long periods of time, where variable time step does not have to much sense.

7.7 Coriolis Integration

Let's integrate the ordinary differential equations resulting from inertial oscillations due to rotation following the text by *Cushman-Roisin and Beckers (2011, chapter 2)*[6]. Let's also apply an explicit Euler scheme for their time integration.

$$\begin{aligned}\frac{du}{dt} - fv &= 0 \rightarrow \frac{\tilde{u}^{n+1} - \tilde{u}^n}{\Delta t} = f\tilde{v}^n \\ \frac{dv}{dt} + fu &= 0 \rightarrow \frac{\tilde{v}^{n+1} - \tilde{v}^n}{\Delta t} = f\tilde{u}^n\end{aligned}\quad (7.7.1)$$

Where n and $n + 1$ indicate the respective time steps. Now, let's compute the squared of norm of the velocity.

$$\|\vec{u}\|^2 = (\tilde{u}^n)^2 + (\tilde{v}^n)^2 = (1 + f^2 \Delta t^2)^n [(\tilde{u}^0)^2 + (\tilde{v}^0)^2] \quad (7.7.2)$$

From the norm of \vec{u} , it is deduced that its value increases with time. Kinetic energy, which is proportional to this norm will also increase indicating that this method is unstable.

On the other hand, if an implicit Euler method is used, the squared norm of the velocity at time step n can be expressed as:

$$\begin{aligned}\frac{du}{dt} - fv &= 0 \rightarrow \frac{\tilde{u}^{n+1} - \tilde{u}^n}{\Delta t} = f\tilde{v}^{n+1} \\ \frac{dv}{dt} - fu &= 0 \rightarrow \frac{\tilde{v}^{n+1} - \tilde{v}^n}{\Delta t} = f\tilde{u}^{n+1}\end{aligned}\quad (7.7.3)$$

$$\|\vec{u}\|^2 = (\tilde{u}^n)^2 + (\tilde{v}^n)^2 = (1 + f^2 \Delta t^2)^{-n} [(\tilde{u}^0)^2 + (\tilde{v}^0)^2] \quad (7.7.4)$$

In this case, energy vanishes with time so it is an energy diffusive method.

If a semi-implicit method is used such that:

$$\begin{aligned}\frac{\tilde{u}^{n+1} - \tilde{u}^n}{\Delta t} &= f [(1 - \alpha)\tilde{v}^n + \alpha\tilde{v}^{n+1}] \\ \frac{\tilde{v}^{n+1} - \tilde{v}^n}{\Delta t} &= f [(1 - \alpha)\tilde{u}^n + \alpha\tilde{u}^{n+1}]\end{aligned}\quad (7.7.5)$$

Where $0 \leq \alpha \leq 1$, Cushman-Roisin and Beckers show that

$$(\tilde{u}^n)^2 + (\tilde{v}^n)^2 = \left[\frac{1 + (1 - \alpha)^2 f^2 \Delta t^2}{1 + \alpha^2 f^2 \Delta t^2} \right] [(\tilde{u}^0)^2 + (\tilde{v}^0)^2] \quad (7.7.6)$$

If $\alpha = 1/2$ is choose, then $\|\tilde{u}^n\|^2 = \|\tilde{u}^0\|^2$ and kinetic energy is conserved.

For this reason, velocities after Coriolis are computed such as:

$$\begin{aligned}u^{n+1} &= u^n + 0.5f \left(v^n + v^{n+1} \right) \\ v^{n+1} &= v^n - 0.5f \left(u^n + u^{n+1} \alpha \right)\end{aligned}\quad (7.7.7)$$

In Equation(7.7.7) some previous aspects must be taken into account before start the computation.

First of all, velocity at the actual time, say u^n , is considered to be the velocity obtained from the Adams Bashforth's integration, this is called *predictor velocity* u^* . While u^n and v^n are the old velocities before Correcting by Coriolis.

Moreover, velocity v^n needs for an interpolation due to the staggering of the mesh. This interpolation places the velocity vector v in the same position as the vector u . See Figure(15).

$$\begin{aligned}u_{avg(j,i)}^n &= 0.25(u_{j+1,i-1}^n + u_{j+1,i}^n + u_{j,i}^n + u_{j,i-1}^n) \\ v_{avg(j,i)}^n &= 0.25(v_{j,i}^n + v_{j,i+1}^n + v_{j-1,i}^n + v_{j-1,i+1}^n)\end{aligned}\quad (7.7.8)$$

Finally, the Coriolis parameter f is computed for ellipsoidal coordinates as:

$$f_{j,i} = 2\Omega \sin(\varphi_{j,i}) + u_{j,i}^n \frac{\sin(\varphi_{j,i})}{r_z(\varphi_{j,i})} \quad (7.7.9)$$

for the $u_{j,i}^{n+1}$ equation.

$$f_{j,i} = 2\Omega \sin(\varphi_{j,i}) + u_{avg,j,i}^n \frac{\sin(\varphi_{j,i})}{r_z(\varphi_{j,i})} \quad (7.7.10)$$

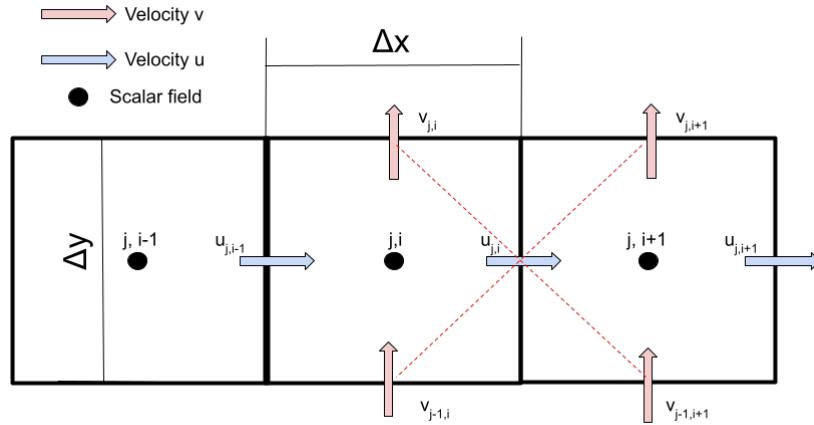


FIGURE 15: v^n velocity interpolation.

for the $v_{j,i}^{n+1}$ equation.

Applying these concepts into Equation(7.7.7) the following expressions for the u and v velocities components are obtained.

$$u_{j,i}^{n+1} = \frac{u_{j,i}^* - (\alpha f_{j,i} \Delta t)^2 u_{j,i}^n + 2\alpha f_{j,i} \Delta t v_{avg(i,i)}^n}{1 + (\alpha f_{j,i} \Delta t)^2} \quad (7.7.11)$$

$$v_{j,i}^{n+1} = \frac{v_{j,i}^* - (\alpha f_{j,i} \Delta t)^2 v_{j,i}^n - 2\alpha f_{j,i} \Delta t u_{avg(i,i)}^n}{1 + (\alpha f_{j,i} \Delta t)^2} \quad (7.7.12)$$

Equation(7.7.11) and Equation(7.7.12) give the final form of the velocity field at the time step $(n+1)$.



8 MMS Validation method

The method of Dynamic code testing is used in programming to verify that code is well implemented. There are plenty of different tests that can be done to ensure that the programme works as expected, although some of them are much harder to implement than others and requires from expert judgement.

The most common methods used for code validation are: Method of Exact solution (*MES*) and Method of Manufactured Solutions (*MMS*). These methods allows the programmer to control the order of accuracy and obtain the error rate without need of expert or previous judgements [8].

Considering the following partial differential equation (PDE):

$$\frac{D\vec{u}}{Dt} = g \quad (8.0.1)$$

where \vec{u} is the solution and g is the source term. The method of exact solutions (*MES*) chooses the function g and then, solve the equation (8.0.1) using mathematical tools as separations of variables. On the other hand, method of manufactures solutions (*MMS*) applies the same procedure backward, this is, first \vec{u} function is chosen and then total derivative (D/Dt) is applied to obtain g .

As both methods offer the same advantages (within the scope of this project), *MMS* is used since it is simpler to execute than the exact solution method. The material derivative introduces non-linear terms which makes the choice of g difficult. However, when the solution function \vec{u} is created, some rules must be followed. (see- *Code Verification by the Method of Manufactured Solutions, 2000* [8])

In the *shallow Water model*, advective terms can be validated separately as:

$$\mathcal{D}f = 0 \quad (8.0.2)$$

where \mathcal{D} is the total derivative and f is the advection term to validate. The manufactured solution is denoted by the sub-index M , so an arbitrary f_M function is chosen, and after



applying the material derivative on it, a source term F_M appears transforming (8.0.2) into:

$$\mathcal{D}f_M = F_M \quad (8.0.3)$$

It seems to be intuitive that the code must validate the following equation:

$$\mathcal{D}f - F_M = 0$$

Once numerical solution is computed the error between both functions at the whole domain is evaluated as:

$$\|f - F_M\|_\infty = \max|f - F_M| \quad (8.0.4)$$

For discretized domains, the expression (8.0.4) is proportional to a constant K and to the grid's size (h).

$$e = \|f - f_M\| = Kh^p \quad (8.0.5)$$

where p is the accuracy order of the function.

In the *Shallow Water* model validation for periodic Boundary conditions in x and y it is convenient to use a periodic function, as an example the following functions are used:

$$\begin{aligned} u_M &= \sin(2\pi x)\cos(2\pi y)t \\ v_M &= \cos(2\pi x)\sin(2\pi y)t \\ h_M &= 0.01\cos(2\pi x)\cos(2\pi y)\cos(t) + D \end{aligned} \quad (8.0.6)$$

t is considered to be a constant different from zero set to any time instant, and D must follow the aspect ratio hypothesis: $D \ll L$.

The validation process consists on the evaluation of advection terms at each mesh point, always taking into account staggering. In order to simplify the analysis, domain is represented as a square with constant space discretization (Δx and Δy). This process should give a second order accuracy for the advection terms while for the time integration is third order.

In the following pages, each term of the *Shallow Water* equations is described, although it is important to notice that even though the validation is only described for the u , it is done in the same way for the v and η .

Advection Terms validation. P1

As it was developed in section (7.3) the P_1 term is expressed as:

$$P_{1u} = \nabla \cdot (u\vec{u}) = \frac{\partial(uu)}{\partial x} + \frac{\partial(uv)}{\partial y} \quad (8.0.7)$$

$$P_{1v} = \nabla \cdot (v\vec{u}) = \frac{\partial(vu)}{\partial x} + \frac{\partial(vv)}{\partial y} \quad (8.0.8)$$

On the other hand, the computational method resolve the part \tilde{P}_{1u} of the advection terms as:

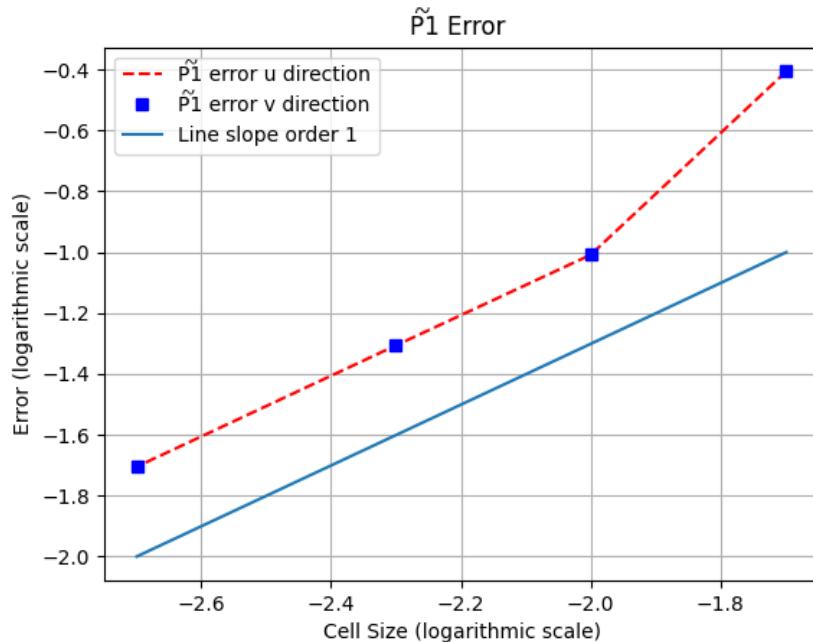
$$\tilde{P}_{1u} = -\Delta t(\nabla \cdot (u\vec{u}))$$

hence error is expressed such as: $e = \|P_{1u} + \tilde{P}_{1u}/\Delta t\|_\infty$.

Applying the *MMS* method to the P_1 advection, the error obtained is reduced as the number of mesh points increase. Moreover, since the Courant number condition must always be fulfilled, variables Δt and Δx can not be arbitrary values. This concept is explained in more detail at section (7.4). Simulation parameters for the P_1 term are:

- $\Delta x = [50, 100, 200, 500]$
- $\Delta t = 10^{-6}$

It may seem strange that the order of accuracy is less than the predicted by the theory but it is due to the fact that the *TVD* scheme switches between a first and second order scheme (see “*High Resolution Schemes using Flux Limiters for Hyperbolic Conservation Laws*”, 1984 [12]). The same behaviour is found at previous studies [14] and [2]. Taking these studies as reference, results can be considered accurate enough.


 FIGURE 16: \tilde{P}_{1u} and \tilde{P}_{1v} advection error. Constant Δt

Advection Terms validation. P2

The same procedure explained in the $P1$ section is followed to validate the $P2$ term. It is expressed as:

$$P_{2u} = u \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) \quad (8.0.9)$$

$$P_{2v} = v \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) \quad (8.0.10)$$

These expressions are similar to the numerical ones so the error is expressed as:

$$e = \|P_{2u} - \tilde{P}_{2u}\|_\infty.$$

Order of accuracy for the P_2 is second order, as theory predicts. Simulation parameters are:

- $\Delta x = [50, 100, 200, 500]$

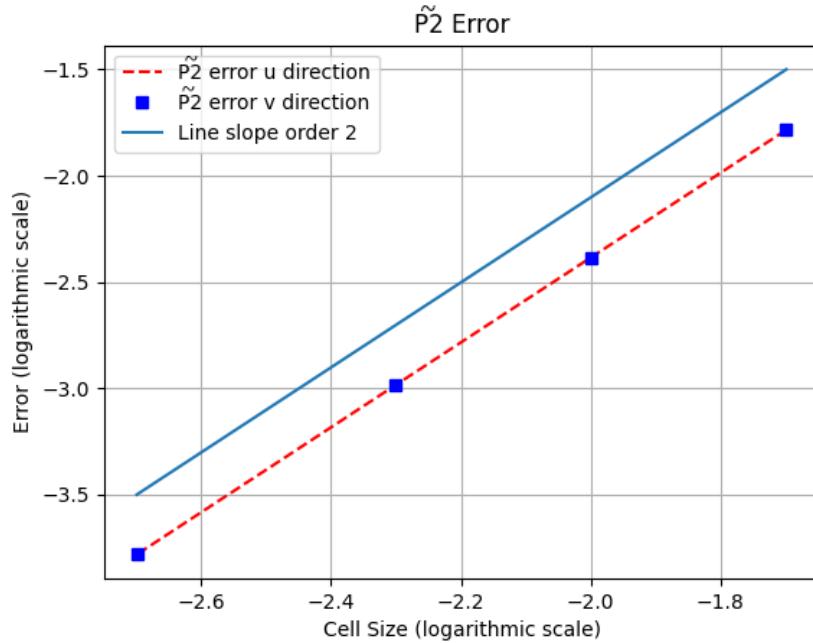


FIGURE 17: \tilde{P}_{2u} and \tilde{P}_{2v} advection error. Constant Δt

Advection Terms validation. Ph, Pp

Continuity equation term P_h and the pressure contribution P_p are also validated in the same way. In this case, theory predicts an error of second order.

$$P_h = \nabla \cdot (h\vec{u}) = \frac{\partial hu}{\partial x} + \frac{\partial hv}{\partial y} \quad (8.0.11)$$

$$P_{pu} = -g \frac{\partial \eta}{\partial x} \quad (8.0.12)$$

$$P_{pv} = -g \frac{\partial \eta}{\partial y} \quad (8.0.13)$$

In the same way as in the $P1$ case, the computational term is affected by the Δt variable, leading to:

$$\tilde{P}_h = \Delta t (\nabla \cdot (h\vec{u}))$$

On the other hand, P_p follows the same tendency as the P_2 term and its computational expression (\tilde{P}_p) do not vary. Taking these aspects into account, error of each term is expressed as:

$$e = \|P_h - \tilde{P}_h/\Delta t\|_\infty$$

$$e = \|P_{pu} - \tilde{P}_{pu}\|_\infty$$

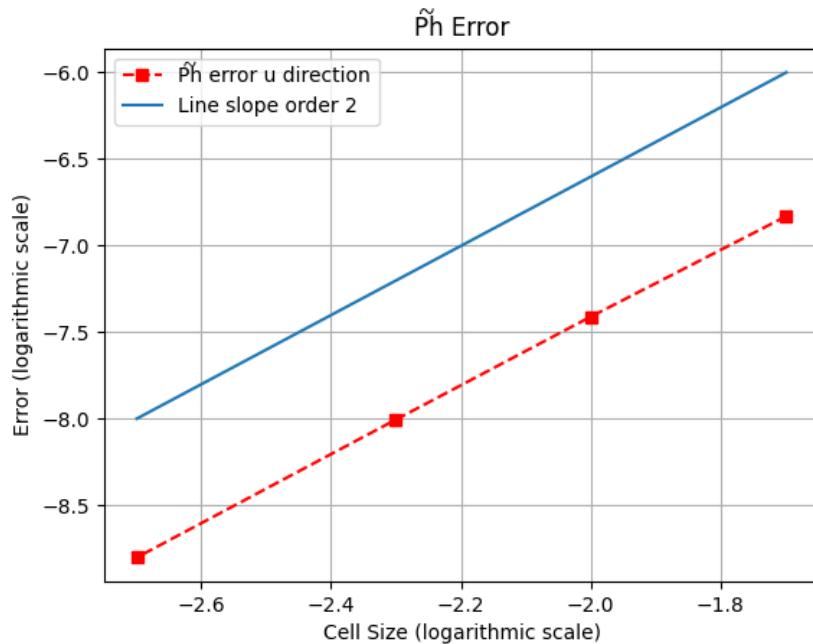


FIGURE 18: \tilde{P}_h continuity equation term validation. Constant Δt

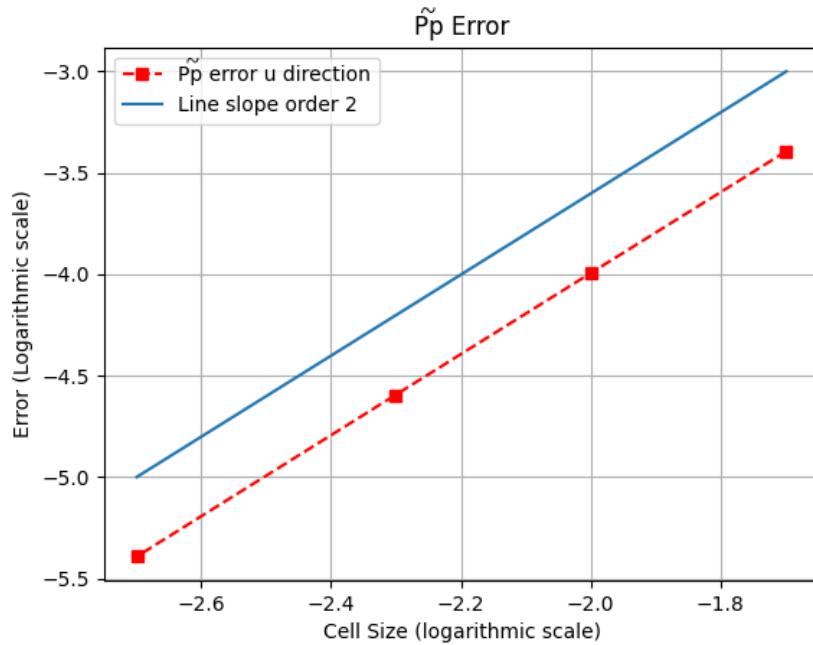


FIGURE 19: \tilde{P}_p pressure equation term validation. Constant Δt

Time integration Validation

In section(7.6) *Shallow Water* equations are integrated in time using a third order Adam Bashforth algorithm, consequently its results must be validate with the *MMS* method.

Adam Bashforth as such, has a third order accuracy. In this section the method is validated when it is applied to the Shallow Water model. It is important to notice that integration in time involves all the advection terms validated above, P_1 , P_2 ad P_p , so the final accuracy will be limited by the lowest accuracy term. This involves that order of accuracy decreases from Third to First order for the u and v velocity components while for the h component (not affected by Courant parameter) its still being second order. These results are agreement with previous studies [2].

Validation is carried out using the manufactured functions (8.0.6) nevertheless Coriolis and pressure terms are not included (6.1.23). The following simulation parameters are used:

- $\Delta x = [50, 100, 200, 500]$
- $\Delta t = 10^{-6}$

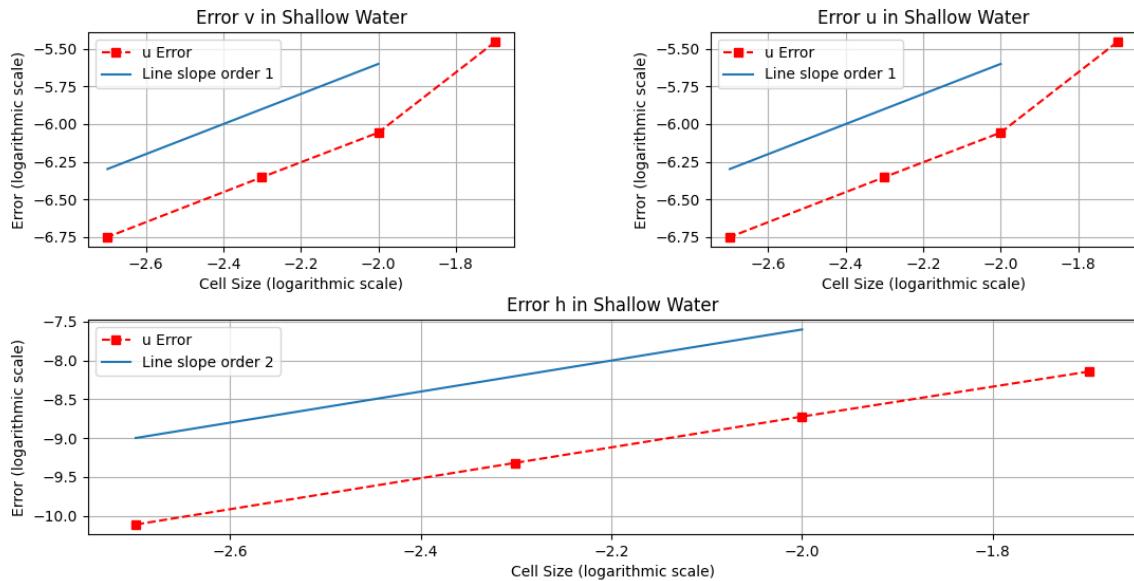
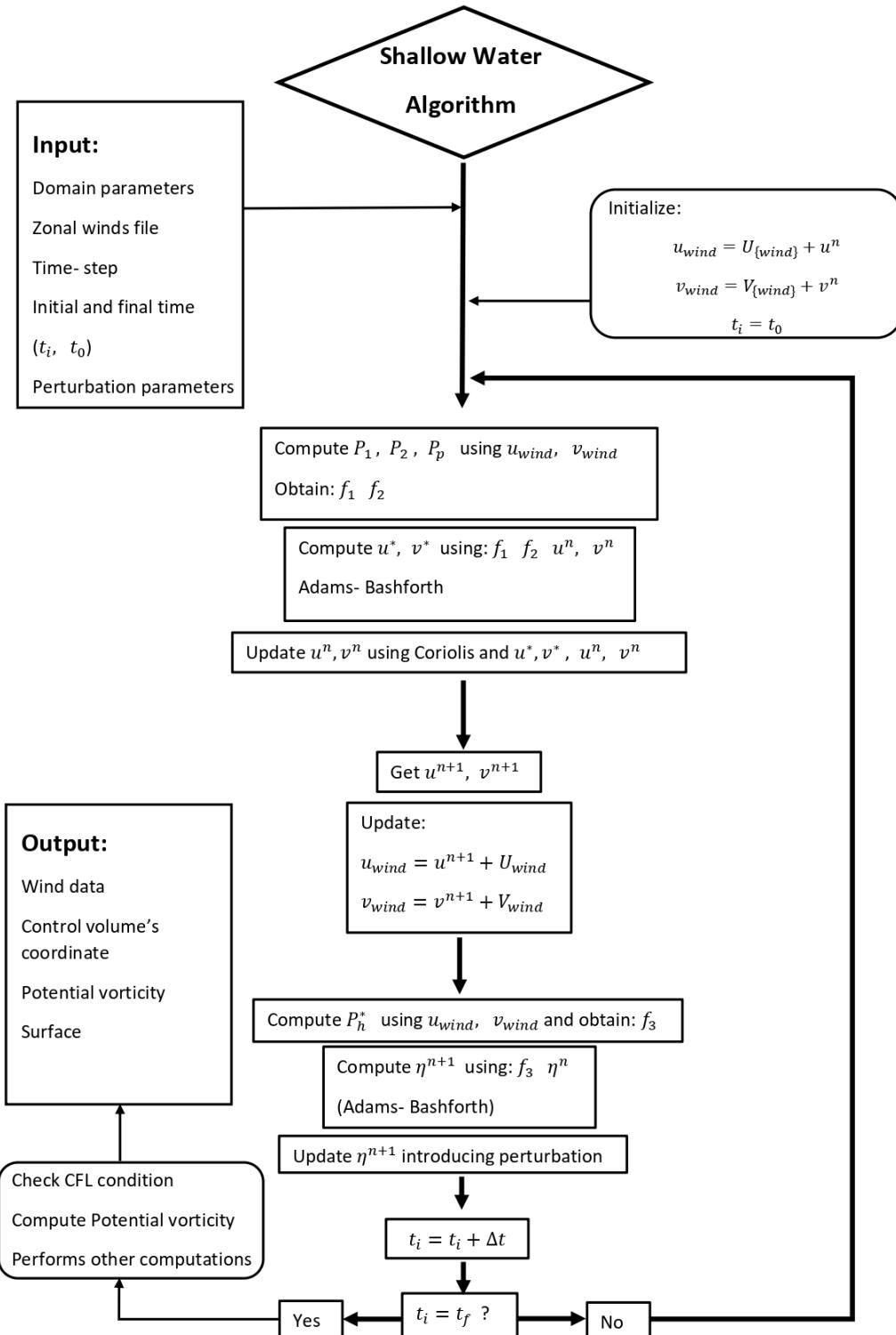


FIGURE 20: Adam Bashforth validation for the Shallow Water model. Constant Δt

9 Numerical Experiments

The numerical experiments are carried out using the following computational algorithm.



9.1 Gravity Wave Simulation

Section(7.1) explains some of the different types of grids used for atmospheric simulation purposes and which grid is selected for the *SW* model. In order to compute the fluid's behaviour at the frontier of the domain, boundary conditions (*BC*) must be defined. They are implemented using the *halo* concept.

The halo is a corona of cells around the domain, see Figure(10). These cells play an important role when u , v and η have to be analysed on the contour. In the *SW* model implemented in this text a double halo is needed, this allows to compute the advective terms: (7.3.11), (7.3.15) and (7.3.18) at the domain's frontier. It is important to remember that, in this code, number 1 is considered to be the first index of the matrix instead of number 0 (as *Python* language defines for default).

Taking this aspect into account, the length of the vector u , in the horizontal direction dimension N_x , is $u = [1, Nx + 5]$ where the domain's size (without the halos) is: $u = [3, Nx + 3]$.

The following *BC* are implemented in the code allowing to simulate different situations.

Double Periodic Conditions. DPC

Periodic conditions allows to simulate an infinite domain where the phenomena produced into the main grid is reproduced around it, see Figure(21). Its implementation is the same for u and v variables.

- Bottom

$$\begin{aligned} u[1, :] &= u[Ny + 1, :] & v[1, :] &= v[Ny + 1, :] \\ u[2, :] &= u[Ny + 2, :] & v[2, :] &= u[Ny + 2, :] \end{aligned}$$

- Top

$$\begin{aligned} u[Ny + 3, :] &= u[3, :] & v[Ny + 3] &= v[3, :] \\ u[Ny + 4, :] &= u[4, :] & v[Ny + 4, :] &= v[4, :] \end{aligned}$$

- Left

$$u[:, 2] = u[:, Nx + 2]$$

$$v[:, 2] = v[:, Nx + 2]$$

$$u[:, 1] = u[:, Nx + 1]$$

$$v[:, 1] = v[:, Nx + 1]$$

- Right

$$u[:, Nx + 3] = u[:, 3]$$

$$v[:, Nx + 3] = v[:, 3]$$

$$u[:, Nx + 4] = u[:, 4]$$

$$v[:, Nx + 4] = v[:, 4]$$

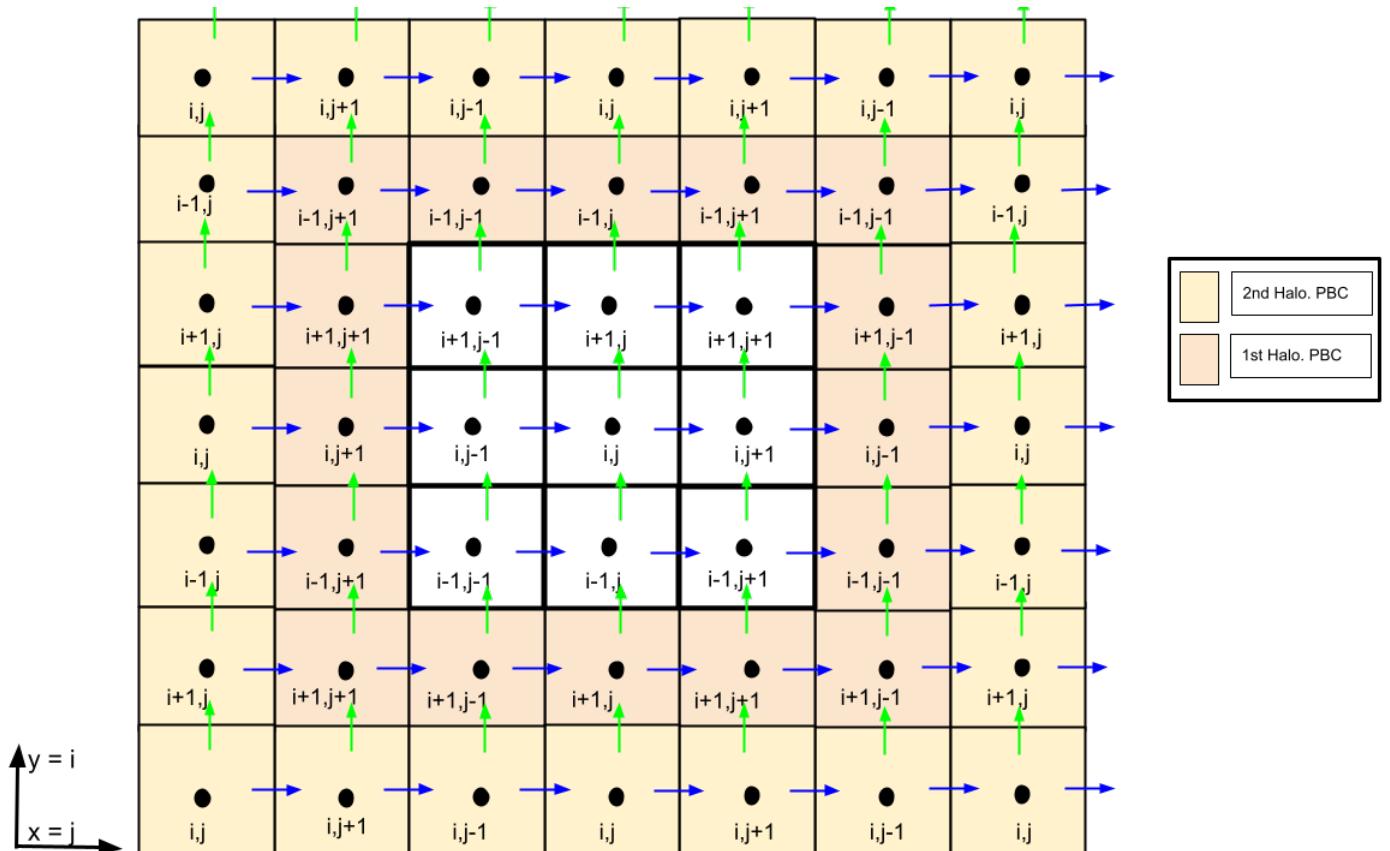


FIGURE 21: Double Periodic Conditions

The main idea is to represent the propagation of gravity waves when an initial disturbance is added. This simulation will help to corroborate if the model works as expected in a homogeneous and non-rotating layer of fluid, and also to start gaining some intuition on how the model works.

The disturbance is modelled using Gaussian function such as:

$$\eta(x, y) = A \exp \left[- \left(\frac{(x - x_0)^2}{2\theta_x^2} + \frac{(y - y_0)^2}{2\theta_y^2} \right) \right] \quad (9.1.1)$$

Where A is the amplitude of the function, (x_0, y_0) establishes the *Gaussian's* centre and θ_x , θ_y stand for standard deviations of the function.

These values can not be randomly chosen, the *Courant* condition must be always fulfilled:

$$\mathcal{C} = \frac{U\Delta t}{\Delta x}$$

If Courant number is fixed at a value of $\mathcal{C} = 0.5$ and domain is discretized in a constant number of square cell $N = 100$, then time increment can be estimated considering that the phase speed propagation of a gravity wave in a *SW* system for $D = 5m$ is $U = \sqrt{gD} = 7.00m/s$, resulting:

$$\Delta t|_{max} = 0.14s$$

However, in order to obtain a higher resolution, time step of $\Delta t = 0.05$ s is used.

Taking this aspects into account, gravity wave simulation is carried out in a square of (200m x 200m) with a fluid's layer depth of $D = 5m$. Initial disturbance geometry is defined by:

A [m]	θ_x	θ_y	x_0 [m]	y_0 [m]
1	4	4	100	100

TABLE 2: Wave simulation (PBC). Simulation parameters

In Figure(23) the simulated gravity wave is shown at different time steps.

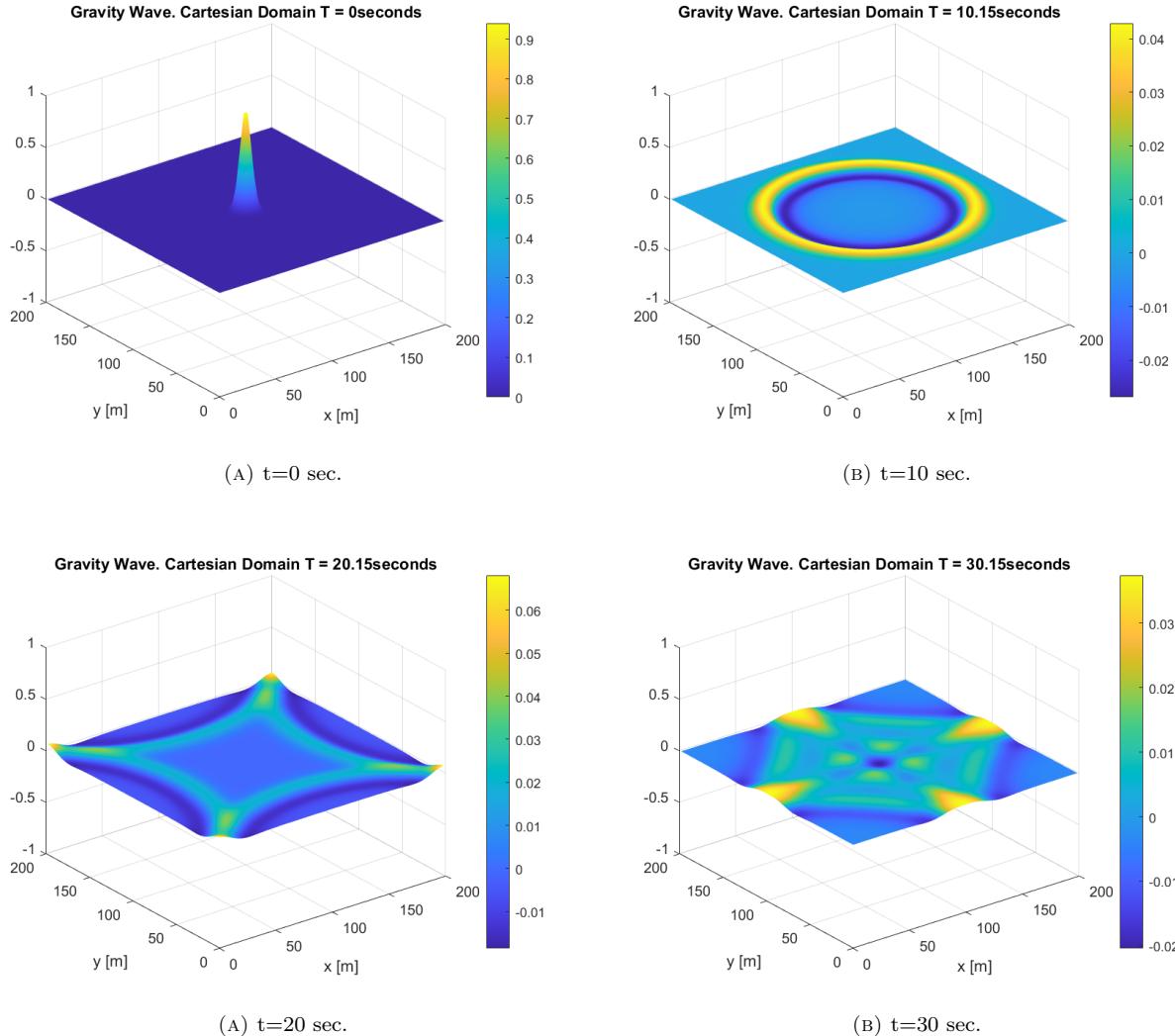
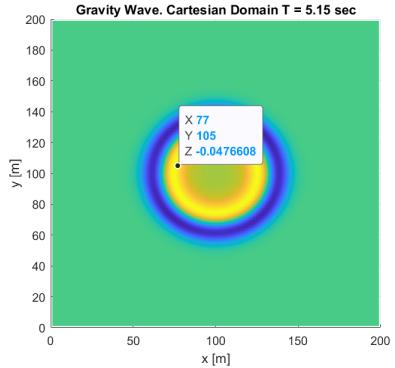


FIGURE 23: Wave simulations at different time steps. Cartesian domain (200 x 200 m) and Periodic Boundary Conditions (PBC)

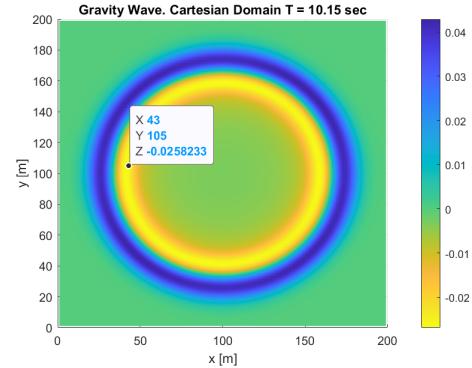
t_0 [s]	x [m]	y [m]	η [m]
0	101	101	0.94
10	175	103	0.041
20	27	27	0.04
30	101	173	0.026

TABLE 3: Wave simulation (PBC). Surface elevation at different points and time steps

Finally, the wavefront velocity can be compared for the numerical and analytical cases. As explained above, the wavefront is moving at a speed of 7m/s ($c_a = 7\text{m/s}$). Comparing waves of two different time steps, velocity can be computed.



(A) Wavefront at $t=5.15$ sec.



(B) Wavefront at $t=10.15$ sec.

$$c_n = \frac{x_2 - x_1}{t_2 - t_1} = \frac{77 - 43}{10.15 - 5.15} = 6.8\text{m/s}$$

Error between analytical and numerical value is: $\epsilon = 3\%$, which is affordable.

This simulation is carried out using a Cartesian domain with a double periodic boundary condition. However, this is not the only possibility, other Boundary conditions can be implemented.

Full Slip Conditions. FSC

This boundary conditions represents a closed domain where any particle of fluid can go through its boundaries, however when the fluid reaches the contour wall, it slides down the contour wall. It is an ideal situation without friction but it is very useful to simulate some cases with negligible friction.

- Bottom

$$\begin{aligned} u[1, :] &= u[3, :] & v[1, :] &= 0 \\ u[2, :] &= u[3, :] & v[2, :] &= 0 \end{aligned}$$

- Top

$$\begin{aligned} u[Ny + 3, :] &= u[Ny + 2, :] & v[Ny + 3, :] &= 0 \\ u[Ny + 4, :] &= u[Ny + 2, :] & v[Ny + 4, :] &= 0 \\ . & & v[Ny + 2, :] &= 0 \end{aligned}$$

- Left

$$\begin{aligned} u[:, 2] &= 0 & v[:, 2] &= v[:, 3] \\ u[:, 1] &= 0 & v[:, 1] &= v[:, 2] \end{aligned}$$

- Right

$$\begin{aligned} u[:, Nx + 3] &= 0 & v[:, Nx + 3] &= v[:, Nx + 2] \\ u[:, Nx + 4] &= 0 & v[:, Nx + 4] &= v[:, Nx + 3] \\ u[:, Nx + 2] &= 0 & & \end{aligned}$$

In this case η halo must be also adapted. In *FSC* conditions η is established as 0 in the hole halo positions.

The same simulation using *FSC* conditions is shown in Figure(27). In this case, the same initial simulation parameters are used. The only difference is at the location of the perturbation which is displaced from centre to the position: ($x_0 = 75$, $y_0 = 75$). Now the reflection of the wave at the contour of the domain can be appreciated.

Figure(25) shows the main domain and *FSC* applied to the halos. It is important to remember that surface elevation (η) is settled to 0 in the halos.

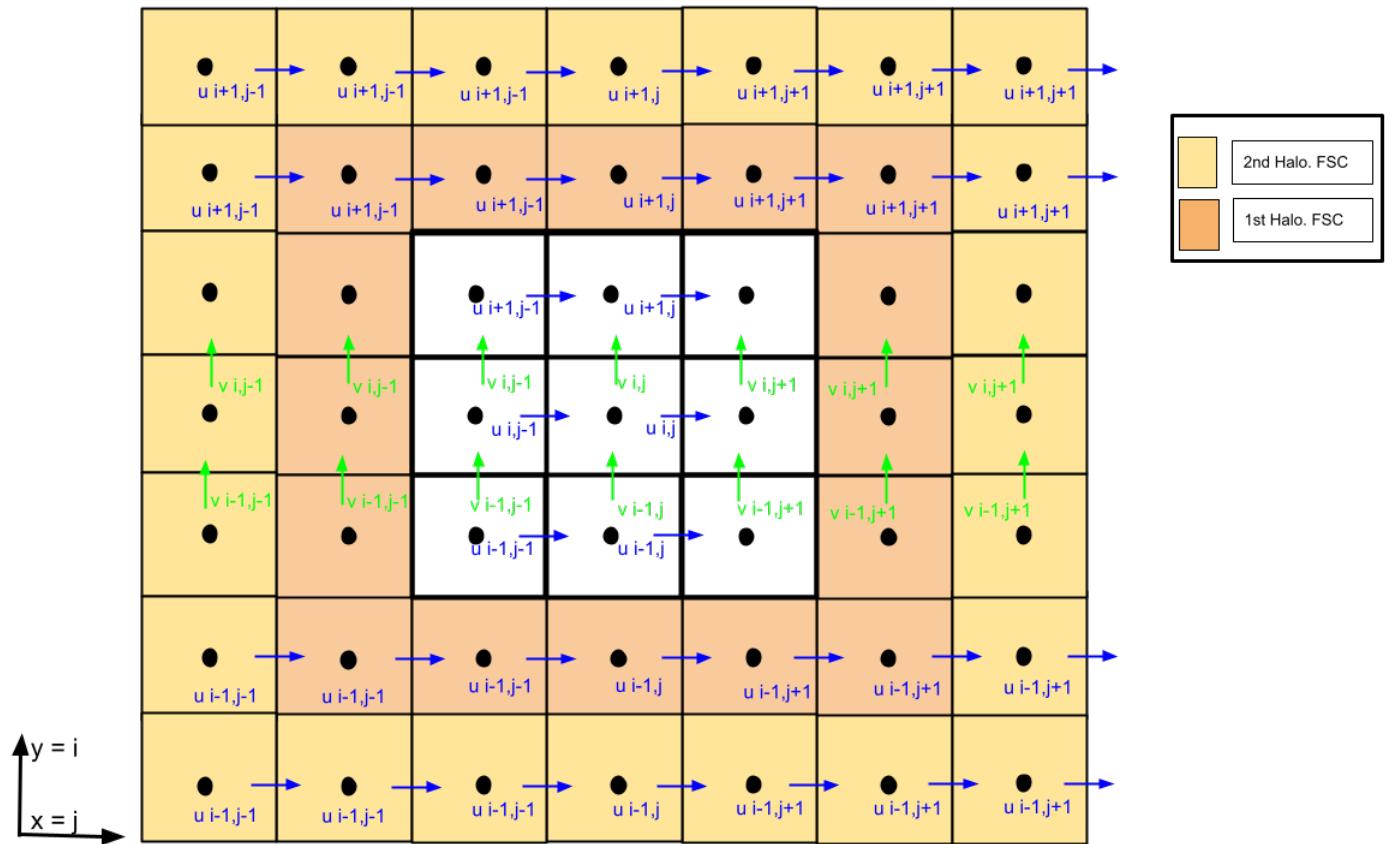


FIGURE 25: Full Slip Conditions

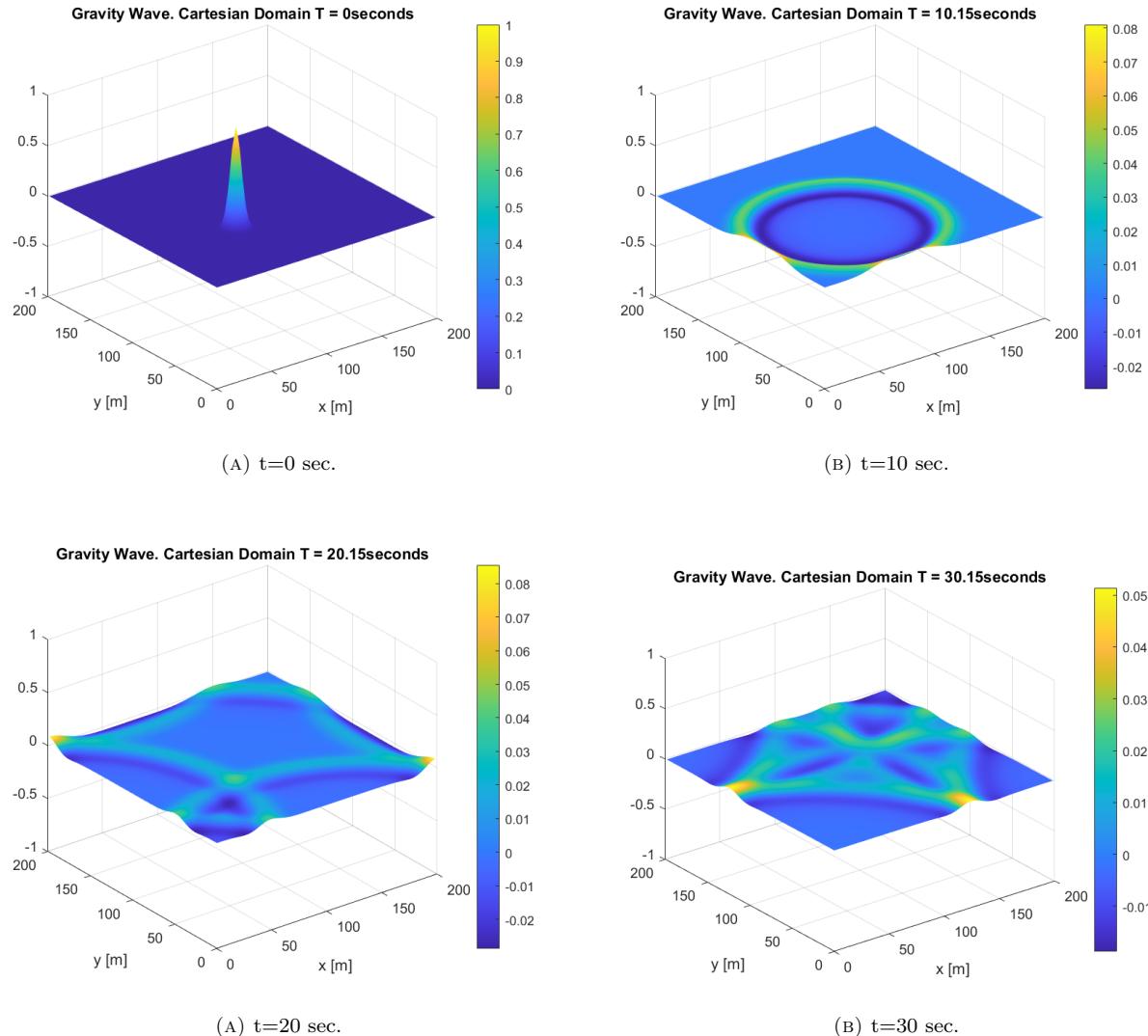


FIGURE 27: Wave simulations at different time steps. FSC Boundary Conditions.
Cartesian domain (200 x 200 m)

t_0 [s]	x [m]	y [m]	η [m]
0	75	75	1
10	79	149	0.041
20	71	69	0.041
30	127	153	0.025

TABLE 4: Wave simulation results. FSC Boundary Conditions

9.2 Cartesian vortex simulation

Once the gravity wave is successfully implemented the next step is to introduce a vortex into the model. By means of this simulation the integration of the Coriolis parameter can be tested (see Section(7.7)) as well as new types of boundary conditions.

Different simulations will be carried out using distinct boundary conditions. Firstly, vortex is introduced in a Cartesian and double periodic domain, while for the second simulation Canal conditions in a elliptical domain are used. This allows to test each part of the code independently.

The perturbation for the vortex creation and the zonal wind are created using the geostrophic balance.

9.2.1 Geostrophic balance

If $R_0 \ll 1$, it can be shown from a scale analysis of equations(6.1.12a) and (6.1.12b), that the material derivatives of u and v can be neglected in front of the Corilis acceleration $\vec{f} \times \vec{u} = f\vec{k} \times \vec{u}$, and therefore the momentum equations are simplified to

$$\begin{aligned} -fv &= -g \frac{\partial \eta}{\partial x} \\ fu &= -g \frac{\partial \eta}{\partial y} \end{aligned} \quad (9.2.1)$$

Or

$$\vec{g} \times \vec{u} = -g \Delta \eta \quad (9.2.2)$$

Then it is said that the velocity is in geostrophic equilibrium with the dynamic pressure. It means that when this balance holds, wind speed is directly proportional to the dynamic pressure. This wind is also known as geostrophic wind. Equation(9.2.2) indicates that the wind gets stronger as the pressure gradient increases, or in other words, when the isobars on the dynamic pressure (except for a constant depending only on z , see section(6.1)), $\rho(x, y, t) = \rho_0 g \eta(x, y, t)$, get closer together. Equation(9.2.2) also tells that geostrophic wind must be perpendicular to isobars.

Equation(9.2.2) is also called a *diagnostic* equation, because it just expresses the balance of two quantities, \vec{u} and $\vec{\eta}$, but not their time evolution. So this equation has infinite solutions.

On the other hand, (6.1.12) is a pair of *prognostic* equations, because give η , we can compute the evolution of the horizontal velocities. Equations (6.1.23) are in fact a set of three prognostic equations which completely determine the three variables, u , v and η , and its evolution in time.

If the geostrophic wind is strictly zonal, $v = 0$, $u = U(y)$, and according to (9.2.1) η can only be y dependent. In this case $U(y)$ can be obtained as

$$\eta_z(y) = - \int \frac{f(y)}{g(y)} U(y) dy + const \quad (9.2.3)$$

$f(y)$ and $g(y)$ indicate that the Coriolis parameter and gravity can also be a function of latitude. $\eta_z(y)$ is the resulting topography for the surface layer in geostrophic balance with $U(y)$. This is a very interesting result, because in the SW simulations of the atmospheres of Jupiter and Saturn, (9.2.4) shows that we can introduce zonal wind as free surface topography.

Given $U(y)$, SW model should be initialized by computing $\eta_z(y)$, fixing it as the initial topography of the free surface, and introducing $U(y)$ as the velocity initial condition ($u_0 = U$; $v_0 = 0$). Afterwards, perturbations to $\eta_z(y)$ may be simply added and let the model evolve freely. Nevertheless we show below that we can simplify our model without the need of introducing $\eta_z(y)$. Although it is out of the scope of this course, if we introduce $\eta(y)$, we also need to introduce topography to the bottom of the model if want to keep it still stable, but we will circumvent it in the next section.

9.2.2 Introduction zonal winds to the SW equations

Let's assume that the free surface topography can be decomposed into the sum $\eta = \eta' + \eta_z$, where η' represents the disturbance added to the zonal wind topography η_z . The resulting equation for v is therefore

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + fu = -g \frac{\partial \eta'}{\partial y} - g \frac{\partial \eta_z}{\partial y} = -g \frac{\partial \eta'}{\partial y} + fU \quad (9.2.4)$$

This equation can be written as

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + f(u - U) = -g \frac{\partial \eta'}{\partial y} \quad (9.2.5)$$



If the velocity u is decomposed as the sum of a perturbation and the zonal wind $u = u' + U$. Equation(9.2.5) can be written as

$$\frac{\partial v}{\partial t} + (u' + U)\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + fu' = -g\frac{\partial \eta'}{\partial y} \quad (9.2.6)$$

In Equation(9.2.6), v is equal to the perturbation value v' , but the notation will not be changed.

If the same decomposition is applied to the u equation, the following expression is obtained

$$\frac{\partial u'}{\partial t} + (u' + U)\frac{\partial u'}{\partial x} + v\frac{\partial(u' + U)}{\partial y} - fv = -g\frac{\partial \eta'}{\partial x} \quad (9.2.7)$$

Now, the mass conservation equation must be

$$\frac{\partial \eta}{\partial t} + \frac{\partial(h(u' + U))}{\partial x} + \frac{\partial(hv)}{\partial x} = 0 \quad (9.2.8)$$

Or in other words, both situations both, adding the topography η_z to the free surface or adding algebraically the zonal wind $U(z)$ to the horizontal velocity perturbation are *absolutely equivalent*. In both cases we must impose the condition that h must have the undisturbed thickness D. Numerically is quite simple, we must only add the zonal wind to the perturbation velocity but *not* to the Coriolis acceleration term.

Zonal winds for Jupiter and Saturn are listed in the Appendix.

9.2.3 Perturbation introduction

Vortex in the *SW* model is created by the combination of a Gaussian function and the Coriolis effect. Gaussian function has the same expression as the one seen in the previous paragraph for the Cartesian domain.

$$\mathcal{S}(x, y) = A \exp \left[\left(-\frac{(x - x_0)^2}{2\theta_x^2} \right) - \left(\frac{(y - y_0)^2}{2\theta_y^2} \right) \right] \quad (9.2.9)$$

Vortex is introduced progressively into the simulation, this will avoid some instabilities into the model since a sudden disturbance can cause the model to become unstable for a long time, delaying the emergence of the vortex.

The main idea is to increment vortex height's at every iteration until the 25% of the total simulation time is reached. At this point, no more flux is added to the surface perturbation and the Gaussian function drops over time as can be seen in Figure(28).

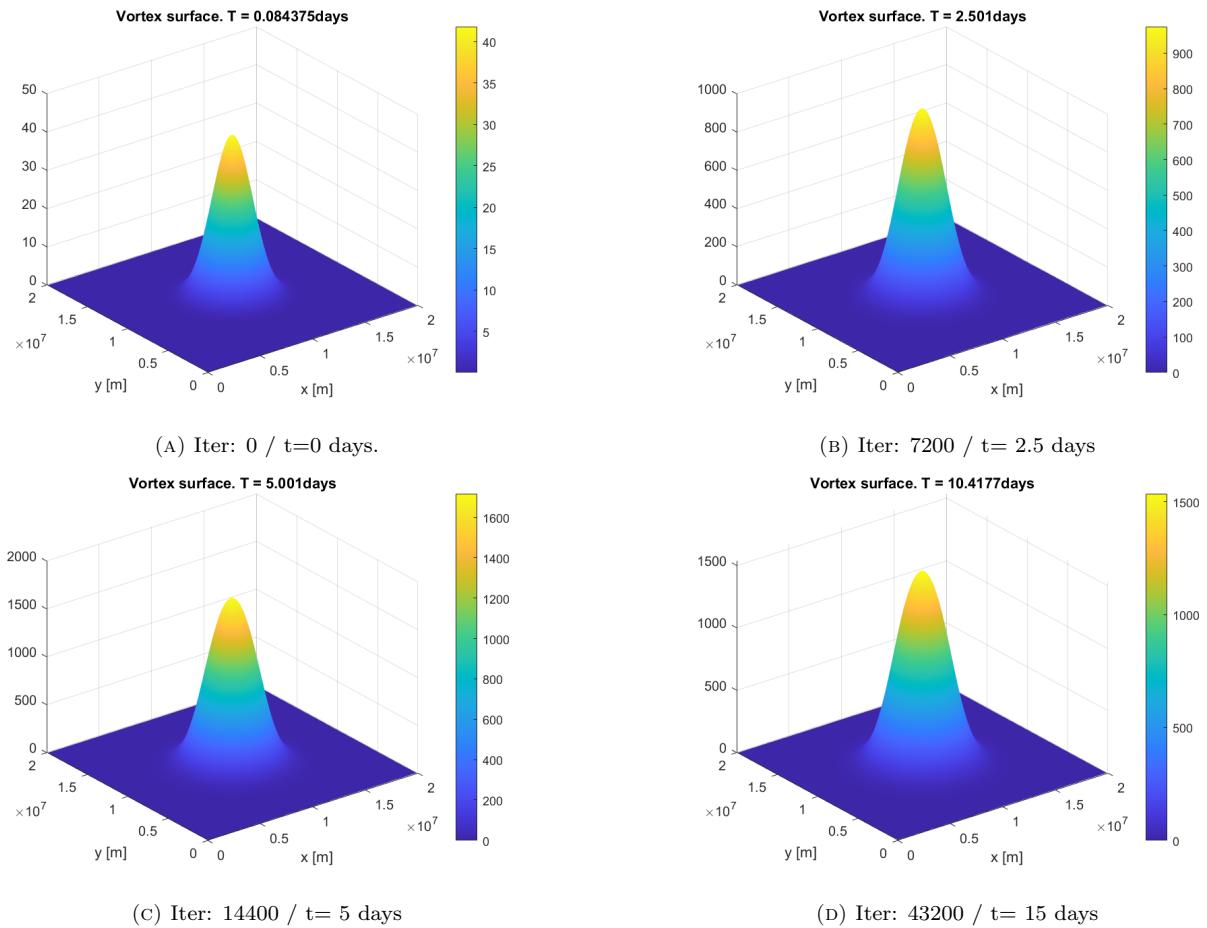


FIGURE 28: Surface elevation at different time steps. Cartesian domain. $\Delta t = 30s$. $t_f = 20$ days

It is important to notice about the numerical dissipation which creates a decrease in the final height of the vortex. This effect is produced due to the high time step ($\Delta t = 30\text{s}$) used in simulation carried out in Figure(28), the lower the time step, the lower the numerical dissipation.

	A (Max amplitud)	R_{max*}	θ_x	θ_y
Domain: Cartesian	3000 m	3500km	1400	1400

TABLE 5: Initial parameters used to introduce surface perturbation into the Cartesian domain

* R_{max} is the maximum radius of the Gaussian. For larger radii the value of the perturbation is set to 0.

Vortex rotation is displayed using the potential vorticity ($\Pi = q$). This property is conserved through time (as it is explained in section(6.2)) which allows to use it as a tracer. Tracers displays the path of a fluid's particle over the domain (like a cloud does in the atmosphere).

Potential vorticity is expressed as:

$$\Pi = q = \frac{f + \omega}{h} \quad (9.2.10)$$

Where f stands for the Coriolis parameter, ω is the relatively vorticity and h is the total fluids height.

However, q is computed at each up-right cell's corner as can be seen in Figure(10). This fact requires an interpolation for the η variable at equation: $h = D + \eta$

$$h = D + \frac{\eta(i, j) + \eta(i, j + 1) + \eta(i + 1, j) + \eta(i + 1, j + 1)}{4}$$

While relatively vorticity is computed such as:

$$\omega = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$$

9.2.4 Simulation results

Cartesian Vortex

Using the parameters of table(5) the Gaussian function is introduced into the model with a total simulation time of 20 days. The vortex grows for 5 days and the remaining 15 days it is left to evolve freely. In order to reduce the numerical dissipation $\Delta t = 10s$. Jupiter planetary parameters are used to establish the angular velocity of the planet ($\Omega = 1.76 \cdot 10^{-4} rad/sec$) and its gravity ($g_0 = 24.79m/s^2$) .

Finally, the domain is a square ($L_1 \times L_2$) with $L_1 = L_2 = 20000km$ and the layer's deep is set to $D = 1000m$. After the simulation is carried out, the following results are obtained.

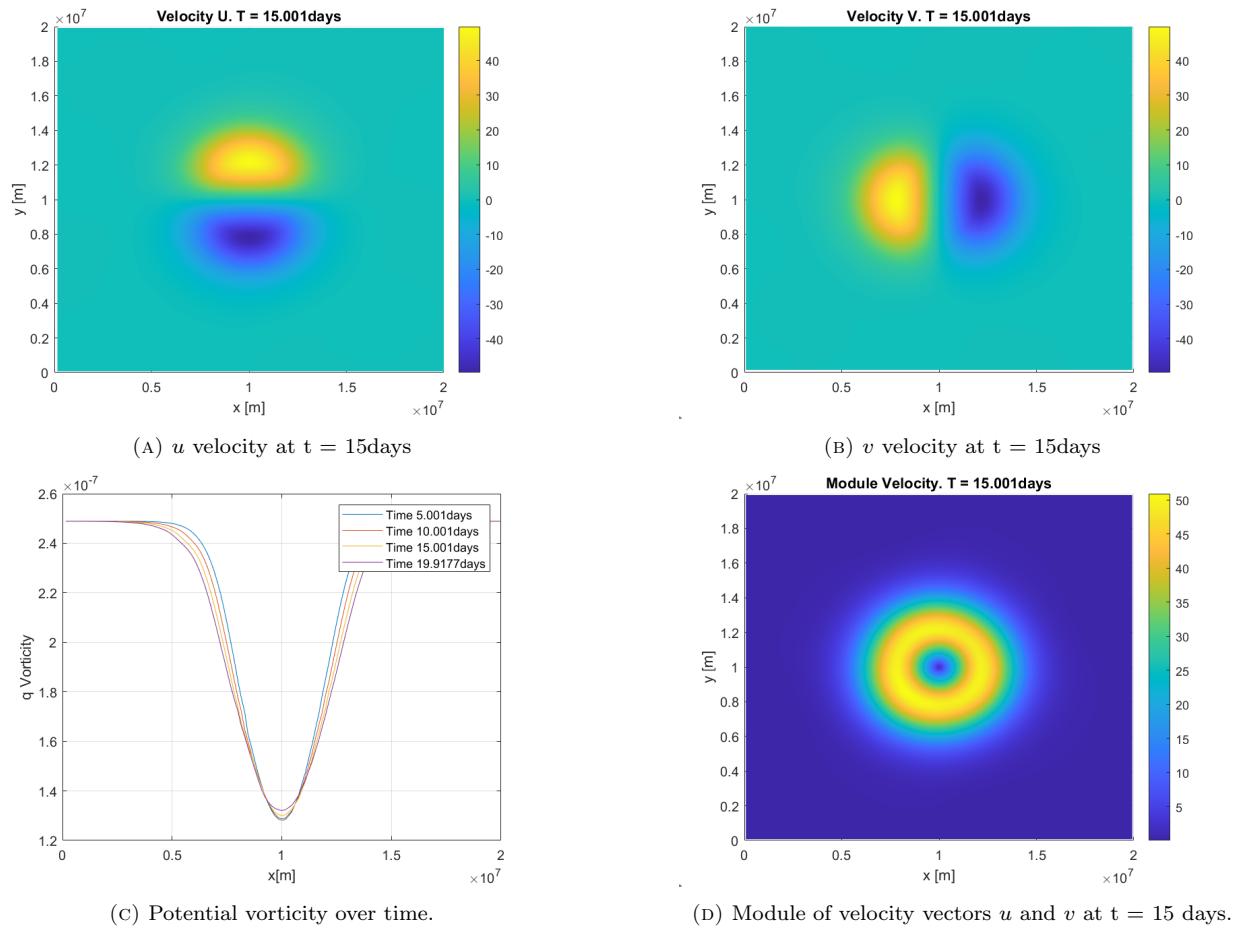


FIGURE 29: Cartesian vortex parameters over the domain (2000 x 2000) km. Time simulation: 20 days

Figure(29c) shows vortex's potential vorticity along the x-axis (y-axis is fixed at $y = L_2/2$) at different time steps. This result corroborates the theory that states the conservation of potential vorticity (section(5.1)) since the small variations observed are due to numerical dissipation.

On the other hand, Figure(29a) and Figure(29b) show the clockwise vortex rotation at time step ($t = 15\text{days}$) while Figure(29d) shows the module of the velocity components u and v inside the vortex.

This rotation is more appreciated displaying the potential vorticity along the domain as a tracer.

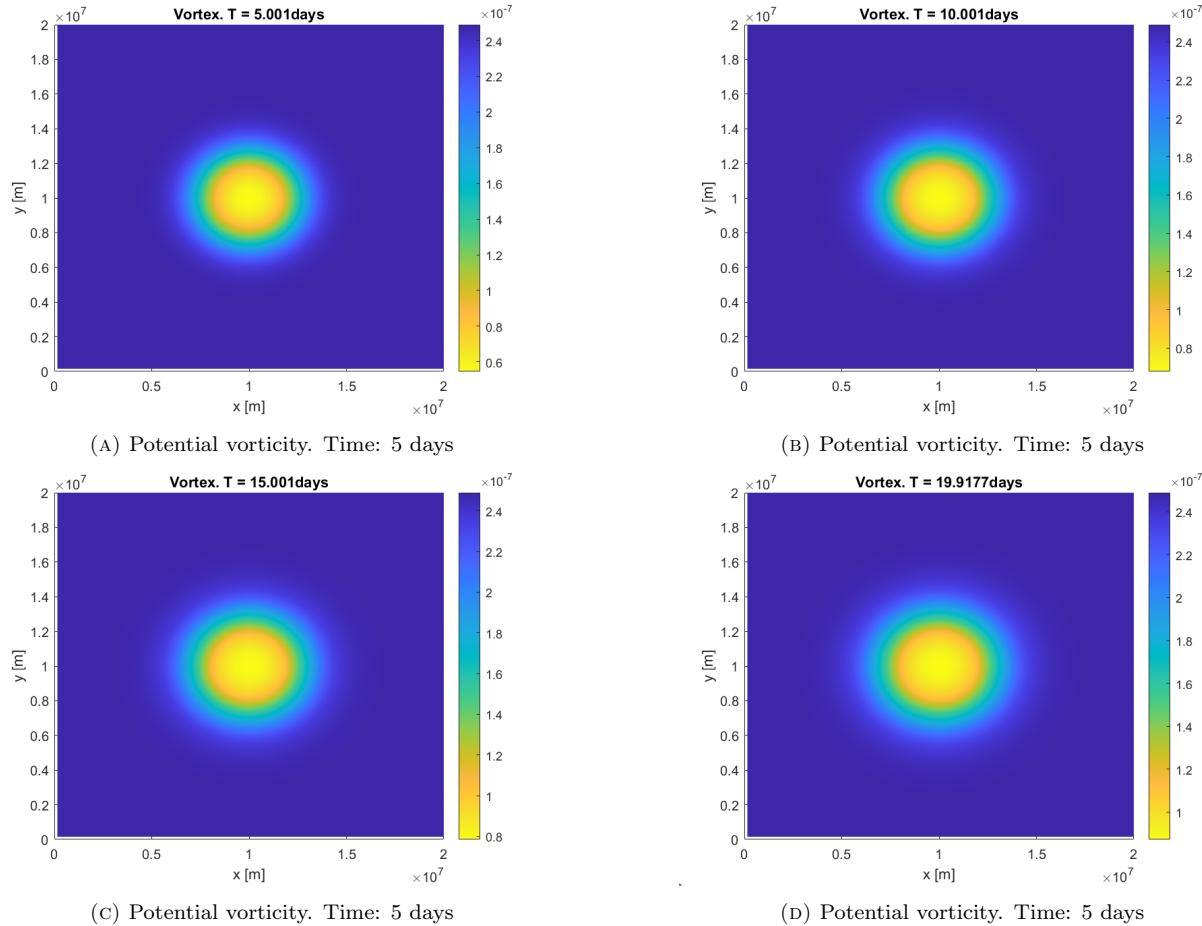


FIGURE 30: vortex at different time steps. Domain (2000 x 2000)km. Time simulation: 20 days

In Figure(30) its rotation is slightly appreciated, in order to obtain a better resolution number of points in the domain can be increased. In this simulation $N_x = N_y = 100$ points were used.

9.3 Elliptical vortex simulation

The next step is to introduce zonal winds explained at section(9.4) in a planetary domain. To do so, elliptic equations are used.

9.3.1 Derivation of ellipsoidal coordinates

Canal conditions are implemented in order to simulate a region between two longitudes. It is specially suited to simulate the atmospheres of the giant planets. However these boundary conditions can not be applied if the *SW* equations(6.1.23) are not adapted to the planetary curvature of the atmosphere.

In order to do so, the domain is considered to be an ellipsoid (i.e a revolution ellipse). This leads to define a new coordinate system which can be seen in Figure(31). In this new domain a displacement in the x -axis is related to a variation in longitude while a displacement in y -axis is related to a latitude displacement.

- $x \rightarrow \theta$ (longitude)
- $y \rightarrow \varphi$ (latitude)

The first term to be adapted to ellipsoidal coordinates is the continuity equation, which is the easiest. This change in the frame of reference is done applying the divergence operator in elliptical coordinates to a vector field \vec{F} . This is:

$$\nabla \cdot \vec{F} = \frac{1}{r_Z(\varphi)} \frac{\partial F_\theta}{\partial \theta} + \frac{1}{r_M(\varphi)} \frac{\partial F_\varphi}{\partial \varphi} - \frac{\sin(\varphi)}{r_Z(\varphi)} F_\varphi \quad (9.3.1)$$

Obtaining the new expression for the continuity equation:

$$\frac{\partial \eta}{\partial t} + \frac{1}{r_Z(\varphi)} \frac{\partial(hu)}{\partial \theta} + \frac{1}{r_M(\varphi)} \frac{\partial(hv)}{\partial \varphi} - \frac{\sin(\varphi)}{r_Z(\varphi)} hv = 0 \quad (9.3.2)$$

Where $r_Z(\varphi)$ and $r_M(\varphi)$ stands for the zonal and meridional radius respectively. Its use helps to simplify the notation which is commonly used in Jovian planets literature. Both variables can be related to the geometrical parameters of the planet by the expressions:



$$\begin{aligned} r_z(\varphi) &= \frac{R_e^2}{\sqrt{R_p^2 \tan^2(\varphi) + R_e^2}} \\ r_M(\varphi) &= \frac{r_z(\varphi) / \cos(\varphi)}{\sin^2(\varphi) + \left(\frac{R_e}{R_p}\right)^2 \cos^2(\varphi)} \end{aligned} \quad (9.3.3)$$

Where φ is the planetographic latitude to be defined later.

In the momentum equation an especial treatment is needed for the advection term. It is calculated using the divergence operator in conjunction with the relation: $(\vec{u} \cdot \nabla)\vec{u} = \nabla(\vec{u} \cdot \vec{u}/2) - \vec{u} \times (\nabla \times \vec{u})$. Its mathematical development is a tedious procedure so the final expression obtained is written below. It can also derived by using tensor calculus.

$$(\vec{u} \cdot \nabla)\vec{F} = \begin{bmatrix} \frac{u_1}{r_z(\varphi)} \frac{\partial F_\theta}{\partial \theta} + \frac{u_2}{r_M(\varphi)} \frac{\partial F_\theta}{\partial \varphi} - \frac{\sin(\varphi)}{r_z(\varphi)} u_1 u_2 \\ \frac{u_1}{r_z(\varphi)} \frac{\partial F_\varphi}{\partial \theta} + \frac{u_2}{r_M(\varphi)} \frac{\partial F_\varphi}{\partial \varphi} + \frac{\sin(\varphi)}{r_z(\varphi)} u_1 u_1 \end{bmatrix} \quad (9.3.4)$$

The second term to be analysed is the Coriolis force. For its mathematical development, derivation of scale parameters in curvilinear coordinates is needed. It can be found at section: "Appendix A" from reference [2].

Once all this process is carried out, it is concluded that the Coriolis term remains invariant. i.e: $f = 2\Omega \sin(\varphi)$. f is called the Coriolis parameter.

Applying the change of coordinates from Cartesian to ellipsoidal, the following expressions is obtained.

$$\vec{F}_c = \begin{bmatrix} -2\Omega v \sin(\varphi) \\ 2\Omega u \sin(\varphi) \\ 0 \end{bmatrix} = \begin{bmatrix} -fv \\ fu \\ 0 \end{bmatrix} \quad (9.3.5)$$

Taking all these aspects into account, the momentum equation is rewrite as:

$$\begin{aligned} \frac{\partial u}{\partial t} + \frac{u}{r_z(\varphi)} \frac{\partial u}{\partial \theta} + \frac{v}{r_M(\varphi)} \frac{\partial u}{\partial \varphi} - \left(f + \frac{\sin(\varphi)}{r_z(\varphi)} u \right) v &= \frac{-g}{r_z(\varphi)} \frac{\partial \eta}{\partial \theta} \\ \frac{\partial v}{\partial t} + \frac{u}{r_z(\varphi)} \frac{\partial v}{\partial \theta} + \frac{v}{r_M(\varphi)} \frac{\partial v}{\partial \varphi} + \left(f + \frac{\sin(\varphi)}{r_z(\varphi)} u \right) u &= \frac{-g}{r_M(\varphi)} \frac{\partial \eta}{\partial \varphi} \end{aligned} \quad (9.3.6)$$

Where the gradient of a scalar $\eta(\varphi, \theta)$ is

$$\Delta \eta = \left[\frac{1}{r_z(\varphi)} \frac{\partial \eta}{\partial \theta}, \frac{1}{r_M(\varphi)} \frac{\partial \eta}{\partial \varphi}, 0 \right]$$

Summing up, the *Shallow Water* equations are presented below in the ellipsoidal frame of reference.

$$\begin{aligned} \frac{\partial u}{\partial t} + \frac{u}{r_Z(\varphi)} \frac{\partial u}{\partial \theta} + \frac{v}{r_M(\varphi)} \frac{\partial u}{\partial \varphi} - \left(f + \frac{\sin(\varphi)}{r_Z(\varphi)} u \right) v &= \frac{-g}{r_Z(\varphi)} \frac{\partial \eta}{\partial \theta} \\ \frac{\partial v}{\partial t} + \frac{u}{r_Z(\varphi)} \frac{\partial v}{\partial \theta} + \frac{v}{r_M(\varphi)} \frac{\partial v}{\partial \varphi} + \left(f + \frac{\sin(\varphi)}{r_Z(\varphi)} u \right) u &= \frac{-g}{r_M(\varphi)} \frac{\partial \eta}{\partial \varphi} \\ \frac{\partial \eta}{\partial t} + \frac{1}{r_Z(\varphi)} \frac{\partial(hu)}{\partial \theta} + \frac{1}{r_M(\varphi)} \frac{\partial(hv)}{\partial \varphi} - \frac{\sin(\varphi)}{r_Z(\varphi)} hv &= 0 \\ h - D - \eta + h_b &= 0 \end{aligned} \quad (9.3.7)$$

Computationally thinking, the fact of changing the domain from Cartesian to Ellipsoidal does not affect to the numerical expression. However, some aspects have to be noticed.

Firstly, in the continuity equation an additional terms appears when Elliptical coordinates are introduced. This term has to be added to the P_{1h} before apply the Adams Bashforth integration.

$$f_3 = P_{1h} - \Delta t \cdot \left(\frac{\sin(\varphi)}{r_Z(\varphi)} h^n[i, j] v^n[i, j] \right) \quad (9.3.8)$$

Secondly, distance's increments in the advective terms (Δx and Δy) have to be re-defined to adapt to the new form of the mesh. In order to do so, the following distances are defined along the domain:

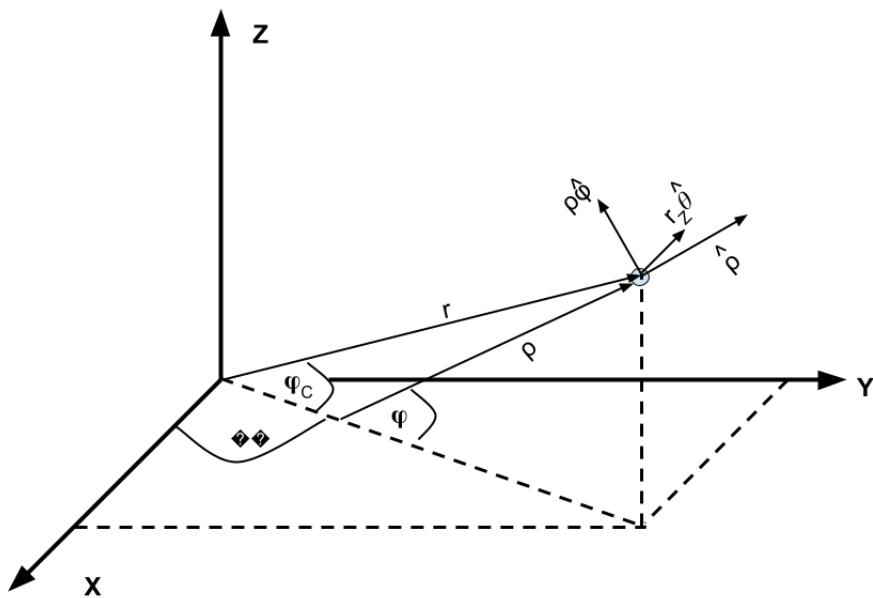
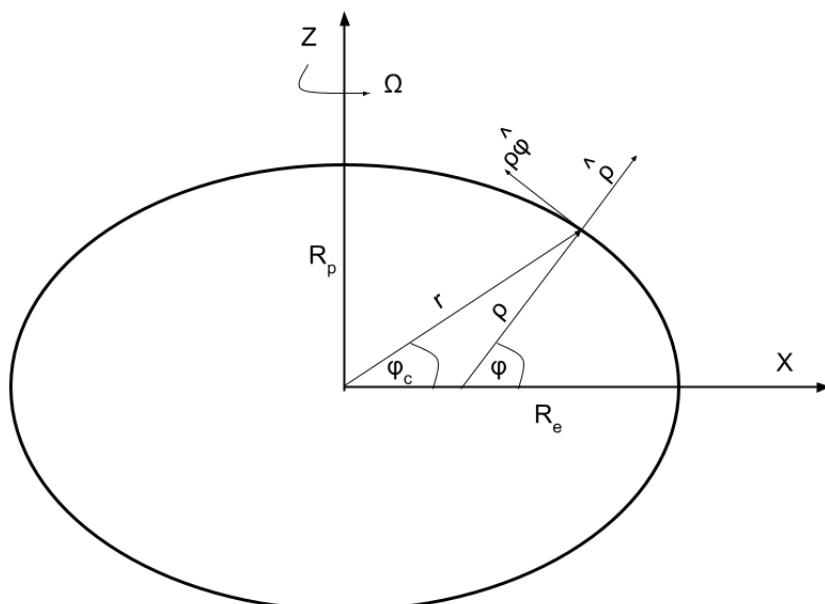


FIGURE 31: Elliptical coordinates


 FIGURE 32: Elliptical coordinates. XZ plane projection

Text by Sánchez-Lavega (2011) is followed. On Earth, spherical coordinates are used to locate a point on the surface or atmosphere: the longitude λ and latitude angles φ , and the distance r to the centre of the planet. This coordinate system is referred with respect the rotation axis and the geometric centre, assuming that is shape is a perfect sphere. In the case of Jupiter and Saturn, due to its fluid nature and their rapid rotation, these planets have the shape of an oblate ellipsoid of revolution. For these planets, no rigid reference observable surface exists, so oblate spherical coordinates are used. In this situation there are two definitions of latitude (Figure (33)).

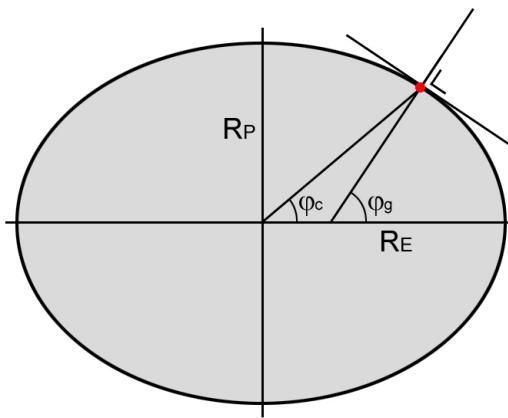


FIGURE 33: Definitions of latitude for an oblate planet. R_E and R_P are the equatorial and polar radii respectively. φ_c and φ_g the planetocentric and planetographic latitudes. The red dot is a surface point.

The planetographic latitude (φ_g) is the angle between the line perpendicular to a tangent plane containing a surface point on the planet's ellipsoid, and the semimajor axis, and the planetocentric latitude (φ_c) is the angle of a line passing through the centre of the planet and a surface point and the semi-major axis. Both are related by the expression

$$\tan(\varphi_g) = \left(\frac{R_E}{R_P} \right)^2 \tan(\varphi_c) \quad (9.3.9)$$

Expression(9.3.9) is very useful for transformation between both latitudes. In the SW equations (9.3.3), (9.3.5), and (9.3.7), are given as a function of planetographic latitude.

In rapidly rotating planets such as Jupiter and Saturn, we must take into account the effect of centrifugal acceleration on the planet's rotating frame of reference. In dis case, gravity at any planetocentric latitude (φ_c), gravity must be corrected in the following way:

$$g(\varphi_c) = g_0(\varphi_c) - \Omega^2 r(\varphi_c) \cos(\varphi_c) \quad (9.3.10)$$

Where g_0 is expressed as:

$$g_0(\varphi_c) = \frac{GM}{r(\varphi_c)^2}$$

and G is the universal gravitational constant, M is the mass of the planet and Ω is the rotation angular velocity.

Geometrical parameter $r(\varphi_c)$ also has to be defined.

Finally, boundary conditions used for simulations representing a planet strip have to be presented.

Canal Conditions.

Once DPC and FSC are defined, the canal conditions are trivial. It is used to represent the planetary domain for a given values of longitude and latitude ranges.

This conditions are implemented using periodic conditions for the left and right sides of the domain while for the top and bottom Full Slip conditions are used. see Figure(34)

- Bottom

$$\begin{aligned} u[1, :] &= u[3, :] & v[1, :] &= 0 \\ u[2, :] &= u[3, :] & v[2, :] &= 0 \end{aligned}$$

- Top

$$\begin{aligned} u[Ny + 3, :] &= u[Ny + 2, :] & v[Ny + 3, :] &= 0 \\ u[Ny + 4, :] &= u[Ny + 2, :] & v[Ny + 4, :] &= 0 \\ . & & v[Ny + 2, :] &= 0 \end{aligned}$$

- Left

$$\begin{aligned} u[:, 2] &= u[:, Nx + 2] & v[:, 2] &= v[:, Nx + 2] \\ u[:, 1] &= u[:, Nx + 1] & v[:, 1] &= v[:, Nx + 1] \end{aligned}$$

- Right

$$\begin{aligned} u[:, Nx + 3] &= u[:, 3] & v[:, Nx + 3] &= v[:, 3] \\ u[:, Nx + 4] &= u[:, 4] & v[:, Nx + 4] &= v[:, 4] \end{aligned}$$

In the same way as in the previous boundary conditions, eta must be setted to a 0 value in the top and bottom of the halos while for the left and right periodic conditions are used.

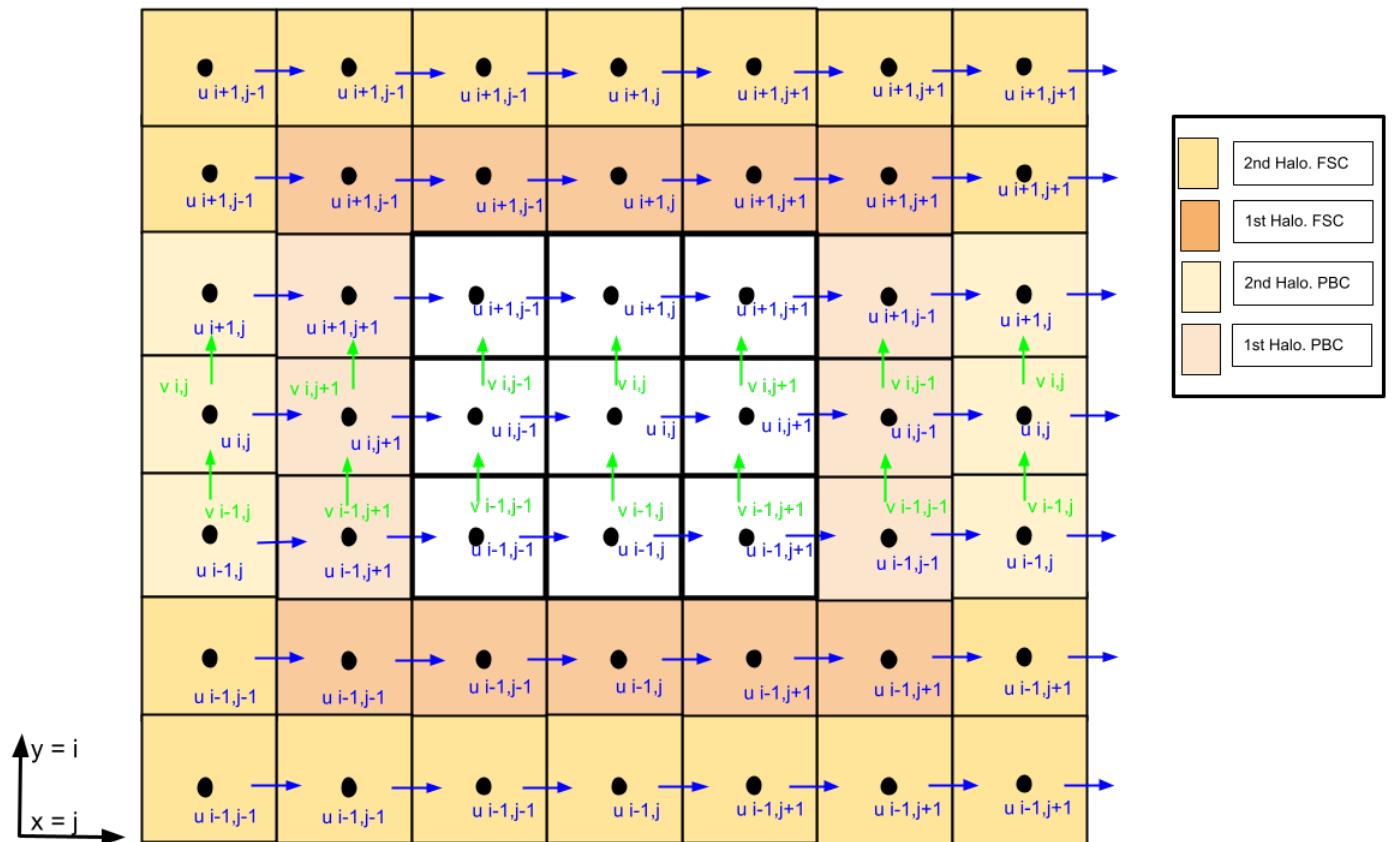


FIGURE 34: Canal Boundary conditions

9.3.2 Perturbation introduction

Perturbation explained in the past section is based on the Cartesian domain parameters (section (9.2.3)). The same expression has to be changed in order to adapt to the elliptical coordinates. This is:

$$\mathcal{S}(\theta, \varphi) = A \exp \left\{ - \left[\left(\frac{\theta - \theta_0}{a_0} \right)^2 + \left(\frac{\varphi - \varphi_0}{b_0} \right)^2 \right]^n \right\} \quad (9.3.11)$$

Where a_0 and b_0 correspond to major and minor semi axis respectively, and n is a constant used to dampen the disturbance.

In order to define the maximum radius of the perturbation in the elliptical domain, the following approximation is used:

$$r_{i,j} = \frac{R_E + R_P}{2} \arccos (\sin \varphi_0 \sin \varphi_{i,j} + \cos \varphi_0 \cos \varphi_{i,j} \cos(\theta_{i,j} - \theta_0)) \quad (9.3.12)$$

This expression defines the radius value between a point located at $P(\varphi, \theta)$ and the centre of the perturbation $P(\varphi_0, \theta_0)$

	A	φ_0	θ_0	a_0	b_0	n
Domain: Elliptical	1500 m	-19.62 deg	7.35 deg	0.14	0.077	3

TABLE 6: Initial parameters used to introduce perturbation into the Elliptical domain

g_0	24.79 m/s^2	θ_0	-30 deg
Ω	$1.76 \cdot 10^{-4} \text{ rad/s}$	θ_1	40 deg
R_e	71492 km	φ_0	-30
R_p	66854 km	φ_1	0

TABLE 7: Elliptical domain, physical simulation parameters

9.3.3 Simulation results

The results shown below have been obtained using the parameters described in the Table(6) as initial variables with a deep's layer of $D = 1000m$.

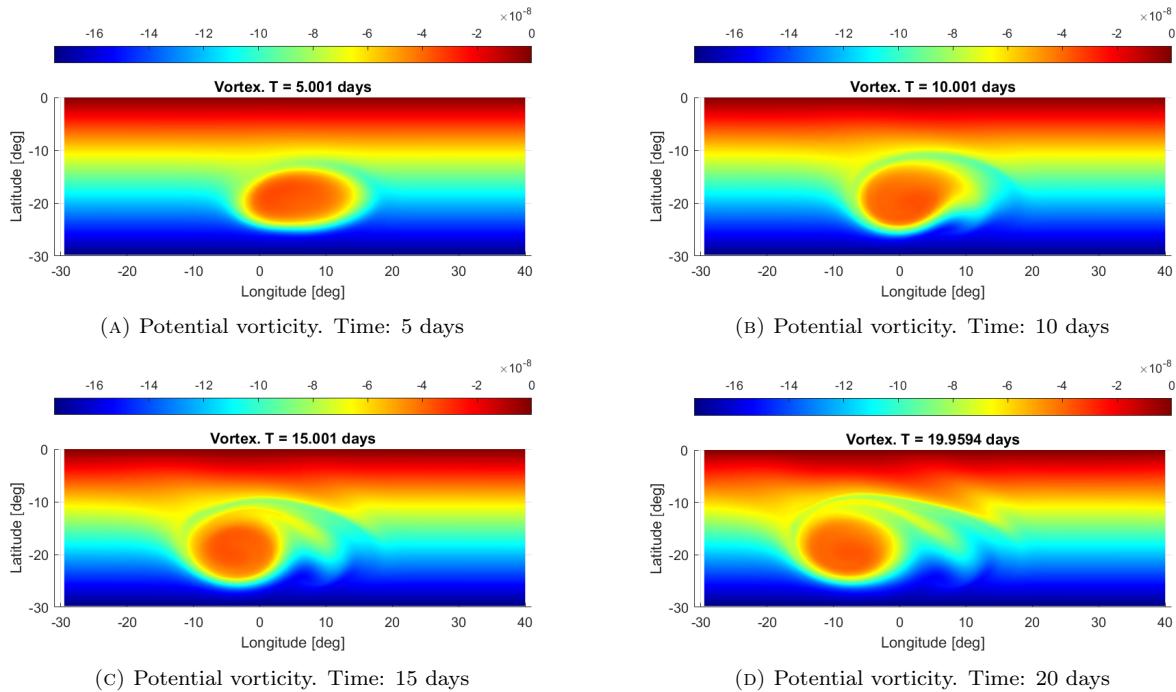


FIGURE 35: vortex at different time steps. Elliptical domain. Time simulation: 20 days

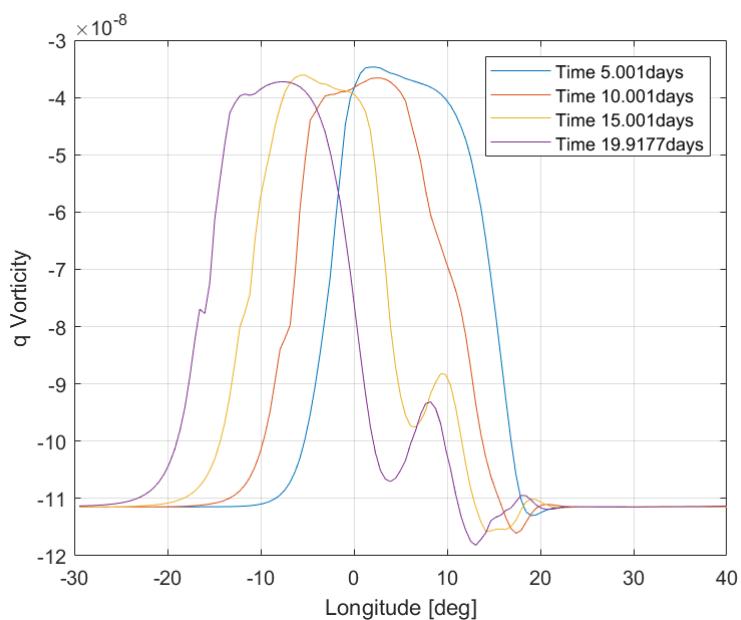


FIGURE 36: Potential vorticity over time. Elliptical domain

Figure(35) shows the vortex evolution over time into an Elliptical domain during 20 days. This characteristic domain is the cause of the movement shown in the graphs.

In contrast to the results displayed in Section(9.2), rotation can be appreciated in these graphics without the need of plotting velocity components.

Figure(36) shows the potential vorticity along the $x-axis$ at different time steps. In this case, its behaviour is irregular and some instabilities and displacements, caused by the elliptical mesh, can be seen. Despite of this oscillations, it is clearly seen that the potential vorticity $\Pi = q$ is conserved over time.

Another important aspect to be understood is how the values of potential vorticity change depending on the latitude of the simulation. If the same simulation shown in Figure(35) is carried out at positive values of latitude (φ), vortex aspect will change into a blue colour aspect. This phenomena is caused by the direction of the Coriolis parameter (\vec{f}) and the absolute vorticity vector ($\vec{\omega} = \zeta$) at different planetary latitudes. These directions and their corresponding vortex colours can be seen in Figure(37).

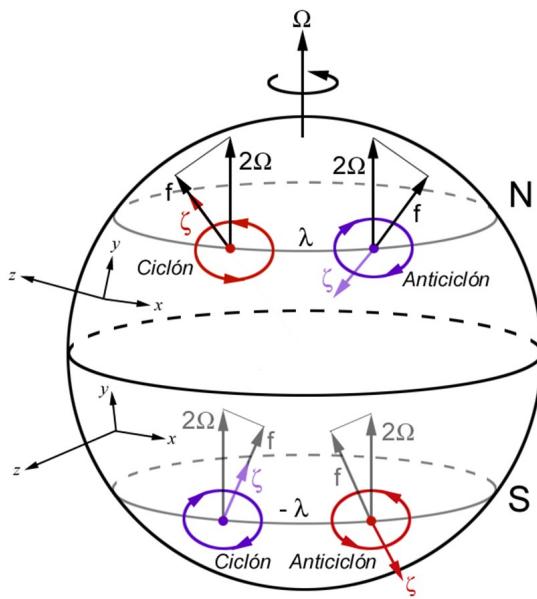


FIGURE 37: Vortex colour at different latitudes. Image created by Dr. Enrique Garcia Melendo.

Once all these experiments are carried out, the final step is to introduce Zonal winds into the model. This, in conjunction with the Jupiter physical parameters, allows to implement the Great Red Spot.

9.4 Great Red Spot

Brief introduction to the global atmospheric motions of the giant planets

The giant planets of the Solar System, Jupiter, Saturn, Uranus, and Neptune, are also called gas giants because they lack a solid surface, at least in the sense we have on Earth. Wind speeds for these planets are measured with respect to the rotation period of the magnetic field supposedly rooted to the deep interior of the planet, and therefore representing the true rotation period. The way the winds are retrieved is by tracking the motion of cloud features visible on the upper cloud deck and comparing them to the magnetic field's rotation period. At this cloud level, the winds of four planets form a massive and persistent motion parallel to longitude circles also called as *zonal winds*.

Observations show that the global circulation of the gas giants is dominated by a system alternated zonal jet streams, i.e., directed along circles parallel to the equator (see Figure(38) for the Jupiter and Saturn cases). They can therefore be described with a high degree of accuracy a function of latitude $u = U(\varphi)$, where U and u are velocities in the east-west direction.

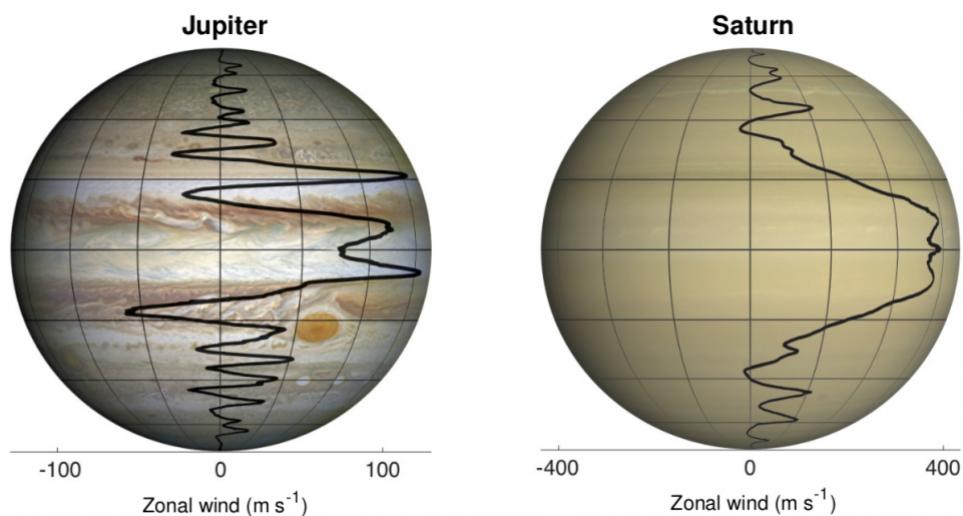


FIGURE 38: Jupiter (left) and Saturn (right) zonal wind system superimposed on the visible cloud deck.
From Kaspi et al. 2019, <https://arxiv.org/pdf/1908.09613.pdf>.

One of the most notable features of the zonal wind system on Jupiter and Saturn is the existence of a very broad equatorial jet stream blowing eastwards, whose maximum velocities

reach 150ms^{-1} on Jupiter and 500ms^{-1} on Saturn (about 2/3 of the propagation speed of sound in the $H_2 - H_e$ atmosphere of Saturn). In contrast, Uranus and Neptune show only an eastward-directed jet stream at mid-latitudes, and another very broad, westward-directed jet stream centred at the equator showing velocities of the order of up to -400ms^{-1} in the case of Neptune, (see Figure(39)).

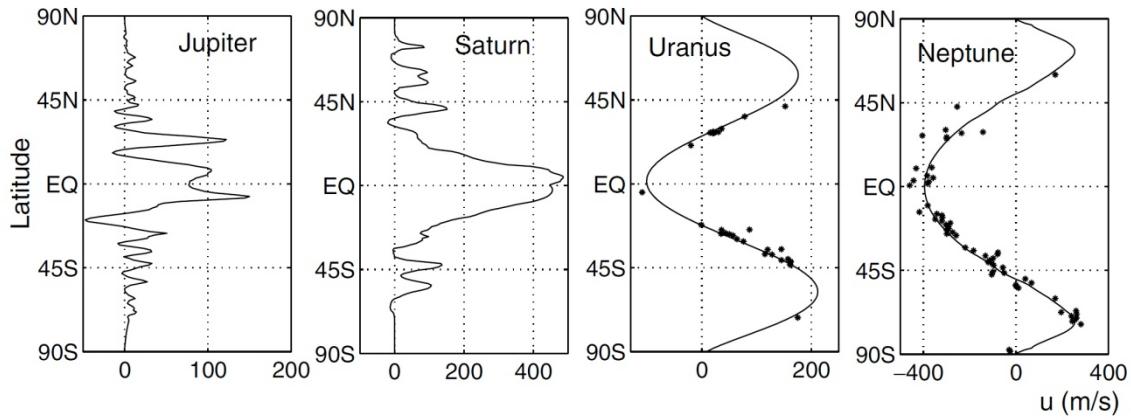


FIGURE 39: Comparison of the four zonal wind systems on the giant planets of the Solar System.
Sukoriansky et al. (2002).

Zonal winds show a very strong interaction with meteorological phenomena on these planets such as vortices, storms, etc. Therefore it is absolutely necessary to include zonal winds in the SW model if realistic simulations are to be performed.

9.4.1 Perturbation introduction

In this case, the Gaussian function has the same expression as in the elliptical vortex simulation - i.e Equation(9.3.11) - however, its parameters are different and should replicate the conditions of the the Great Red Spot (*GRS*). In order to do that Geostrophic equilibrium is be used. This phenomenon is found at large scale movements in planetary atmospheres where Coriolis force is balanced with the pressure gradient. This force the fluid to rotate instead of spread around the atmosphere. A deeper explanation of this phenomena is found in [3].

$$\begin{aligned} fv &= g \frac{\partial \eta}{\partial x} \\ fu &= -g \frac{\partial \eta}{\partial y} \end{aligned} \quad (9.4.1)$$

It is possible to impose this condition on the velocity in the fluid, giving the following expression:

$$\begin{aligned} u &= -\frac{g}{f} \frac{\partial \eta}{\partial y} \\ v &= \frac{g}{f} \frac{\partial \eta}{\partial x} \end{aligned} \quad (9.4.2)$$

Relation between $(\partial \eta / \partial x, \partial \eta / \partial y) \rightarrow (\partial \eta / \partial \theta, \partial \eta / \partial \varphi)$ is established using the "chain rule".

$$\begin{aligned} \frac{\partial \eta}{\partial x} &= \frac{\partial \eta}{\partial \theta} \frac{\partial \theta}{\partial x} = \frac{1}{r_Z} \frac{\partial \eta}{\partial \theta} \\ \frac{\partial \eta}{\partial y} &= \frac{\partial \eta}{\partial \varphi} \frac{\partial \varphi}{\partial y} = \frac{1}{r_M} \frac{\partial \eta}{\partial \varphi} \end{aligned} \quad (9.4.3)$$

After evaluate each derivative and introduce it into Equation(9.4.2), velocity components: u and v are defined.

$$u = \frac{2ng}{r_M f b_0^2} (\varphi - \varphi_0) \Psi^{n-1} \eta(\theta, \varphi) \quad (9.4.4)$$

$$v = \frac{-2ng}{r_Z f a_0^2} (\theta - \theta_0) \Psi^{n-1} \eta(\theta, \varphi) \quad (9.4.5)$$

Where Ψ is defined such as:

$$\Psi = \left(\left[\frac{\theta - \theta_0}{a_0} \right]^2 + \left[\frac{\varphi - \varphi_0}{b_0} \right]^2 \right)$$

Geostrophic balance allows to introduce the perturbation in a different way: it is introduced at the initial time (t_0) with its maximum altitude representing the *GRS* since the initial

time. However, perturbation velocities' fields have to be introduced at the beginning of the simulation (t_0) as it is expressed in Equation(9.4.4) and Equation(9.4.5).

Finally, geometrical parameters for the *GRS* are found. To do so, expression for the maximum tangential velocity is computed, this allows to define the vortex amplitude (A). The point where this condition is fulfilled is defined as:

$$\varphi|_{umax} = \varphi_0 \pm b_0 \left[1 - \frac{1}{2n}^{\frac{1}{2n}} \right] \quad (9.4.6)$$

$$\theta|_{vmax} = \theta_0 \pm a_0 \left[1 - \frac{1}{2n}^{\frac{1}{2n}} \right] \quad (9.4.7)$$

These equations allow to compute the value of the vortex's semi-axis a_0 and b_0 .

Then, maximum tangential velocity (u_{max}) is computed at the latitude and longitude where it is maximum:

$$u_{max} = \frac{2ng}{r_M f b_0} \left(1 - \frac{1}{2n} \right)^{\frac{2n-1}{2n}} A \exp \left[- \left(-\frac{1}{2n} \right) \right] \quad (9.4.8)$$

From this expression, the maximum amplitude of the vortex is found.

$$A = \frac{r_M f b_0 u_{max}}{2ng} \left(1 - \frac{1}{2n} \right)^{\frac{1-2n}{2n}} \exp \left[1 - \frac{1}{2n} \right] \quad (9.4.9)$$

The following table expresses the different constraints to simulate the *GRS* perturbation.

n	3	g	24.79 m/s ²
φ_0	-19.62 deg	Ω	$1.76 \cdot 10^{-4}$ rad/s
θ_0	7.35 deg	R_e	71 492 km
$\varphi _{umax}$	-15.30 deg	R_p	66 854 km
$\theta _{vmax}$	15.04 deg	u_{max}	-126.5 m/s

TABLE 8: Geometrical and physical variables used to implement GRS perturbation

9.4.2 Simulation results

The evolution of the *GRS* over 70 days is shown below. It is modelled using the parameters described in Table(8) with $\Delta t = 30s$

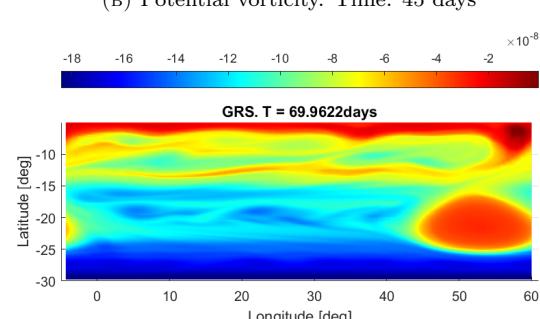
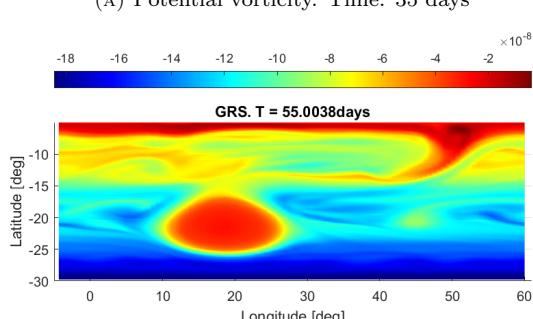
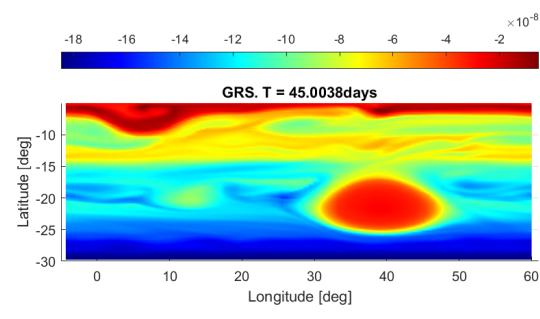
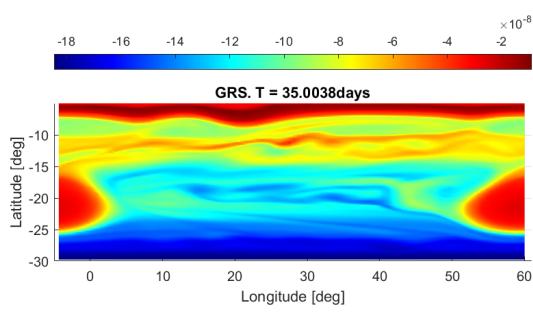
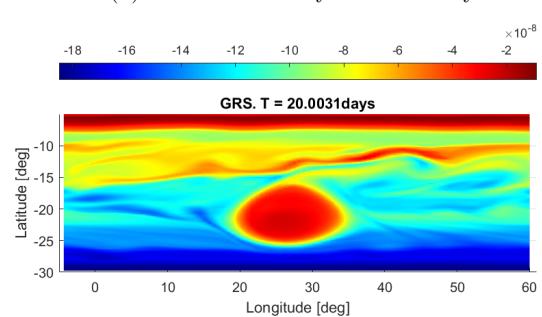
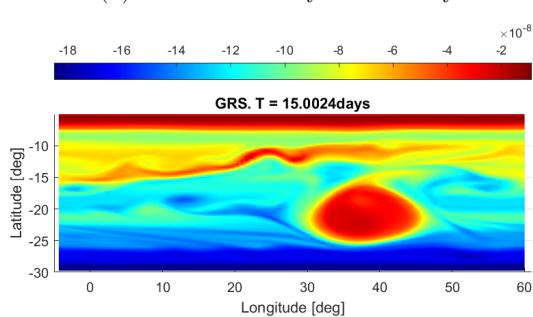
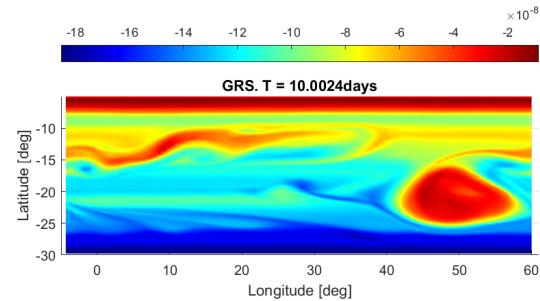
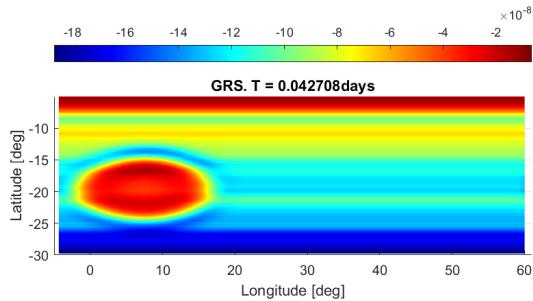


FIGURE 41: Great Red Spot simulation using geostrophic balance. Jupiter planet parameters.
 $N_x = 100$, $N_y = 100$

Simulations shown in Figure(41) represents the Great Red Spot during 70 days in a Canal condition. Initial perturbation is introduced using Geostrophic balance which give a successful result. However, it takes a long time to stabilise (around 20 days) which makes it necessary to extend the simulation time in order to be able to appreciate the phenomenon without distortions.

Once the red spot has stabilised, it is possible to extract data from the simulation for comparison with real values collected by spacecrafts on different missions.

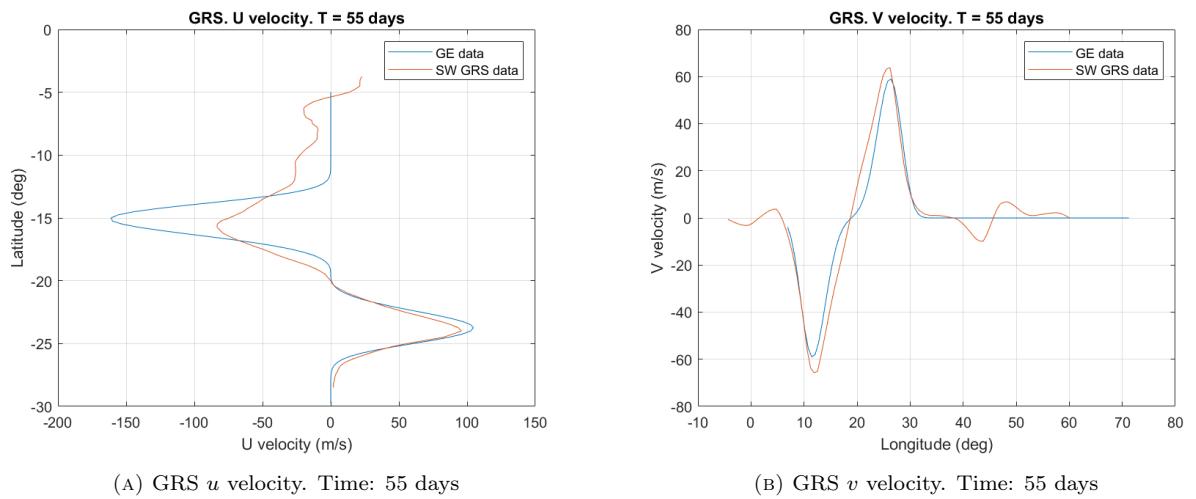
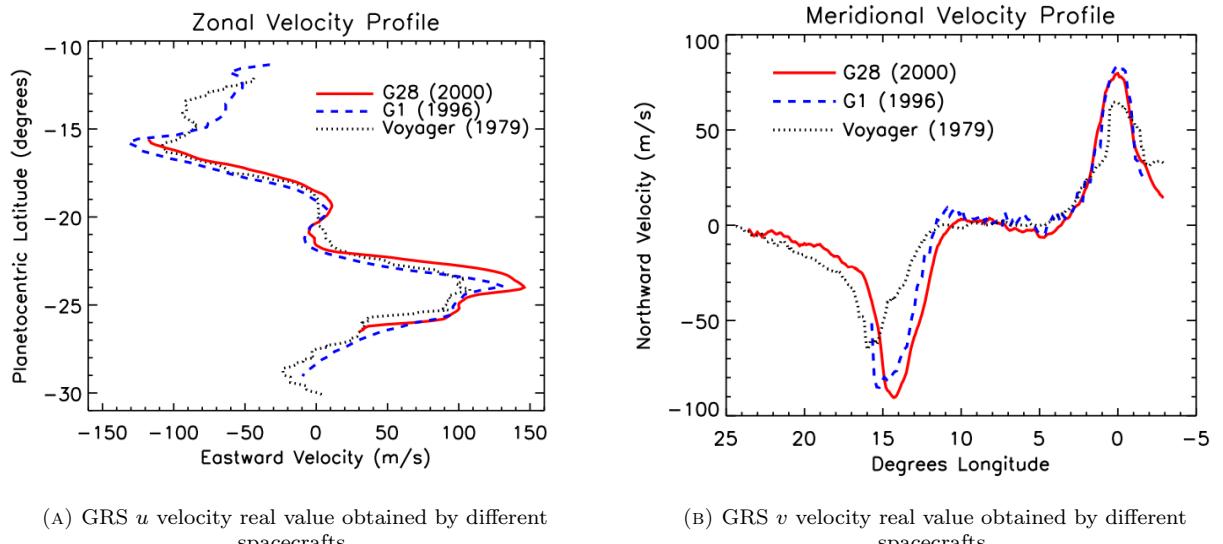


FIGURE 42: GRS simulation using geostrophic balance. u and v velocity obtained with the Geostrophic Equilibrium (GE) and Shallow Water (SW) model



(A) GRS u velocity real value obtained by different

spacecrafts.

(B) GRS v velocity real value obtained by different

spacecrafts.

FIGURE 43: GRS u and v velocity value obtained by different missions.
(Velocity and Vorticity Measurements of Jupiter's Great Red Spot Using Automated Cloud Feature Tracking [5])

Comparing analytical velocity values shown in Figure(42) with those plotted in Figure(43) it can be seen that the introduction of the Geostrophic Equilibrium vortex is a good approximation because it follows the same trend as the real *GRS* velocity. Geostrophic Equilibrium data is obtained from Equation(9.4.5) and (9.4.4). Some similarities can be observed at the peaks values and where are they positioned in the domain, which allows to consider the Geostrophic Equilibrium as a valid approximation.

On the other hand, *SW* model velocities follow a correct trend near the peaks, stabilising outside the red spot. The u component of the velocity is more damped because it is more affected by the zonal winds, unlike the v -component which follows the Geostrophic trend accurately.

Since the values obtained do not closely follow the behaviour of the real data, it would be necessary to refine the model by several improvements: addition of hyperviscosity and different layers or the introduction of thermodynamic effects.

Another way to increase the accuracy would be to decrease the time-step (Δt) and increase the number of grid points (N_x, N_y). This would result in less energy dissipation but would greatly increase the simulation time and processing power required.

All these features are beyond the scope of this study and it is up to the reader to continue the investigation.

Finally, some graphical results are added, comparing real photos of the *GRS* with simulation frames. These comparisons demonstrate how the model is able to replicate the flow moments in the red spot. However, some phenomena are slightly appreciated due to the low resolution of the simulation.

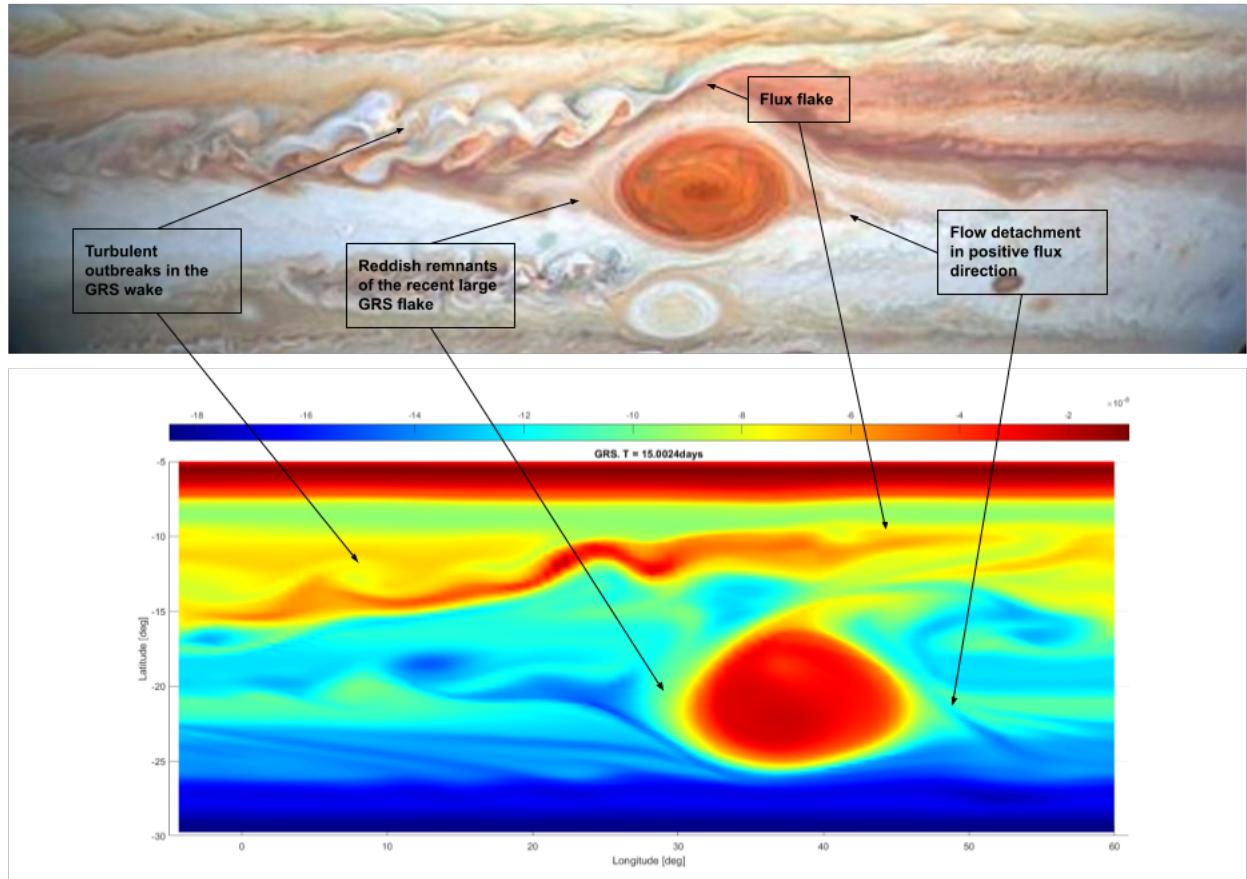


FIGURE 44: GRS image from: <https://hubblesite.org> compared with *SW* simulation.
 $T = 55$ days, $\Delta t = 30s$, $N_x = 100$, $N_y = 100$

9.5 Saturn storms

Using the same concepts applied in the Great Red Spot simulation [9.4] and the Zonal Winds posted on the Appendix(12.1), it is possible to use the *Shallow Water* model developed in this study to represent the storms created at the high latitudes of the Saturn planet.

Geostrophic Equilibrium can not be used to implement the perturbation, instead of applying this strategy, vortex will be introduced using the same equation and method as the one presented in the Section(9.3.2). However after each time-step, during the whole time of simulation, flux mass has to be injected in order to preserve the spot size. This flux mass (Q) is defined such as the mass volume per unit of time $Q[m^3/s]$ and the amplitude of the vortex at each time step is computed as: $A = Q \cdot \Delta t / (\pi ab)$

Where a and b are the major and minor axis of the ellipse formed by the projection of the Gaussian on the domain.

Another important aspect to take into account is that Zonal winds will move the spot over time, so the centre of the vortex has to be computed after each time step in order to guess where to do the flux mass injection. Considering small time steps $\Delta t \approx 5s$, spot centre is computed as: $\theta_0^{n+1} = \theta_0^n + V_{spot} \cdot \Delta t$.

Where V_{spot} is a constant displacement velocity and θ_0 is the longitude spot centre.

The following table summarises simulation parameters.

n	1	Ω	$1.62 \cdot 10^{-4}$ rad/s
φ_0	38.5 deg	R_E	60268 Km
θ_0	22.5 deg	R_P	54364 Km
$Q [m^3/s]$	$1 \cdot 10^{11}$	V_{spot}	-21 m/s
θ	100	R_{max}	1 deg

TABLE 9: Saturn storm simulation parameters.

9.5.1 Simulation results

In this section the results for the Saturn storm simulation are presented. This simulation is carried out using $\Delta t = 30s$ and a deep layer of $D = 500m$

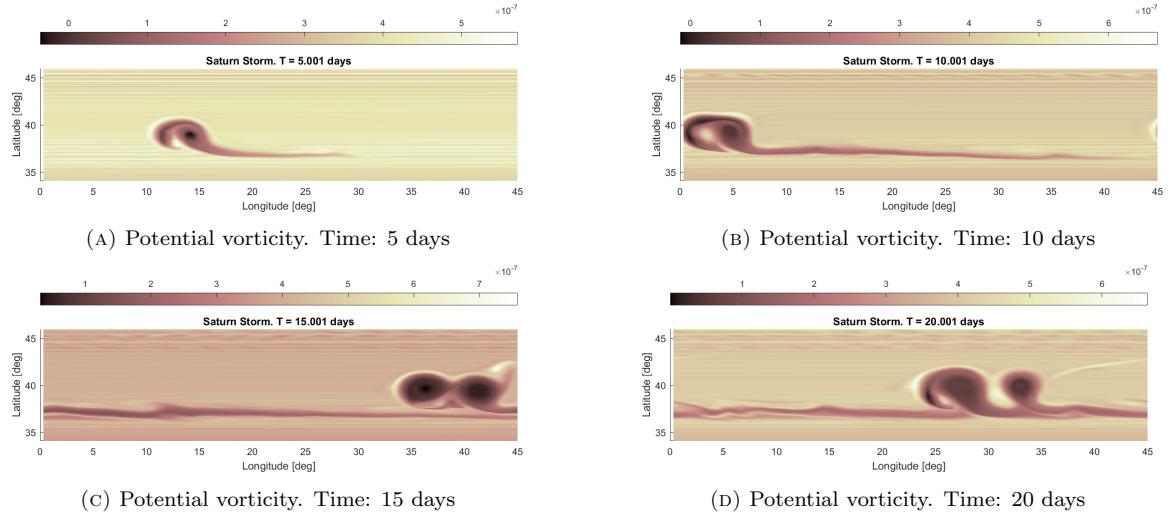


FIGURE 45: Saturn storm simulation. $\Delta t = 30s$, $D = 500m$



FIGURE 46: Real image of the Saturn storm in 2010. At this time the storm was still growing

It is important to notice that this simulation is very sensitive to the initial parameter, so that a small change in the initial conditions can cause the simulation to become unstable. In Figure(45d) can be seen how the storm is starting to become stable creating the same shape as the one seen in Figure(46).

10 Conclusions and Future developments

Throughout this study, the hypotheses, mathematical formulations and numerical implementations of the Shallow Water model are developed, allowing the reader to understand and replicate those concepts independently introducing himself/herself to computational fluid dynamics through practical and real-world examples.

As a test of the usefulness and validity of the model, real data on Jupiter's Great Red Spot (GRS) are compared with those obtained in the simulation. This comparison shows a mismatch in the horizontal component of the velocity due to the series of assumptions made. In order to increase the accuracy it would be necessary to include new factors such as thermodynamic effects, multi-layering or the extension of a three-dimensional model, concepts that can be implemented as part of a future study.

Despite the apparent simplicity of the model it is possible to see how this simplified version allows modelling the behaviour of the red spot, creating the same movements in the flow (wakes, flow ejections and turbulent zones).

Taking advantage of the available zonal wind data, a final simulation of the storms produced on Saturn is added. This further experiment again demonstrates to the reader the versatility and potential of the developed software. Such storms develop slowly and require more computing power to obtain accurate results, but the computing power of a mid-range computer is more than sufficient to obtain a satisfactory resolution.

Finally, the results obtained together with the frames shown in both experiments demonstrate how this model is suitable for applying complex concepts of computational fluid dynamics, obtaining valid results with a certain degree of accuracy, thus fulfilling the objectives of this project.

11 Budget

At the end of every engineering project it is usually common to introduce a budget explaining every task of the project, how much time does it take to solve it and the total price of the project.

The price is calculated on the basis of the average salary paid by a company to a Junior engineer, which is around 15€/h, in conjunction with the cost of production. In this case, this cost is refereed to the energy invested on computation. It is computed as:

$$\text{Energy Cost} = (\text{Energy Price}) \cdot (\text{Laptop energy consumption}) \cdot (\text{Time of simulation})$$

Energy price is established at: 0.2517 €/Kwh and Laptop energy consumption is 120W.

The project is divided into three distinct tasks: researching and project writing, software development and debugging and simulation process. The following table summarises how much time is invested for every task, its corresponding unitary cost and the total cost of the project.

Task	Amount (h)	Unitary cost (€/h)	Cost (€)
Documentation research	130	15	1950
Documenetation drafting	200	15	3000
Software development*	371	15	5565
Computation resources	236	0.03	7.00
		Total Cost	10522.00 €

TABLE 10: Project budget

* Software development is broken down on the following page.

TASK	FEBRUARY				MARCH				APRIL				MAY				JUNE				Time Cost (h)
	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	
Code Implementation	Advection Terms P1 and P2 implementation																				14
	Terms P1h and Pp implementation																				4
	Terms Validation																				80
	Adam Bashforth scheme implementation																				3
	Term Validation																				36
	Wave simulation																				20
	Coriolis Implementation																				8
	Storm perturbation implementation																				22
	Bug Fixes																				140
	Red spot simulation																				44
																				TOTAL (h)	371

References

- [1] Jhon Anderson. *Computational fluid dynamics, second edition*. Vol. 9780521769. 2010, pp. 1–1034. ISBN: 9780511780066. DOI: 10.1017/CBO9780511780066.
- [2] Marc Andrés Carcasona. “Study: Simulation of Planetary Atmospheres”. In: (2020). URL: <http://hdl.handle.net/2117/331436>.
- [3] David G. Andrews. *An introduction to Atmospheric Physics*. Ed. by Cambridge. 2nd Editio. 1390, pp. 99–117. ISBN: 9780521872201.
- [4] *APPLY master program*. URL: <https://apply-project.eu>.
- [5] David S. Choi et al. “Velocity and vorticity measurements of Jupiter’s Great Red Spot using automated cloud feature tracking”. In: *Icarus* 188.1 (2007), pp. 35–46. ISSN: 00191035. DOI: 10.1016/j.icarus.2006.10.037. arXiv: 1301.6119.
- [6] Benoit Cushman-Roisin and Jean-Marie Beckers. “Introduction to Geophysical Fluid Dynamics. Physical and Numerical Aspects”. In: *Analysis* 101 (2010), p. 786. ISSN: 00746142.
- [7] James Holton. *An Introduction to Dynamic Meteorology, Volume 88 - 4th Edition*. 2004. (Visited on 06/16/2021).
- [8] Patrick Knupp Kambiz Safari. *Code Verification by the Method of Manufactured Solutions*. Ed. by Sandia National Laboratories. 2000.
- [9] Jochen kampf. *Ocean modelling for beginners: Using open-source software*. 2009, pp. 1–175. ISBN: 9783642008191. DOI: 10.1007/978-3-642-00820-7.
- [10] Fermi M.I. *Pemanfaatan Computational Fluid Dynamics (Cfd)*. Vol. 2. 2. 2016. ISBN: 9780131274983. DOI: 10.24114/eb.v2i2.4393.
- [11] André Marten. “Study of planetary atmospheres”. In: (1996), pp. 308–315. DOI: 10.1007/978-3-540-69999-6_45. URL: <http://hdl.handle.net/2117/331436>.
- [12] Sweby P.K. “High resolution schemes using flux limiters for hyperbolic conservation laws”. In: *SIAM Journal on Numerical Analysis* 21.5 (1984), pp. 995–1011.
- [13] Joseph Pedlosky. *[Joseph_Pedlosky]-Geophysical_Fluid_Dynamics(z-lib.org)-1.pdf*. 1987.
- [14] Arnau Sabatés. “Shallow Water Model Code for Applications to Atmosphere Dynamics”. In: (2018), p. 73.
- [15] Hilary Weller. “Numerics : The analysis and implementation of numerical methods for solving differential equations”. In: (2015).



- [16] Vladimir Zeitlin. *Geophysical Fluid Dynamics: Understanding almost everything with rotating shallow water models: Amazon.es: Zeitlin, Vladimir: Libros en idiomas extranjeros*. 2018. (Visited on 06/16/2021).

APENDIX.

Jupiter Zonal Winds.

TABLE 11: Jupiter Zonal Winds data

φ	u(m/s)								
-81.58	-0.508	-73.42	-3.012	-71.37	-2.716	-69.52	-3.074	-67.7	22.585
-78.93	-0.341	-73.38	-2.882	-71.32	-2.955	-69.47	-2.733	-67.64	23.763
-78.89	-0.189	-73.31	-2.697	-71.28	-3.165	-69.42	-2.375	-67.6	24.978
-78.56	-0.053	-73.26	-2.516	-71.23	-3.341	-69.37	-1.983	-67.56	26.22
-78.05	0.067	-73.2	-2.303	-71.17	-3.529	-69.33	-1.544	-67.5	27.446
-78.03	0.171	-73.16	-1.977	-71.12	-3.803	-69.28	-1.176	-67.46	28.811
-77.28	0.259	-73.13	-1.902	-71.08	-4.119	-69.23	-0.592	-67.41	30.065
-77.22	0.332	-73.07	-1.708	-71.03	-4.328	-69.19	-0.067	-67.37	31.805
-77.16	0.388	-73.03	-1.645	-71	-4.651	-69.14	0.599	-67.32	33.112
-76.56	0.429	-72.97	-1.573	-70.92	-4.92	-69.1	1.32	-67.27	34.309
-76.39	0.454	-72.94	-1.678	-70.87	-5.189	-69.06	2.134	-67.22	35.262
-76.33	0.463	-72.9	-1.687	-70.82	-5.486	-69	2.756	-67.18	36.542
-76.27	0.457	-72.84	-1.802	-70.78	-5.684	-68.96	3.305	-67.14	37.703
-76.21	0.434	-72.79	-2.053	-70.74	-5.739	-68.91	3.947	-67.09	38.747
-75.13	0.396	-72.75	-2.355	-70.69	-5.711	-68.86	4.833	-67.05	39.727
-75.03	0.342	-72.71	-2.444	-70.63	-5.531	-68.83	5.599	-67	40.232
-74.95	0.272	-72.66	-2.429	-70.6	-5.164	-68.78	6.041	-66.96	40.724
-74.91	0.186	-72.61	-2.173	-70.54	-5.237	-68.74	6.767	-66.91	40.881
-74.87	0.085	-72.59	-1.945	-70.5	-5.421	-68.69	7.453	-66.86	40.789
-74.83	-0.033	-72.54	-1.89	-70.45	-5.475	-68.64	8.191	-66.82	40.479
-74.78	-0.166	-72.49	-1.779	-70.42	-5.331	-68.59	8.657	-66.77	40.059
-74.74	-0.167	-72.45	-1.574	-70.37	-5.382	-68.55	9.359	-66.72	39.302
-74.69	-0.661	-72.39	-1.563	-70.33	-5.319	-68.5	9.959	-66.68	38.446
-74.64	-1.113	-72.36	-1.492	-70.27	-5.201	-68.46	10.464	-66.64	37.471
-74.6	-1.538	-72.31	-1.403	-70.24	-5.181	-68.41	10.958	-66.59	36.266
-74.15	-2.044	-72.27	-1.438	-70.19	-5.184	-68.37	11.492	-66.55	35.073
-74.04	-2.447	-72.23	-1.454	-70.14	-5.112	-68.33	11.888	-66.49	34.038
-74.02	-2.849	-72.17	-1.515	-70.1	-4.991	-68.28	12.354	-66.45	32.842
-73.98	-3.298	-72.13	-1.501	-70.05	-4.985	-68.23	12.897	-66.4	31.638
-73.93	-3.396	-72.04	-1.56	-70	-4.936	-68.19	13.536	-66.36	30.403
-73.89	-3.447	-71.97	-1.554	-69.95	-4.91	-68.15	14.24	-66.31	29.226
-73.84	-3.678	-71.96	-1.624	-69.91	-4.808	-68.1	14.996	-66.25	27.983
-73.8	-3.739	-71.84	-1.669	-69.87	-4.697	-68.05	15.864	-66.22	26.594
-73.75	-3.513	-71.81	-1.787	-69.82	-4.658	-68.01	16.777	-66.18	25.337
-73.7	-3.38	-71.78	-1.879	-69.77	-4.328	-67.96	17.324	-66.13	24.191
-73.64	-3.444	-71.73	-1.884	-69.73	-4.229	-67.91	18.132	-66.09	23.068
-73.61	-3.213	-71.69	-2.11	-69.69	-3.862	-67.87	18.778	-66.03	22.073
-73.55	-3.214	-71.53	-2.199	-69.65	-3.708	-67.83	19.583	-65.99	21.187
-73.53	-3.172	-71.46	-2.329	-69.6	-3.594	-67.77	20.528	-65.95	20.367
-73.47	-3.118	-71.42	-2.445	-69.56	-3.307	-67.73	21.439	-65.9	19.584

TABLE 12: Continue I. Jupiter Zonal Winds data

φ	u(m/s)								
-65.85	18.831	-64	5.078	-62.14	12.725	-60.27	10.357	-58.38	-3.597
-65.81	18.274	-63.96	4.916	-62.09	13.389	-60.23	9.618	-58.33	-3.743
-65.76	17.798	-63.9	4.772	-62.04	13.963	-60.17	8.992	-58.28	-3.802
-65.72	17.46	-63.86	4.677	-62	14.556	-60.13	8.408	-58.24	-3.844
-65.67	17.191	-63.83	4.66	-61.96	15.155	-60.08	7.905	-58.19	-3.885
-65.62	17.134	-63.77	4.682	-61.91	15.724	-60.04	7.436	-58.14	-3.926
-65.58	17.343	-63.73	4.697	-61.86	16.232	-59.99	6.99	-58.1	-3.916
-65.52	17.054	-63.68	4.679	-61.82	16.802	-59.95	6.442	-58.05	-3.896
-65.48	17.201	-63.63	4.668	-61.77	17.347	-59.89	5.9	-58	-3.894
-65.43	17.295	-63.59	4.62	-61.72	17.824	-59.85	5.401	-57.95	-3.884
-65.39	17.492	-63.53	4.641	-61.68	18.236	-59.8	4.891	-57.9	-3.878
-65.33	17.395	-63.5	4.585	-61.64	18.609	-59.75	4.373	-57.85	-3.871
-65.3	17.251	-63.44	4.511	-61.59	18.85	-59.7	4.015	-57.81	-3.904
-65.26	17.034	-63.39	4.428	-61.54	19.073	-59.66	3.669	-57.76	-3.915
-65.21	16.356	-63.36	4.397	-61.49	19.352	-59.61	3.189	-57.71	-3.912
-65.16	15.914	-63.31	4.399	-61.44	19.564	-59.56	2.729	-57.66	-3.906
-65.12	15.26	-63.26	4.476	-61.39	19.651	-59.51	2.372	-57.62	-3.889
-65.07	14.567	-63.22	4.512	-61.36	19.804	-59.47	1.939	-57.58	-3.89
-65.02	13.922	-63.17	4.635	-61.32	19.831	-59.42	1.599	-57.53	-3.966
-64.97	13.285	-63.13	4.775	-61.26	19.744	-59.38	1.356	-57.48	-4.083
-64.93	12.636	-63.08	4.952	-61.21	19.671	-59.33	1.066	-57.43	-4.116
-64.88	12.047	-63.03	5.162	-61.15	19.523	-59.28	0.864	-57.39	-4.18
-64.84	11.493	-62.99	5.437	-61.11	19.275	-59.23	0.737	-57.33	-4.293
-64.8	10.898	-62.93	5.615	-61.07	19.202	-59.18	0.569	-57.29	-4.384
-64.74	10.413	-62.89	5.937	-61.03	18.973	-59.15	0.381	-57.24	-4.506
-64.69	9.883	-62.84	6.335	-60.98	18.739	-59.1	0.135	-57.19	-4.606
-64.65	9.376	-62.79	6.707	-60.93	18.414	-59.05	-0.048	-57.14	-4.65
-64.6	8.954	-62.74	7.119	-60.88	18.031	-58.99	-0.289	-57.09	-4.727
-64.56	8.553	-62.69	7.5	-60.84	17.65	-58.94	-0.571	-57.05	-4.808
-64.51	8.206	-62.65	7.999	-60.79	17.141	-58.89	-0.857	-57.01	-4.883
-64.46	7.889	-62.61	8.383	-60.74	16.547	-58.85	-1.101	-56.95	-4.965
-64.42	7.555	-62.57	8.782	-60.7	15.989	-58.81	-1.421	-56.91	-5.006
-64.38	7.234	-62.51	9.137	-60.64	15.412	-58.77	-1.726	-56.87	-5.033
-64.33	6.903	-62.47	9.546	-60.59	14.744	-58.72	-2.019	-56.82	-5.083
-64.28	6.627	-62.42	9.913	-60.56	14.136	-58.67	-2.32	-56.77	-5.094
-64.24	6.346	-62.38	10.327	-60.5	13.467	-58.61	-2.603	-56.72	-5.095
-64.19	6.034	-62.33	10.781	-60.46	12.777	-58.58	-2.842	-56.66	-5.078
-64.14	5.739	-62.29	11.184	-60.41	12.135	-58.53	-3.105	-56.61	-5.062
-64.1	5.479	-62.23	11.556	-60.37	11.614	-58.48	-3.292	-56.57	-4.971
-64.05	5.228	-62.18	12.092	-60.32	11.011	-58.43	-3.437	-56.52	-5.017

TABLE 13: Continue II.Jupiter Zonal Winds data

φ	u(m/s)	φ	u(m/s)	φ	u(m/s)	φ	u(m/s)	φ	u(m/s)
-56.48	-5.075	-54.55	6.87	-52.61	39.515	-50.66	5.929	-48.69	-13.006
-56.42	-5.034	-54.51	7.842	-52.57	39.255	-50.61	5.42	-48.64	-13.539
-56.38	-5.035	-54.45	8.836	-52.52	39.064	-50.56	4.912	-48.59	-14.024
-56.33	-5.04	-54.41	9.873	-52.47	38.858	-50.51	4.443	-48.53	-14.408
-56.28	-5.029	-54.36	11.097	-52.43	38.544	-50.46	3.873	-48.48	-14.81
-56.23	-5.057	-54.31	12.397	-52.38	38.011	-50.41	3.443	-48.43	-15.117
-56.19	-4.99	-54.26	13.546	-52.33	37.48	-50.36	3.019	-48.39	-15.492
-56.14	-4.989	-54.22	14.894	-52.27	36.722	-50.3	2.587	-48.34	-15.967
-56.1	-5.042	-54.17	16.143	-52.22	36.027	-50.26	2.116	-48.29	-16.408
-56.04	-5.063	-54.12	17.322	-52.18	35.297	-50.21	1.734	-48.24	-16.735
-55.99	-5.109	-54.07	18.726	-52.13	34.254	-50.16	1.317	-48.18	-16.91
-55.93	-5.11	-54.02	19.919	-52.08	33.248	-50.11	0.933	-48.14	-17.15
-55.91	-5.205	-53.99	21.127	-52.03	32.287	-50.07	0.479	-48.09	-17.3
-55.86	-5.303	-53.93	22.352	-51.98	31.312	-50.02	0.009	-48.04	-17.339
-55.81	-5.438	-53.88	23.654	-51.93	30.267	-49.97	-0.383	-48	-17.291
-55.76	-5.576	-53.82	25.03	-51.88	29.282	-49.92	-0.756	-47.94	-17.231
-55.7	-5.622	-53.79	26.413	-51.82	28.262	-49.87	-1.182	-47.89	-17.104
-55.66	-5.642	-53.74	27.789	-51.79	27.279	-49.83	-1.607	-47.85	-16.979
-55.62	-5.477	-53.68	29.098	-51.74	26.318	-49.78	-2.012	-47.79	-16.854
-55.56	-5.257	-53.65	30.468	-51.69	25.322	-49.72	-2.459	-47.74	-16.694
-55.51	-4.93	-53.58	31.865	-51.65	24.145	-49.67	-2.902	-47.69	-16.458
-55.48	-4.581	-53.54	33.204	-51.6	22.974	-49.63	-3.392	-47.64	-16.152
-55.43	-4.138	-53.49	34.539	-51.55	21.582	-49.57	-3.889	-47.59	-15.777
-55.38	-3.734	-53.43	35.637	-51.49	20.344	-49.52	-4.459	-47.54	-15.22
-55.33	-3.214	-53.39	36.499	-51.45	19.246	-49.47	-4.919	-47.49	-14.731
-55.27	-2.626	-53.34	37.341	-51.39	18.054	-49.42	-5.293	-47.44	-14.264
-55.23	-2.109	-53.3	37.991	-51.35	17	-49.37	-5.726	-47.39	-13.675
-55.17	-1.522	-53.25	38.515	-51.29	16	-49.32	-6.139	-47.34	-13.074
-55.13	-0.92	-53.2	39.058	-51.24	14.991	-49.27	-6.572	-47.29	-12.489
-55.09	-0.286	-53.14	39.376	-51.2	14.141	-49.22	-7.025	-47.24	-12.009
-55.04	0.458	-53.1	39.605	-51.15	13.272	-49.18	-7.492	-47.19	-11.519
-54.99	1.104	-53.06	39.83	-51.1	12.446	-49.13	-8.054	-47.14	-10.908
-54.94	1.811	-53.01	39.863	-51.05	11.613	-49.08	-8.536	-47.09	-10.363
-54.88	2.398	-52.96	40.064	-51	10.694	-49.03	-9.091	-47.04	-9.659
-54.84	2.86	-52.91	40.403	-50.95	9.778	-48.99	-9.628	-46.99	-8.88
-54.79	3.694	-52.86	40.164	-50.9	9.175	-48.93	-10.221	-46.94	-7.977
-54.75	4.663	-52.8	39.658	-50.86	8.505	-48.89	-10.832	-46.88	-7.057
-54.69	5.245	-52.76	39.533	-50.81	7.865	-48.83	-11.452	-46.84	-5.953
-54.64	5.765	-52.72	39.417	-50.75	7.174	-48.78	-12.067	-46.79	-4.732
-54.61	6.459	-52.67	39.207	-50.71	6.526	-48.74	-12.602	-46.74	-3.377

TABLE 14: Continue III. Jupiter Zonal Wind data

φ	u(m/s)								
-46.69	-2.087	-44.68	30.037	-42.65	29.567	-40.6	3.686	-38.53	0.71
-46.63	-0.621	-44.62	30.081	-42.6	28.808	-40.54	3.263	-38.48	1.822
-46.59	0.905	-44.58	30.099	-42.55	28.097	-40.49	2.695	-38.43	3.175
-46.54	2.227	-44.53	30.31	-42.49	27.334	-40.45	2.254	-38.37	4.501
-46.5	3.625	-44.48	30.521	-42.44	26.606	-40.4	1.82	-38.32	5.782
-46.45	5.021	-44.42	30.72	-42.4	25.861	-40.34	1.387	-38.26	7.04
-46.4	6.505	-44.38	30.901	-42.34	25.147	-40.28	0.902	-38.22	8.483
-46.35	8.162	-44.32	31.142	-42.29	24.382	-40.24	0.655	-38.17	9.532
-46.3	9.856	-44.28	31.435	-42.23	23.644	-40.18	0.069	-38.12	10.831
-46.23	11.493	-44.23	31.704	-42.18	22.916	-40.13	-0.485	-38.06	12.154
-46.19	13.042	-44.17	31.983	-42.13	22.162	-40.08	-1.055	-38	13.577
-46.15	14.589	-44.12	32.36	-42.09	21.452	-40.03	-1.664	-37.96	14.941
-46.09	16.165	-44.07	32.747	-42.04	20.68	-39.98	-2.264	-37.9	16.448
-46.04	17.622	-44.02	33.159	-41.99	19.99	-39.92	-2.885	-37.85	17.774
-45.99	19.012	-43.97	33.52	-41.93	19.255	-39.86	-3.419	-37.79	19.046
-45.94	20.262	-43.92	33.936	-41.88	18.547	-39.83	-3.851	-37.74	20.165
-45.88	21.438	-43.87	34.337	-41.82	17.877	-39.77	-4.381	-37.69	21.307
-45.84	22.627	-43.81	34.694	-41.77	17.203	-39.72	-5.065	-37.64	22.654
-45.78	23.545	-43.76	35.29	-41.73	16.488	-39.67	-5.636	-37.59	23.927
-45.74	24.654	-43.71	35.588	-41.68	15.794	-39.62	-6.167	-37.54	25.21
-45.69	25.599	-43.66	35.966	-41.62	15.11	-39.57	-6.651	-37.49	26.559
-45.64	26.521	-43.62	36.229	-41.56	14.446	-39.51	-7.086	-37.43	27.878
-45.59	27.421	-43.57	36.386	-41.51	13.752	-39.45	-7.35	-37.38	29.155
-45.54	28.156	-43.51	36.517	-41.46	13.028	-39.4	-7.586	-37.34	30.435
-45.48	28.731	-43.46	36.581	-41.41	12.376	-39.35	-7.732	-37.28	31.63
-45.44	29.18	-43.41	36.529	-41.36	11.753	-39.3	-7.749	-37.23	32.782
-45.37	29.429	-43.36	36.452	-41.32	11.098	-39.24	-7.61	-37.18	33.787
-45.34	29.631	-43.3	36.262	-41.26	10.421	-39.2	-7.356	-37.13	34.752
-45.29	29.614	-43.26	36.084	-41.22	9.853	-39.15	-7.156	-37.07	35.638
-45.24	29.649	-43.21	35.833	-41.16	9.231	-39.1	-6.831	-37.02	36.464
-45.19	29.737	-43.16	35.511	-41.11	8.671	-39.05	-6.43	-36.96	37.181
-45.13	29.73	-43.11	35.097	-41.06	8.141	-39	-6.16	-36.91	38.044
-45.08	29.78	-43.06	34.616	-41.02	7.66	-38.94	-5.9	-36.86	38.952
-45.03	29.818	-43.01	34.075	-40.96	7.131	-38.88	-5.469	-36.81	39.827
-44.97	29.802	-42.96	33.551	-40.9	6.631	-38.84	-4.864	-36.76	40.664
-44.92	29.763	-42.91	32.97	-40.85	6.319	-38.79	-4.389	-36.7	41.472
-44.88	29.857	-42.86	32.336	-40.81	5.799	-38.74	-3.499	-36.66	42.086
-44.83	29.992	-42.8	31.607	-40.75	5.225	-38.69	-2.668	-36.6	42.85
-44.79	30.029	-42.75	30.961	-40.7	4.728	-38.63	-1.65	-36.55	43.404
-44.73	30.022	-42.7	30.273	-40.65	4.151	-38.58	-0.43	-36.49	43.842

TABLE 15: Continue IV. Jupiter Zonal Wind data

φ	u(m/s)	φ	u(m/s)	φ	u(m/s)	φ	u(m/s)	φ	u(m/s)
-36.44	44.307	-34.34	14.129	-32.21	-15.359	-30.06	12.179	-27.9	33.049
-36.39	44.761	-34.28	13.122	-32.15	-15.115	-30.01	12.89	-27.85	33.149
-36.34	45.119	-34.22	12.158	-32.09	-14.878	-29.96	13.614	-27.79	33.263
-36.29	45.422	-34.18	11.213	-32.05	-14.61	-29.91	14.376	-27.73	33.39
-36.24	45.772	-34.12	10.212	-31.99	-14.206	-29.85	15.046	-27.69	33.482
-36.18	46.165	-34.06	9.142	-31.94	-13.842	-29.8	15.765	-27.63	33.605
-36.13	46.249	-34.01	8.141	-31.89	-13.403	-29.74	16.514	-27.58	33.775
-36.07	46.45	-33.96	6.952	-31.84	-12.844	-29.69	17.222	-27.52	33.934
-36.02	46.456	-33.91	5.855	-31.78	-12.163	-29.63	17.868	-27.47	34.128
-35.96	46.341	-33.85	4.774	-31.72	-11.519	-29.57	18.547	-27.41	34.39
-35.91	46.018	-33.8	3.796	-31.66	-10.922	-29.52	19.274	-27.36	34.703
-35.86	45.621	-33.75	2.865	-31.62	-10.204	-29.47	19.999	-27.31	35.035
-35.82	45.081	-33.69	1.948	-31.57	-9.444	-29.42	20.622	-27.25	35.437
-35.76	44.59	-33.65	0.988	-31.51	-8.702	-29.36	21.342	-27.2	35.82
-35.71	43.975	-33.59	0.128	-31.46	-8.045	-29.31	22.134	-27.14	36.295
-35.66	43.358	-33.54	-0.788	-31.4	-7.281	-29.25	22.915	-27.09	36.77
-35.61	42.722	-33.49	-1.768	-31.35	-6.398	-29.21	23.577	-27.03	37.303
-35.55	42.061	-33.44	-2.858	-31.29	-5.552	-29.15	24.265	-26.98	37.824
-35.49	41.269	-33.37	-3.888	-31.24	-4.712	-29.09	24.902	-26.93	38.338
-35.44	40.403	-33.32	-4.911	-31.19	-3.892	-29.04	25.555	-26.87	38.909
-35.39	39.362	-33.27	-5.847	-31.13	-3.058	-28.99	26.125	-26.81	39.444
-35.33	38.279	-33.22	-6.797	-31.08	-2.224	-28.92	26.739	-26.76	39.998
-35.28	37.133	-33.16	-7.695	-31.04	-1.485	-28.88	27.357	-26.71	40.519
-35.22	35.955	-33.12	-8.598	-30.98	-0.702	-28.82	27.899	-26.65	41.005
-35.17	34.75	-33.06	-9.568	-30.92	0.175	-28.77	28.345	-26.6	41.525
-35.12	33.416	-33.01	-10.363	-30.87	0.936	-28.71	28.83	-26.54	41.958
-35.07	32.042	-32.95	-11.081	-30.82	1.67	-28.66	29.317	-26.49	42.312
-35.02	30.664	-32.9	-11.8	-30.77	2.452	-28.61	29.802	-26.43	42.592
-34.98	29.282	-32.84	-12.449	-30.71	3.202	-28.56	30.274	-26.38	42.739
-34.92	27.916	-32.79	-13.04	-30.66	3.965	-28.51	30.668	-26.32	42.814
-34.86	26.486	-32.73	-13.499	-30.6	4.752	-28.44	30.983	-26.27	42.812
-34.81	25.13	-32.69	-13.927	-30.55	5.446	-28.38	31.282	-26.21	42.731
-34.76	23.778	-32.64	-14.322	-30.5	6.183	-28.33	31.549	-26.16	42.507
-34.7	22.502	-32.58	-14.705	-30.45	7.004	-28.27	31.745	-26.1	42.206
-34.64	21.294	-32.53	-15.128	-30.38	7.737	-28.22	32.02	-26.05	41.842
-34.59	20.019	-32.48	-15.382	-30.33	8.478	-28.17	32.268	-26	41.388
-34.54	18.768	-32.42	-15.515	-30.27	9.126	-28.13	32.541	-25.94	40.802
-34.48	17.54	-32.37	-15.598	-30.22	9.961	-28.07	32.657	-25.89	40.172
-34.44	16.363	-32.31	-15.562	-30.17	10.641	-28.01	32.849	-25.83	39.44
-34.4	15.247	-32.25	-15.507	-30.11	11.403	-27.95	32.974	-25.78	38.602

TABLE 16: Continue V. Jupiter Zonal Wind data

φ	u(m/s)	φ	u(m/s)	φ	u(m/s)	φ	u(m/s)	φ	u(m/s)
-25.73	37.66	-23.53	1.819	-21.32	-39.26	-19.1	-47.801	-16.87	-15.876
-25.67	36.666	-23.47	1.033	-21.27	-40.917	-19.04	-47.141	-16.81	-15.239
-25.61	35.575	-23.42	0.36	-21.21	-42.413	-18.99	-46.695	-16.75	-14.432
-25.55	34.421	-23.36	-0.359	-21.15	-43.87	-18.92	-46.103	-16.69	-13.69
-25.51	33.258	-23.3	-1.133	-21.09	-45.246	-18.86	-45.429	-16.63	-12.881
-25.45	32.078	-23.24	-1.895	-21.04	-46.448	-18.81	-44.788	-16.57	-12.034
-25.39	30.878	-23.19	-2.573	-20.98	-47.737	-18.76	-44.099	-16.51	-11.112
-25.33	29.682	-23.13	-3.288	-20.93	-48.863	-18.7	-43.459	-16.45	-10.171
-25.28	28.508	-23.08	-4.034	-20.88	-49.928	-18.65	-42.643	-16.41	-9.222
-25.23	27.346	-23.03	-4.788	-20.82	-50.882	-18.6	-41.844	-16.35	-8.284
-25.17	26.218	-22.98	-5.566	-20.77	-51.75	-18.54	-41.082	-16.3	-7.245
-25.11	25.107	-22.93	-6.333	-20.71	-52.652	-18.48	-40.197	-16.25	-6.176
-25.07	23.972	-22.87	-7.099	-20.67	-53.394	-18.43	-39.297	-16.2	-5.128
-25.02	22.873	-22.81	-7.891	-20.61	-53.962	-18.37	-38.435	-16.14	-4.08
-24.96	21.784	-22.77	-8.672	-20.55	-54.434	-18.32	-37.624	-16.08	-3.052
-24.91	20.736	-22.71	-9.434	-20.49	-54.835	-18.26	-36.814	-16.02	-1.824
-24.86	19.727	-22.65	-10.197	-20.44	-55.195	-18.2	-36.055	-15.97	-0.578
-24.8	18.759	-22.59	-10.992	-20.38	-55.505	-18.14	-35.19	-15.91	0.62
-24.74	17.904	-22.54	-11.848	-20.32	-55.74	-18.09	-34.343	-15.85	1.932
-24.69	17.083	-22.48	-12.847	-20.27	-55.844	-18.03	-33.449	-15.79	3.232
-24.63	16.347	-22.43	-13.897	-20.21	-55.875	-17.98	-32.42	-15.74	4.579
-24.58	15.572	-22.37	-14.893	-20.15	-55.833	-17.92	-31.354	-15.68	5.907
-24.52	14.801	-22.32	-15.951	-20.1	-55.578	-17.86	-30.383	-15.63	7.246
-24.46	14.084	-22.26	-16.991	-20.03	-55.354	-17.81	-29.429	-15.57	8.608
-24.41	13.258	-22.2	-18.017	-19.98	-55.118	-17.76	-28.366	-15.52	9.934
-24.35	12.449	-22.14	-19.084	-19.94	-54.693	-17.7	-27.488	-15.45	11.287
-24.3	11.642	-22.09	-20.106	-19.88	-54.306	-17.64	-26.571	-15.39	12.64
-24.24	10.867	-22.04	-21.296	-19.82	-53.807	-17.58	-25.727	-15.33	14.032
-24.19	10.156	-21.98	-22.615	-19.77	-53.206	-17.53	-24.857	-15.28	15.394
-24.13	9.437	-21.92	-23.902	-19.71	-52.566	-17.47	-23.977	-15.23	16.61
-24.08	8.679	-21.87	-25.078	-19.65	-52.036	-17.41	-23.05	-15.18	17.722
-24.02	7.96	-21.81	-26.333	-19.6	-51.459	-17.35	-22.346	-15.12	18.976
-23.97	7.305	-21.76	-27.475	-19.54	-51.116	-17.3	-21.571	-15.06	20.213
-23.92	6.666	-21.7	-28.901	-19.48	-50.519	-17.24	-20.791	-15	21.173
-23.86	5.995	-21.65	-30.282	-19.44	-50.091	-17.19	-19.952	-14.95	22.08
-23.8	5.289	-21.6	-31.584	-19.38	-49.774	-17.13	-19.216	-14.9	22.752
-23.74	4.646	-21.54	-33.051	-19.32	-49.426	-17.08	-18.55	-14.84	23.542
-23.69	3.942	-21.49	-34.614	-19.26	-48.935	-17.03	-17.89	-14.78	24.193
-23.64	3.233	-21.43	-36.139	-19.21	-48.581	-16.97	-17.157	-14.73	24.714
-23.59	2.504	-21.38	-37.759	-19.14	-48.1	-16.91	-16.599	-14.67	25.339

TABLE 17: Continue VI. Jupiter Zonal Wind data

φ	u(m/s)	φ	u(m/s)	φ	u(m/s)	φ	u(m/s)	φ	u(m/s)
-14.61	25.74	-12.36	41.376	-10.08	54.407	-7.81	137.172	-5.53	118.803
-14.55	26.264	-12.3	41.889	-10.03	55.383	-7.75	138.696	-5.48	117.426
-14.5	26.795	-12.23	42.302	-9.97	56.415	-7.69	140.071	-5.42	115.914
-14.44	27.332	-12.18	42.699	-9.91	57.422	-7.63	141.306	-5.36	114.418
-14.38	27.809	-12.12	43.074	-9.87	58.609	-7.58	142.151	-5.3	113.023
-14.33	28.189	-12.07	43.5	-9.81	59.614	-7.51	143.034	-5.24	111.664
-14.27	28.553	-12.02	43.946	-9.74	60.896	-7.46	143.545	-5.19	110.268
-14.22	29.04	-11.95	44.316	-9.68	62.081	-7.41	143.969	-5.13	108.869
-14.17	29.413	-11.9	44.637	-9.63	63.195	-7.36	144.454	-5.08	107.501
-14.11	29.781	-11.84	44.852	-9.57	64.444	-7.3	144.791	-5.01	106.265
-14.05	29.945	-11.78	45.227	-9.52	65.726	-7.24	145.173	-4.96	105.007
-13.99	30.22	-11.73	45.512	-9.46	67.359	-7.18	145.249	-4.91	103.811
-13.93	30.553	-11.67	45.765	-9.41	68.96	-7.12	145.37	-4.85	102.631
-13.88	30.889	-11.61	45.961	-9.34	70.779	-7.06	145.456	-4.79	101.392
-13.82	31.343	-11.56	46.201	-9.29	72.751	-7.01	145.386	-4.73	100.195
-13.78	31.621	-11.5	46.428	-9.23	74.728	-6.95	145.285	-4.67	99.164
-13.71	31.989	-11.44	46.575	-9.18	77.115	-6.9	144.771	-4.62	98.225
-13.65	32.401	-11.39	46.768	-9.11	79.373	-6.84	144.296	-4.56	97.387
-13.59	32.711	-11.32	46.99	-9.06	81.821	-6.79	143.682	-4.5	96.587
-13.54	33.064	-11.27	47.216	-9.01	84.332	-6.73	142.908	-4.44	95.984
-13.48	33.49	-11.22	47.328	-8.96	86.875	-6.66	142.203	-4.39	95.398
-13.42	33.84	-11.16	47.48	-8.9	89.572	-6.61	141.524	-4.33	94.589
-13.37	34.292	-11.1	47.662	-8.84	92.08	-6.55	140.655	-4.27	94.057
-13.31	34.704	-11.05	47.891	-8.78	94.546	-6.5	139.806	-4.22	93.465
-13.25	35.136	-11	48.029	-8.72	97.077	-6.45	138.942	-4.16	92.865
-13.19	35.561	-10.95	48.078	-8.66	99.729	-6.38	138.005	-4.09	92.319
-13.15	35.979	-10.89	48.097	-8.6	102.528	-6.32	136.959	-4.04	91.689
-13.09	36.42	-10.82	48.118	-8.55	105.242	-6.27	135.823	-3.97	91.204
-13.03	36.843	-10.76	48.084	-8.5	108.057	-6.22	134.705	-3.93	90.662
-12.96	37.254	-10.71	48.105	-8.44	110.917	-6.16	133.535	-3.87	90.143
-12.92	37.577	-10.65	48.344	-8.38	113.701	-6.1	132.306	-3.82	89.621
-12.86	37.963	-10.6	48.527	-8.32	116.641	-6.04	130.993	-3.77	89.084
-12.8	38.227	-10.55	48.889	-8.26	119.403	-5.98	129.627	-3.71	88.628
-12.74	38.559	-10.49	49.403	-8.2	122.033	-5.93	128.355	-3.64	88.089
-12.69	38.796	-10.44	49.995	-8.15	124.534	-5.88	127.081	-3.59	87.543
-12.63	39.215	-10.38	50.586	-8.09	126.991	-5.82	125.738	-3.54	87.044
-12.57	39.61	-10.32	51.289	-8.04	129.294	-5.77	124.41	-3.49	86.464
-12.52	40.028	-10.27	52.058	-7.98	131.403	-5.7	123.024	-3.43	85.943
-12.46	40.405	-10.21	52.798	-7.92	133.409	-5.64	121.692	-3.37	85.413
-12.4	40.853	-10.14	53.566	-7.86	135.334	-5.59	120.281	-3.31	84.852

TABLE 18: Continue VII. Jupiter Zonal Wind data

φ	u(m/s)	φ	u(m/s)	φ	u(m/s)	φ	u(m/s)	φ	u(m/s)
-3.25	84.309	-0.96	70.254	1.3	71.755	3.59	81.992	5.89	110.243
-3.19	83.705	-0.9	70.13	1.36	71.75	3.64	82.414	5.94	110.546
-3.14	83.156	-0.84	70.031	1.42	71.729	3.71	82.76	5.99	110.958
-3.08	82.6	-0.77	69.965	1.47	71.754	3.77	83.133	6.04	111.284
-3.03	82.046	-0.73	69.958	1.53	71.72	3.82	83.567	6.1	111.394
-2.96	81.492	-0.66	69.945	1.59	71.687	3.86	84.082	6.16	111.348
-2.91	80.98	-0.61	69.927	1.65	71.664	3.93	84.455	6.22	111.625
-2.85	80.459	-0.56	69.98	1.71	71.64	3.99	84.83	6.27	111.84
-2.8	79.944	-0.51	70.054	1.77	71.613	4.05	85.366	6.33	111.815
-2.74	79.436	-0.45	70.064	1.82	71.629	4.1	86.056	6.39	111.693
-2.68	78.966	-0.39	70.097	1.88	71.732	4.17	86.649	6.45	111.484
-2.63	78.52	-0.32	70.138	1.94	71.888	4.22	87.259	6.5	111.354
-2.57	78.071	-0.26	70.247	2	72.085	4.28	87.774	6.56	111.314
-2.51	77.661	-0.21	70.326	2.04	72.315	4.33	88.395	6.61	111.239
-2.45	77.293	-0.16	70.441	2.09	72.561	4.39	89.204	6.67	110.901
-2.4	76.862	-0.1	70.576	2.15	72.83	4.45	89.933	6.72	110.58
-2.34	76.512	-0.05	70.741	2.21	73.047	4.5	90.807	6.78	110.12
-2.28	76.105	0	70.865	2.28	73.286	4.55	91.769	6.85	109.445
-2.22	75.709	0.05	71.021	2.34	73.49	4.61	92.66	6.9	109.052
-2.16	75.353	0.1	71.208	2.39	73.719	4.67	93.73	6.96	108.417
-2.1	75.021	0.15	71.325	2.45	74.038	4.73	94.77	7.02	107.881
-2.04	74.75	0.22	71.49	2.5	74.359	4.8	95.704	7.08	107.248
-1.98	74.471	0.27	71.588	2.55	74.796	4.84	96.548	7.13	106.646
-1.93	74.212	0.33	71.684	2.61	75.255	4.91	97.402	7.18	106.033
-1.88	73.939	0.39	71.807	2.67	75.654	4.96	98.252	7.24	105.333
-1.82	73.703	0.45	71.869	2.73	75.839	5.01	99.206	7.3	104.762
-1.77	73.483	0.51	71.932	2.79	76.156	5.08	100.139	7.36	104.052
-1.72	73.208	0.56	71.949	2.85	76.536	5.13	100.988	7.41	103.506
-1.65	72.946	0.61	71.941	2.9	76.908	5.19	101.877	7.47	102.883
-1.59	72.661	0.67	71.903	2.95	77.28	5.25	102.767	7.52	102.374
-1.53	72.369	0.73	71.884	3.01	77.687	5.3	103.726	7.58	101.774
-1.48	72.107	0.79	71.884	3.07	78.175	5.36	104.507	7.64	101.236
-1.42	71.866	0.84	71.881	3.13	78.617	5.42	105.227	7.7	100.845
-1.36	71.625	0.9	71.846	3.19	78.981	5.48	106.123	7.76	100.369
-1.3	71.352	0.97	71.855	3.24	79.412	5.53	106.896	7.82	99.981
-1.25	71.1	1.03	71.893	3.3	79.766	5.58	107.655	7.88	99.678
-1.2	70.908	1.08	71.865	3.36	80.229	5.64	108.286	7.92	99.329
-1.13	70.703	1.14	71.861	3.42	80.665	5.7	108.858	7.98	99.114
-1.09	70.551	1.19	71.808	3.48	81.108	5.76	109.448	8.05	98.763
-1.02	70.391	1.24	71.779	3.53	81.565	5.82	109.853	8.1	98.576

TABLE 19: Continue VIII. Jupiter Zonal Wind data

φ	u(m/s)	φ	u(m/s)	φ	u(m/s)	φ	u(m/s)	φ	u(m/s)
8.15	98.397	10.43	70.408	12.7	40.668	14.94	3.326	17.2	-20.024
8.21	98.269	10.48	69.513	12.76	39.462	15	2.638	17.26	-20.119
8.27	98.09	10.53	68.583	12.81	38.321	15.06	1.87	17.31	-20.138
8.33	97.791	10.59	67.678	12.87	37.089	15.12	1.232	17.36	-20.186
8.38	97.589	10.65	66.756	12.93	35.876	15.18	0.568	17.42	-20.16
8.44	97.395	10.71	65.872	12.97	34.654	15.24	-0.105	17.47	-20.112
8.5	96.906	10.77	64.964	13.03	33.533	15.29	-0.778	17.54	-19.987
8.56	96.451	10.82	64.043	13.08	32.424	15.34	-1.405	17.58	-19.853
8.62	95.972	10.88	63.201	13.14	31.345	15.4	-1.977	17.64	-19.658
8.67	95.357	10.94	62.489	13.2	30.267	15.45	-2.594	17.7	-19.452
8.73	94.912	11	61.767	13.27	29.201	15.51	-3.215	17.76	-19.21
8.78	94.302	11.05	61.067	13.32	28.218	15.56	-3.848	17.82	-18.858
8.84	93.575	11.11	60.654	13.38	27.067	15.62	-4.467	17.87	-18.421
8.89	92.85	11.17	60.231	13.44	25.946	15.67	-5.102	17.92	-18.083
8.95	92.126	11.23	59.78	13.49	24.881	15.74	-5.73	17.98	-17.678
9.01	91.476	11.29	59.297	13.54	23.842	15.79	-6.366	18.04	-17.212
9.05	90.709	11.34	58.867	13.6	22.887	15.85	-6.988	18.1	-16.672
9.1	90.088	11.39	58.5	13.65	21.887	15.9	-7.569	18.15	-16.082
9.18	89.349	11.44	58.07	13.71	21.005	15.96	-8.12	18.21	-15.554
9.24	88.639	11.49	57.663	13.76	20.133	16.02	-8.732	18.26	-14.906
9.29	87.83	11.55	57.224	13.82	19.308	16.07	-9.349	18.32	-14.284
9.35	87.113	11.62	56.821	13.88	18.467	16.14	-10.052	18.38	-13.549
9.39	86.4	11.68	56.34	13.93	17.679	16.19	-10.718	18.43	-12.799
9.46	85.601	11.74	55.847	13.99	16.819	16.24	-11.383	18.48	-12.029
9.53	84.785	11.8	55.35	14.04	15.973	16.3	-12.062	18.54	-11.211
9.58	84.018	11.85	54.775	14.1	15.108	16.35	-12.745	18.6	-10.421
9.64	83.268	11.91	54.118	14.16	14.286	16.41	-13.416	18.66	-9.63
9.69	82.413	11.95	53.384	14.22	13.487	16.46	-14.095	18.71	-8.731
9.75	81.646	12.01	52.608	14.27	12.632	16.52	-14.715	18.76	-7.845
9.81	80.746	12.06	51.833	14.33	11.77	16.58	-15.338	18.81	-6.985
9.87	79.864	12.12	50.977	14.39	10.936	16.62	-15.894	18.87	-6.04
9.92	78.926	12.18	50.131	14.45	10.11	16.68	-16.527	18.92	-5.13
9.98	77.988	12.24	49.129	14.51	9.322	16.74	-17.015	18.98	-4.202
10.04	77.044	12.3	48.143	14.56	8.555	16.79	-17.547	19.05	-3.147
10.09	76.039	12.35	47.159	14.61	7.802	16.86	-17.971	19.11	-2.085
10.15	75.017	12.41	46.143	14.67	7.082	16.91	-18.458	19.16	-0.996
10.2	74.081	12.47	45.16	14.72	6.294	16.97	-18.892	19.22	0.237
10.26	73.125	12.52	44.063	14.77	5.547	17.02	-19.259	19.27	1.452
10.32	72.149	12.58	43.061	14.83	4.753	17.08	-19.537	19.32	2.696
10.38	71.257	12.64	41.914	14.88	4.015	17.14	-19.908	19.38	4.104

TABLE 20: Continue IX. Jupiter Zonal Wind data

φ	u(m/s)	φ	u(m/s)	φ	u(m/s)	φ	u(m/s)	φ	u(m/s)
19.44	5.309	21.66	91.642	23.86	157.03	26.05	57.19	28.22	19.84
19.49	6.68	21.71	94.184	23.91	156.086	26.11	55.173	28.28	19.503
19.54	8.06	21.76	96.759	23.97	154.816	26.16	53.177	28.33	19.091
19.6	9.444	21.81	99.365	24.02	153.22	26.21	51.265	28.39	18.824
19.65	10.887	21.86	102.009	24.07	151.296	26.27	49.439	28.44	18.422
19.7	12.485	21.92	104.559	24.14	149.154	26.32	47.659	28.5	17.94
19.76	14.195	21.98	106.961	24.19	146.838	26.38	45.907	28.56	17.338
19.82	15.909	22.04	109.408	24.24	144.192	26.43	44.238	28.61	16.899
19.87	17.677	22.09	111.69	24.3	141.424	26.49	42.611	28.66	16.226
19.93	19.545	22.14	113.856	24.36	138.44	26.53	41.076	28.72	15.423
19.99	21.274	22.2	116.092	24.4	135.371	26.59	39.637	28.77	14.603
20.04	22.957	22.26	118.366	24.45	132.277	26.65	38.23	28.83	13.574
20.1	24.716	22.31	120.598	24.51	129.059	26.7	36.813	28.89	12.592
20.16	26.647	22.37	122.772	24.57	125.742	26.76	35.53	28.94	11.601
20.22	28.706	22.42	124.867	24.63	122.429	26.81	34.201	28.99	10.368
20.27	30.817	22.48	126.979	24.69	119.088	26.86	32.922	29.04	9.211
20.32	33.031	22.53	129.005	24.74	115.641	26.92	31.717	29.1	7.941
20.38	35.382	22.58	130.814	24.79	112.107	26.97	30.706	29.14	6.584
20.44	37.733	22.64	132.608	24.84	108.584	27.02	29.752	29.2	5.21
20.49	40.072	22.7	134.274	24.89	105.129	27.08	28.926	29.25	3.748
20.54	42.311	22.75	136.043	24.96	101.834	27.13	28.065	29.3	2.284
20.6	44.434	22.81	137.936	25.02	98.68	27.19	27.196	29.36	0.744
20.66	46.657	22.87	139.722	25.06	95.601	27.25	26.376	29.42	-0.884
20.71	48.929	22.93	141.441	25.11	92.612	27.3	25.702	29.47	-2.55
20.77	51.301	22.98	142.979	25.17	89.791	27.36	25.071	29.53	-4.238
20.82	53.73	23.04	144.45	25.23	87.157	27.41	24.531	29.59	-5.829
20.88	56.198	23.1	145.882	25.29	84.731	27.47	24.022	29.64	-7.438
20.92	58.691	23.16	147.101	25.35	82.537	27.52	23.578	29.68	-9.048
20.98	61.148	23.21	148.348	25.4	80.498	27.58	23.185	29.74	-10.554
21.05	63.565	23.26	149.57	25.45	78.869	27.63	22.913	29.8	-12.053
21.1	65.949	23.31	150.928	25.5	77.236	27.69	22.587	29.85	-13.522
21.16	68.52	23.37	152.587	25.55	75.254	27.74	22.256	29.9	-14.944
21.2	71.126	23.43	153.793	25.61	73.65	27.8	21.942	29.96	-16.344
21.25	73.691	23.48	155.23	25.65	71.608	27.86	21.638	30.01	-17.739
21.31	76.16	23.53	156.406	25.72	69.617	27.91	21.414	30.06	-19.129
21.37	78.752	23.57	157.41	25.78	67.515	27.97	21.178	30.12	-20.507
21.42	81.343	23.64	157.994	25.83	65.379	28.01	20.902	30.17	-21.717
21.48	83.962	23.69	158.161	25.88	63.285	28.07	20.818	30.22	-23.08
21.53	86.51	23.75	158.003	25.95	61.262	28.12	20.577	30.28	-24.181
21.6	89.036	23.8	157.659	26	59.206	28.17	20.255	30.33	-25.308

TABLE 21: Continue X. Jupiter Zonal Wind data

φ	u(m/s)	φ	u(m/s)	φ	u(m/s)	φ	u(m/s)	φ	u(m/s)
30.4	-26.186	32.52	-13.115	34.65	31.927	36.75	25.069	38.84	-17.102
30.42	-27.05	32.58	-12.056	34.71	32.371	36.79	24.188	38.9	-17.821
30.49	-27.757	32.64	-10.941	34.76	32.738	36.85	23.172	38.95	-18.51
30.55	-28.56	32.69	-9.854	34.82	33.045	36.91	22.07	38.99	-19.097
30.59	-29.206	32.75	-8.809	34.87	33.355	36.96	20.928	39.04	-19.562
30.65	-29.778	32.8	-7.588	34.92	33.553	37.02	19.724	39.1	-19.943
30.7	-30.157	32.85	-6.41	34.97	33.719	37.07	18.491	39.16	-20.375
30.75	-30.688	32.9	-5.157	35.02	33.847	37.12	17.054	39.2	-20.496
30.82	-30.996	32.96	-3.959	35.08	33.858	37.17	15.759	39.25	-20.431
30.88	-31.138	33.01	-2.728	35.13	33.819	37.23	14.424	39.31	-20.455
30.93	-31.286	33.06	-1.55	35.18	33.833	37.28	13.086	39.36	-20.431
30.98	-31.236	33.12	-0.277	35.24	33.882	37.32	11.666	39.41	-20.407
31.03	-31.226	33.17	1.07	35.28	33.844	37.38	10.202	39.46	-20.282
31.09	-31.208	33.22	2.328	35.34	33.744	37.43	8.892	39.51	-20.193
31.13	-30.976	33.28	3.646	35.39	33.666	37.49	7.444	39.56	-19.982
31.18	-30.805	33.33	4.904	35.45	33.501	37.54	6.113	39.61	-19.802
31.24	-30.611	33.39	6.297	35.5	33.347	37.6	4.767	39.66	-19.478
31.3	-30.299	33.43	7.654	35.55	33.217	37.65	3.285	39.72	-19.051
31.36	-29.913	33.48	8.928	35.6	33.021	37.7	1.942	39.78	-18.645
31.41	-29.516	33.54	10.234	35.66	32.706	37.75	0.712	39.82	-18.204
31.46	-29.172	33.6	11.586	35.71	32.442	37.81	-0.47	39.87	-17.775
31.51	-28.731	33.65	12.913	35.76	32.18	37.85	-1.565	39.93	-17.303
31.56	-28.158	33.7	14.246	35.81	31.944	37.9	-2.679	39.97	-16.763
31.61	-27.465	33.76	15.488	35.87	31.751	37.96	-3.818	40.02	-16.171
31.67	-26.663	33.81	16.691	35.92	31.501	38.01	-4.739	40.08	-15.652
31.72	-26.009	33.87	17.964	35.98	31.305	38.06	-5.678	40.13	-15.128
31.79	-25.286	33.91	19.204	36.03	31.016	38.11	-6.403	40.18	-14.625
31.83	-24.459	33.96	20.423	36.08	30.716	38.16	-7.213	40.24	-13.947
31.89	-23.675	34.01	21.622	36.13	30.516	38.22	-7.998	40.29	-13.326
31.93	-23.044	34.07	22.868	36.18	30.294	38.27	-8.702	40.35	-12.783
31.99	-22.3	34.12	24.022	36.23	30.075	38.33	-9.357	40.39	-12.369
32.05	-21.526	34.18	25.123	36.29	29.978	38.38	-10.433	40.44	-11.752
32.1	-20.728	34.23	26.121	36.34	29.64	38.42	-11.229	40.5	-11.188
32.15	-19.876	34.28	27.116	36.4	29.275	38.48	-12.03	40.55	-10.618
32.21	-18.98	34.34	27.995	36.45	28.859	38.53	-12.871	40.58	-9.999
32.26	-18.007	34.39	28.831	36.5	28.429	38.59	-13.702	40.64	-9.408
32.31	-17.024	34.44	29.542	36.55	27.941	38.64	-14.457	40.69	-8.818
32.37	-16.143	34.5	30.367	36.6	27.308	38.69	-15.219	40.74	-8.206
32.42	-15.232	34.55	30.954	36.65	26.631	38.74	-16.003	40.81	-7.538
32.47	-14.123	34.6	31.431	36.7	25.861	38.79	-16.437	40.86	-7.189

TABLE 22: Continue XI. Jupiter Zonal Wind data

φ	u(m/s)								
40.91	-6.631	42.96	23.178	44.99	0.06	47	14.895	48.98	-2.424
40.96	-5.976	43	22.683	45.03	0.15	47.05	15.178	49.03	-2.69
41.02	-5.233	43.05	22.017	45.09	0.298	47.1	15.33	49.08	-2.916
41.07	-4.487	43.1	21.282	45.15	0.413	47.15	15.398	49.13	-3.086
41.12	-3.736	43.15	20.417	45.19	0.543	47.2	15.438	49.17	-3.241
41.17	-3.001	43.2	19.482	45.24	0.663	47.24	15.508	49.23	-3.49
41.22	-2.184	43.25	18.553	45.29	0.878	47.29	15.377	49.28	-3.635
41.27	-1.452	43.31	17.573	45.34	1.08	47.35	15.238	49.32	-3.839
41.32	-0.881	43.37	16.59	45.39	1.26	47.4	15.003	49.37	-3.957
41.36	-0.15	43.41	15.59	45.43	1.482	47.44	14.728	49.44	-4.114
41.42	0.625	43.46	14.609	45.48	1.678	47.48	14.436	49.49	-4.26
41.47	1.54	43.52	13.58	45.53	1.92	47.54	14.067	49.53	-4.402
41.52	2.489	43.56	12.522	45.58	2.223	47.59	13.562	49.57	-4.507
41.58	3.451	43.62	11.466	45.63	2.501	47.64	13.053	49.62	-4.63
41.63	4.459	43.67	10.42	45.69	2.763	47.69	12.498	49.67	-4.668
41.68	5.495	43.72	9.46	45.74	3.105	47.75	11.873	49.72	-4.713
41.73	6.504	43.78	8.528	45.79	3.444	47.79	11.265	49.77	-4.737
41.78	7.544	43.82	7.64	45.82	3.82	47.84	10.606	49.81	-4.763
41.83	8.55	43.86	6.839	45.88	4.239	47.89	9.931	49.87	-4.793
41.88	9.724	43.92	6.009	45.94	4.656	47.94	9.218	49.93	-4.794
41.93	10.94	43.96	5.219	45.99	5.114	47.99	8.47	49.97	-4.776
41.99	12.13	44.02	4.445	46.04	5.561	48.04	7.758	50.01	-4.705
42.04	13.314	44.06	3.728	46.09	6.027	48.09	7.054	50.07	-4.668
42.08	14.462	44.11	3.061	46.14	6.553	48.15	6.283	50.12	-4.551
42.14	15.631	44.17	2.426	46.19	7.086	48.19	5.595	50.16	-4.478
42.18	16.747	44.23	1.887	46.24	7.633	48.24	4.898	50.2	-4.428
42.23	17.726	44.28	1.519	46.29	8.196	48.29	4.241	50.26	-4.418
42.29	18.667	44.33	1.218	46.34	8.767	48.34	3.526	50.31	-4.352
42.35	19.527	44.38	0.978	46.38	9.314	48.39	2.952	50.37	-4.434
42.4	20.332	44.43	0.794	46.44	9.863	48.43	2.286	50.41	-4.436
42.44	21.093	44.47	0.624	46.49	10.405	48.48	1.655	50.46	-4.471
42.5	21.707	44.52	0.589	46.54	10.913	48.54	1.089	50.51	-4.474
42.55	22.355	44.58	0.418	46.59	11.422	48.58	0.551	50.56	-4.534
42.6	22.838	44.62	0.309	46.64	11.89	48.63	0.037	50.6	-4.551
42.65	23.262	44.67	0.193	46.69	12.391	48.69	-0.411	50.65	-4.577
42.71	23.579	44.72	0.112	46.74	12.873	48.74	-0.859	50.71	-4.555
42.76	23.716	44.78	0.089	46.78	13.328	48.79	-1.26	50.76	-4.542
42.81	23.775	44.82	0.029	46.83	13.761	48.83	-1.558	50.8	-4.627
42.85	23.704	44.87	0.002	46.88	14.206	48.87	-1.854	50.85	-4.661
42.9	23.515	44.93	0.027	46.95	14.59	48.93	-2.18	50.9	-4.663

TABLE 23: Continue XII. Jupiter Zonal Wind data

φ	u(m/s)								
50.95	-4.698	52.9	-6.002	54.84	8.518	56.76	14.751	58.66	-6.168
51.01	-4.734	52.95	-5.999	54.89	9.176	56.81	13.954	58.71	-6.158
51.06	-4.812	53	-5.944	54.94	9.819	56.86	13.143	58.76	-6.178
51.1	-4.868	53.05	-5.919	54.99	10.499	56.91	12.21	58.81	-6.112
51.15	-4.957	53.1	-5.888	55.03	11.091	56.96	11.344	58.85	-6.091
51.19	-5.044	53.15	-5.854	55.08	11.721	57.01	10.418	58.89	-6.018
51.24	-5.176	53.2	-5.811	55.13	12.351	57.05	9.549	58.95	-6.013
51.3	-5.285	53.25	-5.767	55.18	13.013	57.08	8.635	58.99	-5.989
51.34	-5.413	53.29	-5.685	55.23	13.609	57.14	7.77	59.05	-5.977
51.39	-5.55	53.34	-5.584	55.28	14.256	57.2	6.87	59.1	-5.935
51.44	-5.676	53.39	-5.444	55.33	14.903	57.24	6.007	59.15	-5.939
51.49	-5.707	53.45	-5.346	55.37	15.528	57.29	5.16	59.18	-5.968
51.55	-5.774	53.49	-5.214	55.42	16.179	57.34	4.298	59.23	-5.977
51.59	-5.859	53.54	-5.031	55.46	16.835	57.38	3.371	59.27	-5.947
51.64	-5.952	53.59	-4.802	55.51	17.353	57.43	2.598	59.32	-5.931
51.69	-6.009	53.63	-4.574	55.56	17.905	57.48	1.854	59.38	-5.965
51.73	-6.026	53.68	-4.386	55.61	18.474	57.53	1.092	59.43	-5.975
51.79	-6.085	53.73	-4.073	55.67	18.944	57.58	0.367	59.48	-6.008
51.84	-6.174	53.78	-3.672	55.71	19.312	57.61	-0.312	59.52	-6.043
51.88	-6.256	53.83	-3.265	55.75	19.718	57.66	-0.992	59.56	-6.085
51.93	-6.308	53.86	-2.864	55.8	20.122	57.7	-1.614	59.61	-6.194
51.99	-6.393	53.91	-2.338	55.85	20.378	57.75	-2.208	59.66	-6.266
52.03	-6.405	53.97	-1.891	55.9	20.633	57.81	-2.786	59.71	-6.442
52.08	-6.442	54.02	-1.409	55.95	20.816	57.87	-3.317	59.75	-6.555
52.12	-6.474	54.07	-0.903	56	20.897	57.91	-3.802	59.8	-6.734
52.17	-6.464	54.12	-0.398	56.05	20.92	57.96	-4.286	59.85	-6.796
52.22	-6.453	54.16	0.172	56.09	20.81	58	-4.694	59.9	-6.841
52.27	-6.417	54.21	0.682	56.14	20.671	58.04	-5.013	59.94	-7.011
52.32	-6.377	54.26	1.171	56.18	20.444	58.09	-5.309	59.99	-7.049
52.36	-6.36	54.31	1.672	56.23	20.199	58.13	-5.556	60.03	-7.101
52.42	-6.351	54.37	2.219	56.29	19.961	58.18	-5.79	60.07	-7.14
52.47	-6.385	54.42	2.792	56.33	19.627	58.23	-5.953	60.13	-7.202
52.52	-6.391	54.46	3.381	56.38	19.272	58.29	-6.065	60.17	-7.273
52.57	-6.345	54.5	3.948	56.43	18.861	58.34	-6.108	60.22	-7.348
52.62	-6.307	54.55	4.569	56.48	18.387	58.38	-6.134	60.27	-7.405
52.67	-6.302	54.61	5.242	56.53	17.816	58.43	-6.045	60.32	-7.458
52.72	-6.249	54.65	5.877	56.57	17.329	58.48	-6.071	60.37	-7.495
52.76	-6.154	54.7	6.563	56.63	16.778	58.52	-6.092	60.41	-7.53
52.8	-6.07	54.74	7.193	56.67	16.163	58.58	-6.138	60.46	-7.543
52.86	-6.03	54.79	7.819	56.72	15.531	58.63	-6.19	60.51	-7.519

TABLE 24: Continue XIII. Jupiter Zonal Wind

φ	u(m/s)								
60.55	-7.513	62.42	-2.898	64.28	13.821	66.13	6.082	67.96	24.344
60.59	-7.487	62.46	-2.658	64.33	14.022	66.18	5.899	68.01	24.644
60.64	-7.434	62.51	-2.404	64.36	14.084	66.23	5.921	68.05	24.93
60.69	-7.408	62.56	-2.1	64.42	14.132	66.27	5.868	68.1	25.405
60.74	-7.334	62.6	-1.801	64.46	14.122	66.32	5.81	68.15	25.476
60.79	-7.282	62.65	-1.526	64.51	14.126	66.36	6.019	68.19	25.336
60.83	-7.181	62.7	-1.141	64.56	13.974	66.41	6.046	68.23	25.376
60.87	-7.115	62.75	-0.754	64.6	13.814	66.45	6.077	68.28	25.539
60.93	-7.03	62.8	-0.359	64.65	13.613	66.5	6.259	68.33	25.669
60.97	-6.944	62.85	0.024	64.7	13.396	66.55	6.469	68.37	25.725
61.03	-6.836	62.89	0.494	64.74	13.044	66.59	6.769	68.42	25.787
61.07	-6.667	62.94	0.827	64.79	12.8	66.64	7.033	68.46	25.801
61.12	-6.59	62.98	1.26	64.84	12.47	66.69	7.481	68.51	25.69
61.17	-6.48	63.03	1.699	64.89	12.103	66.73	7.896	68.55	25.762
61.22	-6.383	63.08	2.139	64.93	11.681	66.77	8.332	68.6	25.694
61.26	-6.288	63.12	2.626	64.97	11.243	66.81	8.949	68.64	25.526
61.31	-6.176	63.16	3.013	65.02	10.933	66.86	9.613	68.69	25.265
61.36	-6.112	63.22	3.515	65.07	10.544	66.91	10.148	68.74	25.237
61.4	-6.035	63.26	4.051	65.12	10.109	66.95	10.927	68.78	25.018
61.44	-5.903	63.31	4.478	65.16	9.766	67	11.596	68.83	24.824
61.49	-5.767	63.35	4.98	65.21	9.376	67.04	12.341	68.88	24.505
61.54	-5.712	63.41	5.513	65.26	8.981	67.09	12.98	68.92	24.305
61.6	-5.562	63.45	6.027	65.3	8.615	67.13	13.662	68.97	24.06
61.63	-5.443	63.49	6.493	65.34	8.294	67.18	14.439	69.01	23.845
61.67	-5.27	63.54	7.006	65.38	7.961	67.23	15.24	69.05	23.568
61.72	-5.122	63.59	7.529	65.43	7.667	67.27	15.992	69.1	23.304
61.77	-5.064	63.63	8.081	65.48	7.411	67.32	16.819	69.14	22.87
61.82	-4.859	63.68	8.534	65.53	7.198	67.37	17.505	69.19	22.534
61.85	-4.752	63.73	9.038	65.58	7.022	67.41	18.179	69.23	22.139
61.91	-4.666	63.78	9.542	65.62	6.858	67.46	18.969	69.28	21.487
61.95	-4.537	63.82	9.944	65.67	6.761	67.51	19.648	69.33	20.86
62	-4.523	63.87	10.469	65.72	6.774	67.55	20.179	69.38	20.242
62.05	-4.415	63.92	11.055	65.76	6.685	67.59	20.742	69.43	19.556
62.1	-4.235	63.97	11.541	65.8	6.567	67.64	21.204	69.46	18.912
62.15	-4.025	64.01	11.96	65.85	6.463	67.68	21.766	69.5	18.05
62.19	-3.798	64.06	12.261	65.89	6.325	67.73	22.32	69.55	17.253
62.23	-3.621	64.1	12.512	65.94	6.273	67.78	22.965	69.6	16.376
62.28	-3.417	64.14	12.791	65.99	6.3	67.83	23.527	69.64	15.339
62.34	-3.21	64.19	13.069	66.03	6.345	67.87	23.811	69.69	14.521
62.37	-3.057	64.24	13.458	66.09	6.247	67.92	24.143	69.74	13.612

TABLE 25: Continue XIV. Jupiter Zonal Wind data

φ	u(m/s)	φ	u(m/s)	φ	u(m/s)
69.79	12.832	71.58	-3.494	73.93	-1.919
69.83	12.136	71.63	-3.796	74.02	-1.969
69.87	11.22	71.69	-3.908	74.16	-2.211
69.92	10.022	71.73	-4.044	74.2	-2.244
69.97	9.307	71.78	-4.04	74.25	-2.446
70.02	8.541	71.81	-4.073	74.32	-2.384
70.05	7.859	71.87	-4.09	74.36	-2.245
70.09	7.179	71.93	-4.054	74.4	-1.995
70.15	6.525	72	-4.112	74.44	-1.476
70.19	5.994	72.02	-3.859	74.51	-1.747
70.23	5.449	72.13	-3.885	74.56	-1.648
70.28	4.911	72.26	-3.735	74.59	-1.67
70.32	4.523	72.35	-3.942	74.65	-1.235
70.37	4.015	72.44	-3.908	74.73	-1.04
70.42	3.647	72.53	-3.615	74.83	-0.854
70.46	3.397	72.56	-3.657	74.89	-0.351
70.51	3.128	72.6	-3.371	74.96	-0.152
70.55	2.98	72.68	-3.217	75.03	0.051
70.6	2.707	72.71	-2.636	75.09	0.258
70.64	2.573	72.77	-2.568	75.16	0.47
70.69	2.485	72.81	-2.261	75.23	0.686
70.73	2.278	72.83	-2.181	75.36	0.907
70.78	2.206	72.9	-2.192	75.45	1.132
70.82	1.956	72.94	-2.129	75.5	1.362
70.86	1.672	72.98	-1.956	75.63	1.596
70.91	1.356	73.01	-1.929	75.67	1.834
70.94	1.01	73.08	-1.881	75.9	2.077
70.99	0.658	73.11	-1.934	76.25	2.324
71.06	0.084	73.17	-1.805	76.98	2.576
71.09	-0.412	73.22	-1.756	77.59	2.832
71.13	-0.791	73.24	-1.773	77.65	3.092
71.19	-1.242	73.28	-1.755	77.69	3.357
71.21	-1.814	73.35	-1.523	78.25	3.627
71.25	-2.126	73.38	-1.421	78.54	3.901
71.32	-2.484	73.42	-1.406	78.72	4.179
71.35	-2.638	73.47	-1.62		
71.41	-2.857	73.5	-1.658		
71.45	-3.155	73.62	-1.62		
71.48	-3.343	73.71	-1.537		
71.55	-3.503	73.84	-1.598		



Saturn Zonal Winds.

TABLE 26: Saturn Zonal Wind data

φ	u(m/s)								
-89.9	7.1	-82.9	29.3	-75.6	31.1	-68	-14.6	-60.1	130.9
-89.8	7.5	-82.7	27.1	-75.4	35.4	-67.7	-14	-59.9	128
-89.7	13.2	-82.5	24.4	-75.2	41.6	-67.7	-13.3	-59.7	122.4
-89.6	47.5	-82.3	24.4	-75	50.8	-67.5	-12.6	-59.6	117.8
-89.4	76.3	-82.2	21.6	-74.9	58	-67.2	-11.7	-59.4	113
-89.2	102.1	-82.1	17.8	-74.8	62.3	-67.1	-11.2	-59.1	109.4
-89	137.4	-81.8	16.1	-74.6	71.1	-67	-10	-58.9	106
-88.9	132.5	-81.7	14.1	-74.3	83.8	-66.7	-8.9	-58.9	102.2
-88.8	152.8	-81.5	12.2	-74.2	90.4	-66.6	-8	-58.7	96.6
-88.6	153.1	-81.4	11.1	-74	91.1	-66.3	-7	-58.5	93
-88.4	146.4	-81.2	8.7	-73.9	92.6	-66.2	-5.9	-58.2	88.6
-88.1	144.1	-81	6.5	-73.7	90.6	-66.1	-4.6	-58	83.1
-88	146.6	-80.8	5.6	-73.5	86.3	-65.9	-3.3	-57.9	79
-88	145.7	-80.7	4.3	-73.3	79.3	-65.7	-2.2	-57.6	74.4
-87.8	139.4	-80.6	3.6	-73.2	71.6	-65.5	-0.7	-57.5	69.4
-87.6	142.6	-80.4	3.4	-73	68.1	-65.4	1.5	-57.3	65.6
-87.3	135	-80.2	2.7	-72.9	56	-65.2	3.3	-57.2	59.8
-87.2	134.4	-80	1.6	-72.7	43.7	-65	4.7	-57	57.4
-87.2	139.7	-79.9	1	-72.5	37.4	-64.7	6.1	-56.7	54.3
-87	135.5	-79.7	0.6	-72.3	33.5	-64.7	7.9	-56.6	51.1
-86.7	130.2	-79.6	-0.1	-72.1	28.8	-64.5	9.8	-56.4	46.2
-86.6	124.3	-79.4	-0.4	-72.1	24.5	-64.2	11.5	-56.1	41.4
-86.4	121.9	-79.2	-0.5	-71.8	18.2	-64.1	12.7	-56.1	38.6
-86.3	120.1	-79	-0.3	-71.6	13.5	-63.9	13.7	-55.9	35.2
-86.2	120.6	-78.9	-0.1	-71.5	11.1	-63.7	15.1	-55.7	32.3
-85.9	114.4	-78.8	0.1	-71.3	7.7	-63.6	16.4	-55.5	30.4
-85.8	109.2	-78.5	0.3	-71.2	4.3	-63.4	17.6	-55.2	28.2
-85.6	110.9	-78.4	0.1	-70.9	0.5	-63.2	19.9	-55.1	26.2
-85.5	104.1	-78.2	0.7	-70.7	-2.8	-63	24.6	-54.9	23.9
-85.4	92.9	-78.1	0.6	-70.6	-3.9	-62.9	36.2	-54.7	22.4
-85.1	87.2	-78	1.1	-70.5	-4.9	-62.6	51	-54.5	21.4
-84.9	79.7	-77.7	2.3	-70.3	-6.5	-62.5	60.5	-54.3	20.9
-84.8	74.6	-77.5	3.3	-70.1	-8.5	-62.2	68.1	-54.2	20.7
-84.7	68.9	-77.4	4.3	-69.9	-9.6	-62.1	83.3	-53.9	20.7
-84.5	62.3	-77.3	5.3	-69.7	-10.9	-62	103.7	-53.7	21.9
-84.3	56.2	-77.1	6.8	-69.7	-12.1	-61.8	112.2	-53.6	23
-84.1	50.7	-76.9	8.7	-69.5	-12.8	-61.6	126.8	-53.4	24
-84	48.5	-76.7	11	-69.2	-13.2	-61.4	133.1	-53.2	25.7
-83.9	44.7	-76.5	12	-69	-14	-61.2	140.4	-53	27.6
-83.7	46.2	-76.4	13.4	-68.9	-14.1	-61	144.7	-52.7	30.6
-83.5	40	-76.3	15.9	-68.7	-14.4	-60.9	146.5	-52.6	34.2
-83.2	35.2	-76	19.9	-68.6	-15	-60.7	143.5	-52.4	36.8
-83.1	33.3	-75.9	24.6	-68.4	-14.8	-60.5	139	-52.1	40
-83.1	32.3	-75.7	28.2	-68.2	-14.9	-60.2	133.8	-51.9	42.8

TABLE 27: Continue I. Saturn Zonal Wind data

φ	u(m/s)								
-51.7	47.2	-43	-1	-33.6	73.9	-23.6	143	-13.1	336.6
-51.5	52.5	-42.7	-2.2	-33.4	78	-23.4	147.9	-12.9	339.6
-51.5	57.2	-42.6	-3.6	-33.2	82.6	-23.2	152.8	-12.7	342.9
-51.3	60.9	-42.4	-4.3	-32.9	85.7	-22.9	157.2	-12.4	346.5
-51.1	65.8	-42.2	-4.9	-32.7	86.2	-22.7	160.8	-12.2	349.7
-50.9	71.4	-41.9	-5.8	-32.5	86.4	-22.5	164.3	-12	353.3
-50.7	78.4	-41.8	-6.4	-32.3	84.4	-22.2	168.7	-11.7	355.8
-50.5	85	-41.5	-7.5	-32	80.3	-22	171	-11.5	357.7
-50.2	91.2	-41.3	-8.3	-31.8	76.1	-21.7	176.4	-11.2	359.5
-50.1	97	-41.1	-9.1	-31.6	71.8	-21.4	181.5	-11	362.9
-49.8	104.3	-40.9	-10.4	-31.4	68	-21.3	185.3	-10.8	363.8
-49.7	112.8	-40.7	-11.4	-31.2	65.6	-21	191.3	-10.4	364.7
-49.5	118.6	-40.5	-11.9	-30.9	63.8	-20.7	196.6	-10.3	366.7
-49.3	122.7	-40.3	-12.1	-30.7	61	-20.6	201.8	-9.9	366.8
-49	125.1	-40.1	-12.8	-30.5	59.3	-20.3	206.3	-9.8	367.8
-48.9	127	-39.8	-13.1	-30.3	59.1	-20.1	209.4	-9.5	368.4
-48.7	123.3	-39.7	-12.3	-30	59.8	-19.8	213.4	-9.2	369
-48.4	130.1	-39.4	-11.5	-29.8	61.2	-19.5	217.7	-9	368.6
-48.2	144	-39.2	-9.6	-29.6	64.1	-19.4	221.4	-8.8	368.8
-48.1	139.5	-39	-7	-29.4	66.9	-19.2	224.4	-8.6	368.7
-47.9	130.3	-38.8	-4	-29.1	68.4	-18.9	228.5	-8.3	368.7
-47.7	119.2	-38.6	-1.1	-28.9	70.8	-18.6	231.8	-8.1	367.5
-47.4	113.7	-38.4	1.8	-28.7	74.4	-18.4	234.7	-7.8	366.5
-47.3	108.5	-38.2	5.2	-28.4	78.4	-18.2	237.2	-7.6	364.4
-47	100.8	-37.9	9.9	-28.2	81.6	-18	239.2	-7.3	365.2
-46.9	90	-37.7	14	-27.9	86.4	-17.7	242.1	-7.1	364.7
-46.6	76.7	-37.5	18.2	-27.8	90.2	-17.5	244.1	-6.8	361.3
-46.5	69	-37.3	23.2	-27.5	95.7	-17.3	246.6	-6.5	361.3
-46.2	63.5	-37.1	28.7	-27.2	100.5	-17	251.7	-6.3	357.6
-46.1	56.5	-36.9	33.4	-27	102.9	-16.8	256.9	-6.1	357.3
-45.8	48.8	-36.7	38.4	-26.9	106.1	-16.5	260.3	-5.9	353.4
-45.7	43.2	-36.4	43.3	-26.6	108.2	-16.3	263.5	-5.6	353
-45.5	37.6	-36.2	46.3	-26.3	110.7	-16	267.6	-5.4	352.3
-45.3	32.8	-36	48.2	-26.2	113.3	-15.8	273.3	-5.1	349
-45.1	26.6	-35.8	50.3	-25.8	116.8	-15.6	283.3	-4.8	349.6
-44.9	22.7	-35.6	51.8	-25.7	119.8	-15.3	288.7	-4.6	348.2
-44.7	20.4	-35.3	53	-25.5	122.9	-15.1	293.2	-4.3	347.8
-44.5	16.2	-35.1	54.6	-25.2	125.2	-14.9	299	-4.1	344.9
-44.2	13.1	-34.9	56.3	-25	126.8	-14.5	305.6	-3.9	345.2
-44	10.7	-34.7	58.4	-24.8	129	-14.4	311	-3.7	344.4
-43.8	7.8	-34.5	60.3	-24.5	131.1	-14.1	316.8	-3.3	354.2
-43.6	5	-34.3	63.1	-24.3	133.2	-13.9	321.8	-3.1	353.8
-43.4	1.6	-34	66.4	-24.1	135.5	-13.6	328.6	-2.9	355.9
-43.1	0.8	-33.7	70.2	-23.9	138.7	-13.4	332.9	-2.7	352.3

TABLE 28: Continue II. Saturn Zonal Wind data

φ	u(m/s)								
-2.4	348.5	8.4	385.5	19	218.5	29.5	78.6	39.1	-20
-2.1	356.5	8.6	385.2	19.3	214.9	29.7	76.4	39.3	-21.2
-1.9	351.1	8.9	386	19.5	210.8	29.9	74.8	39.5	-21
-1.7	347.5	9.2	386.7	19.7	208.6	30.1	73.6	39.8	-21.6
-1.4	353.6	9.3	385.3	20	204.9	30.4	73.2	40	-20.7
-1.1	364.4	9.6	382.6	20.2	202	30.6	72.6	40.2	-19.9
-0.9	370.4	9.8	381.5	20.5	198.7	30.8	72.7	40.4	-18.9
-0.7	370.1	10	379.1	20.7	196.1	31	73.7	40.6	-17.9
-0.4	371.5	10.4	380.4	20.9	192.8	31.3	74.9	40.8	-16.5
-0.2	384.2	10.6	373.3	21.2	190	31.5	75.9	41	-15.5
0.1	389.4	10.9	367.5	21.4	186.5	31.7	75.7	41.2	-13.6
0.3	384.5	11.1	363.2	21.6	184.1	31.9	75.1	41.4	-11.7
0.6	383.3	11.3	362	21.9	181.8	32.1	73.7	41.7	-9.8
0.8	367	11.6	360.5	22.1	179.5	32.4	70.9	41.9	-7.8
1	377.8	11.8	354.2	22.3	176.3	32.6	66.9	42.1	-5.9
1.3	368.1	12	349.7	22.6	173.3	32.8	64	42.3	-3.9
1.6	354	12.3	357	22.8	169.4	33	61.8	42.5	-1.7
1.8	352.9	12.5	353	23	164.9	33.3	60.2	42.7	1
2.1	347.9	12.7	342.1	23.3	161.4	33.5	59.4	42.9	4.5
2.3	353.1	13	336.5	23.5	159	33.7	59	43.1	7.9
2.6	354	13.3	331.7	23.7	155.3	33.9	59	43.3	11.7
2.7	351.5	13.5	326.8	24	152.5	34.1	58.8	43.5	15.7
3	342.3	13.7	321.5	24.2	148.5	34.4	58.9	43.7	21.3
3.2	346.8	13.9	319.7	24.4	144.1	34.6	58.4	43.9	27
3.5	349.4	14.3	312.7	24.7	138.8	34.8	57.8	44.1	32.8
3.7	354.8	14.4	310.4	24.9	135.2	35	56.8	44.3	42
4	357.3	14.7	306.3	25.1	131.1	35.2	55	44.5	46.1
4.3	354.7	15	308.2	25.4	127.8	35.5	52.7	44.8	51.6
4.5	354.7	15.1	306.8	25.6	123.9	35.7	48.9	45	59.4
4.7	359.3	15.5	301.8	25.8	119.6	35.9	44.4	45.2	65.6
4.9	361.6	15.6	297.6	26	116.1	36.1	39.2	45.3	72.5
5.3	360.3	15.9	293.5	26.3	113.9	36.3	32.7	45.6	79.5
5.4	364.4	16.2	286.2	26.5	110.7	36.5	27.6	45.8	86.1
5.7	363.6	16.4	276.8	26.7	106.4	36.8	23	46	92
6	367.7	16.7	267.9	27	103.7	37	17.7	46.2	105.5
6.2	366.3	16.9	260.3	27.2	100.7	37.2	12.9	46.4	114.5
6.5	369.2	17.1	253.9	27.4	98	37.4	8.3	46.6	119.4
6.7	372.7	17.4	249.2	27.7	95.5	37.6	2.7	46.8	124.9
6.9	375.4	17.6	243.1	27.9	93.3	37.8	-3.1	46.9	128.2
7.2	376.3	17.9	239	28.1	91.5	38.1	-7.5	47.2	133.5
7.4	377.8	18.1	235.2	28.3	89.8	38.3	-11.3	47.3	135.5
7.7	380.6	18.3	231.2	28.6	88.2	38.5	-14.4	47.5	138
7.9	381.9	18.6	225.6	28.8	85.4	38.7	-17.1	47.8	144.6
8.2	385.2	18.8	222.1	29	83	38.9	-18.3	47.9	139.8

TABLE 29: Continue III. Saturn Zonal Wind data

φ	u(m/s)								
48.2	141.5	56.7	3.5	64.7	65.4	72.4	-10.5	80.1	12.6
48.3	138.1	56.9	5	64.9	67.8	72.6	-11.3	80.2	12.9
48.6	132.3	57.1	6.3	65.1	70	72.7	-11.9	80.6	14.5
48.8	126	57.3	8.1	65.3	73.5	72.9	-12	80.8	12.2
49	117.3	57.5	10.6	65.5	77.2	73	-12.2	80.9	10.5
49.2	110.8	57.6	14.8	65.6	81.8	73.2	-11.6	81	9.6
49.3	102.9	57.8	16.9	65.8	87.1	73.4	-11.3	81.2	12.7
49.5	95	58	19.6	66	92.2	73.6	-11.1	81.4	7.4
49.7	81.1	58.1	22.4	66.2	95	73.8	-10.9	81.5	10.8
50	61.7	58.3	27.1	66.3	96.4	73.9	-10.6	81.7	9.4
50.2	58.4	58.6	29.9	66.5	96.3	74.1	-10.7	81.8	10.4
50.3	52.3	58.8	33.6	66.7	94.6	74.3	-9.3	82.1	13.7
50.6	50.7	58.9	38.8	66.9	89.1	74.4	-7.7	82.2	12.4
50.8	47.4	59.1	42.3	67	81.2	74.6	-6.3	82.3	20.3
51	45.5	59.3	46.1	67.1	74.7	74.8	-7	82.5	18.8
51.2	44.7	59.5	50.7	67.4	66.2	74.9	-4.7	82.7	22.8
51.3	43.9	59.7	55.6	67.6	59.2	75.1	-4	82.9	23.3
51.5	30	59.9	60.9	67.7	50.6	75.2	-1.2	83	25.4
51.7	25.9	60.1	65.9	67.9	44.6	75.4	-0.9	83.4	33
51.8	24.8	60.2	69.8	68.1	39.7	75.5	2.5	83.6	33
52.1	23.3	60.4	73.8	68.2	34.4	75.8	3.2	83.7	32.9
52.3	21.5	60.6	78.4	68.4	30.6	75.9	9.5	83.9	43.2
52.5	18.8	60.8	83.2	68.6	27.4	76.1	12.3	84	39.7
52.7	16	61	86.9	68.8	24.3	76.3	14.9	84.2	41.6
52.9	13.4	61.1	89.7	69	21.6	76.4	19.3		
53.1	10.9	61.3	91.7	69.1	19.1	76.5	24.3		
53.3	9.2	61.5	92.5	69.3	16.1	76.7	26.3		
53.5	8.1	61.7	92.1	69.5	13	77	34.3		
53.7	5.8	61.9	90.4	69.7	9.3	77	50.1		
53.9	4.2	62.1	87.3	69.8	7.1	77.2	61.8		
54.1	2.7	62.2	84.2	69.9	4.4	77.4	70.3		
54.3	1.7	62.4	81.5	70.2	1.7	77.6	79		
54.4	0.1	62.6	78.5	70.3	-0.6	77.7	74.2		
54.6	-0.9	62.8	75.1	70.5	-1.7	77.9	91.7		
54.8	-1.7	63	71.8	70.7	-2.4	78	94		
55	-2.3	63.1	69.3	70.9	-3.6	78.4	91		
55.2	-2.8	63.3	67.2	70.9	-5.2	78.6	68		
55.4	-2.8	63.5	65.5	71.2	-6.2	78.9	48.3		
55.6	-2.7	63.7	64.1	71.4	-7.2	79	47.8		
55.8	-2.4	63.9	63	71.5	-7.7	79.2	41.2		
56	-1.9	64	62.5	71.7	-8.3	79.4	32.3		
56.1	-1	64.2	62.5	71.8	-9.2	79.6	30.9		
56.3	0.3	64.4	63.2	72.1	-10	79.7	26.2		
56.5	1.2	64.6	64.4	72.1	-10.4	79.9	19.9		