Simulating Mid-Latitude Cyclones with the Shallow Water Model

Nishanth Shubhakar Mandala
New York University
nishanth.mandala@nyu.edu

April 27, 2025

Contents

1. Introduction

Cyclones—intense, rotating low-pressure systems—are among the most significant and dynamic phenomena in atmospheric science. They span a wide spectrum, from tropical cyclones (hurricanes and typhoons) to mid-latitude (extra-tropical) cyclones [17]. Understanding their formation, structure, and evolution requires not only physical intuition but also a strong mathematical framework.

In this project, we employ the shallow water equations as a simplified yet powerful model to simulate the formation of mid-latitude cyclones. Although these equations make the assumption of a single-layer fluid, they capture essential features of atmospheric motion such as horizontal flow, wave dynamics, and mass conservation [20]. Their relative simplicity makes them ideal for exploring large-scale geophysical processes with a focus on conceptual clarity.

A critical element in cyclone dynamics is Earth's rotation, modeled here through the Coriolis force. This apparent force causes moving air to deflect, giving rise to the characteristic spiral structure of cyclones [9]. Another key factor is differential solar heating: the equator receives more sunlight than the poles, establishing temperature and pressure gradients that drive atmospheric circulation [14].

By combining a spherical shallow water model with rotational effects and a simple representation of solar forcing, we simulate the spontaneous emergence and evolution of a cyclone. This framework allows us to investigate which features of real-world storms are recoverable within a one-layer system and provides insight into the strengths and limitations of reduced-complexity atmospheric models.

2. Theoretical Background

Before diving into the meat of the simulation, it's important to understand the theories and models that make the simulation work.

The first building block is realizing that the simulation happens on a sphere — specifically, a simplified version of Earth. The spherical geometry of Earth is described using coordinates called latitude and longitude. Latitudes are imaginary lines that run east-west but measure how far north or south a point is from the equator. Longitudes run north-south and measure how far east or west a point is from the Prime Meridian[1] [5]. For mathematical convenience, we refer to latitude and longitude using the symbols φ and λ (see Appendix 9.1). These are converted into radians and can easily be transformed into Cartesian coordinates to help

[1]The Prime Meridian is defined as 0° longitude and runs through Greenwich, England.

with plotting and computing derivatives.

Next, we define how the air (or fluid, in this model) moves. The simulation uses two wind components: u, which represents wind moving east-west along lines of constant latitude (zonal wind), and v, which represents wind moving north-south along lines of constant longitude (meridional wind). These velocities are updated at each grid point on the spherical grid and are responsible for how the fluid evolves over time (see Appendix 9.2).

To make the simulation a bit more realistic, we also introduce topography — in this case, using "Gaussian mountains." These are artificial but mathematically smooth bumps in the surface height that mimic terrain and influence wind patterns as air moves around them (see Appendix 9.3).

Another key ingredient is the Coriolis force. Because Earth rotates, moving fluids appear to deflect from their path — to the right in the Northern Hemisphere and to the left in the Southern Hemisphere [2]. This effect is described in the model using the Coriolis parameter f, which depends on latitude (see Appendix 9.4). Without it, we wouldn't get the classic spiral shape of cyclones.

The simulation also includes solar forcing — the input of energy from the Sun. In reality, sunlight heats the Earth unevenly, creating temperature gradients that lead to pressure differences, which in turn create wind. In this model, solar forcing is simplified as a moving "hotspot" that injects energy into the fluid over time [10]. This drives cyclone development in the study mode of the simulation.

To prevent the simulation from spiraling into chaos, we include friction. Friction is a resistive force that slows things down, similar to surface drag or internal fluid viscosity. Without it, energy would accumulate endlessly. In the code, friction appears as a damping term on the wind velocities (see Appendix 9.7).

Finally, we have the shallow water equations themselves. These are a set of fluid dynamics equations used when the fluid layer is thin compared to its horizontal extent. They describe horizontal fluid motion and allow the height of the fluid to vary at each point, making them suitable for modeling large-scale atmospheric motion. To keep things from drifting too far from reality, a relaxation term is included that gently returns the fluid height to a baseline value over time (see Appendix 9.5).

All of these elements come together to create a working, interactive model of cyclone dynamics on a rotating planet — simplified, but still full of rich behavior[2].

[2]Please note that the explanations and derivations provided in the Appendix are pulled from a collection of academic literature and do not represent novelty or rigor, but a representation optimized and simplified for the sake of simulation.

3. Governing Equations

3.1. Domain Assumptions

Building upon these foundational concepts—particularly the role of Coriolis effects, topography, and solar forcing—we can now formalize the mathematical framework. In the next section, we present the governing equations that translate these physical ideas into a solvable system describing fluid motion on a rotating sphere.

Before we dive into the equations that govern the state of the model, we must discuss our domain assumptions.

- A thin rotating fluid layer on the surface of a sphere of radius R.
- Coordinates:
  - φ: latitude (radians) (See Appendix 9.1)
  - λ: longitude (radians) (See Appendix 9.1)
- Horizontal velocity components as functions of latitude, longitude, and time (t):
  - $u(\varphi, \lambda, t)$: zonal (east-west) wind speed (See Appendix 9.2)
  - $v(\varphi, \lambda, t)$: meridional (north-south) wind speed (See Appendix 9.2)
- fluid height (analogous to pressure) $h(\varphi, \lambda, t)$ as a function of latitude, longitude, and time.

3.2. Atmospheric Equations

3.2.1 Momentum Equations

In a rotating, shallow fluid layer (like the atmosphere or ocean), the momentum equations describe how the wind (horizontal velocity) evolves in time (See Appendix 9.5):

$$\frac{du}{dt} = -\left(u\frac{1}{R\cos\varphi}\frac{\partial u}{\partial \lambda} + v\frac{1}{R}\frac{\partial u}{\partial \varphi}\right) - \frac{1}{R\cos\varphi}\frac{\partial \Phi}{\partial \lambda} + fv - \frac{uv\tan\varphi}{R} - \gamma u \tag{1}$$

$$\frac{dv}{dt} = -\left(u\frac{1}{R\cos\varphi}\frac{\partial v}{\partial \lambda} + v\frac{1}{R}\frac{\partial v}{\partial \varphi}\right) - \frac{1}{R}\frac{\partial \Phi}{\partial \varphi} - fu + \frac{u^2\tan\varphi}{R} - \gamma v \tag{2}$$

Where,

- u is the zonal wind (east-west, along constant latitude) (See Appendix 9.2)
- v is the meridional wind (north-south, along constant longitude) (See Appendix 9.2)

- R is the Earth's radius
- φ is the latitude (radians) (See Appendix 9.1)
- λ is the longitude (radians) (See Appendix 9.1)
- $h(\varphi, \lambda, t)$ is the fluid height (like atmospheric pressure) (See Appendix 9.5)
- g is the acceleration due to gravity
- b is the topography of the Earth (See Appendix 9.3)
- $\Phi = g(h + b)$ is the geopotential (gravity + topography) (See Appendix9.6)
- $f = 2\Omega \sin \varphi$ is the Coriolis parameter (See Appendix 9.4)
- γ is the friction coefficient (See Appendix 9.7)

3.2.2 The Continuity Equation

The continuity equation expresses conservation of mass in a fluid. In shallow water systems, the "mass" of the fluid is proportional to the height h. So, changes in h reflect whether fluid is piling up or spreading out.

$$\frac{\partial h}{\partial t} = -\frac{1}{R \cos \varphi} \frac{\partial}{\partial \lambda}(hu \cos \varphi) - \frac{1}{R} \frac{\partial}{\partial \varphi}(hv) + Q(\varphi, \lambda, t) - \varepsilon(h - h_0) \tag{3}$$

Where,

- $h(\varphi, \lambda, t)$ is the fluid height (like atmospheric pressure) (See Appendix 9.5)
- u is the zonal wind (east-west, along constant latitude) (See Appendix 9.2)
- v is the meridional wind (north-south, along constant longitude) (See Appendix 9.2)
- R: radius of the Earth
- φ is the latitude (radians) (See Appendix 9.1)
- λ is the longitude (radians) (See Appendix 9.1)
- Q is the external forcing (solar heating) (See Appendix 9.8)
- ε is the relaxation rate (See Appendix 9.5)
- h_0: background (rest) fluid height (See Appendix 9.5)

3.2.3 Solar Forcing

$$Q(\varphi, \lambda, t) = Q_{\text{amp}} \cdot \exp\left(-\left[\frac{(\varphi - \varphi_s(t))^2}{\delta_\varphi^2} + \frac{(\lambda - \lambda_s(t))^2}{\delta_\lambda^2}\right]\right) \tag{4}$$

Where:

- Q_{amp} is the amplitude of solar heating (See Appendix 9.8).
- φ is the latitude (radians) (See Appendix 9.1)
- λ is the longitude (radians) (See Appendix 9.1)
- $\varphi_s(t)$ is the solar latitude (center of the heating zone at time t) (See Appendix 9.8).
- $\lambda_s(t)$ is the solar longitude (center of the heating zone at time t) (See Appendix 9.8).
- $\delta_\varphi, \delta_\lambda$ are the latitudinal and longitudinal width parameters (spread of the heating area) (See Appendix 9.8).

3.3. Initial and Topographic Conditions

The simulation is initialized with a balanced cyclonic vortex and a topographic feature represented by a Gaussian mountain. The cyclone is geostrophically and centrifugally balanced, while the mountain introduces fixed surface perturbation into the height field.

Cyclone Height Field The initial fluid height includes a depression centered at (φ_c, λ_c):

$$h(r) = h_0 - \Delta h \cdot \exp\left(-\left(\frac{r}{\sigma}\right)^2\right) \tag{5}$$

Pressure Gradient The radial derivative of the cyclone's height field is:

$$\frac{dh}{dr} = \left(\frac{2\Delta h}{\sigma^2}\right) r \cdot \exp\left(-\left(\frac{r}{\sigma}\right)^2\right) \tag{6}$$

Gradient Wind Balance The tangential wind speed $V(r)$ is obtained from the gradient wind balance:

$$V(r) = \frac{-f_c r + \sqrt{(f_c r)^2 + 4gr\frac{dh}{dr}}}{2} \tag{7}$$

Cyclone Wind Components The tangential velocity is decomposed into zonal and meridional components:

$$u = -V \cdot \frac{d\varphi}{r}, \quad v = V \cdot \frac{d\lambda}{r} \tag{8}$$

Topographic Height Field A stationary Gaussian mountain is added to the surface topography:

$$b(\varphi,\lambda) = A \cdot \exp\left(-\frac{(\varphi-\varphi_m)^2 + (\lambda-\lambda_m)^2}{\delta^2}\right) \tag{9}$$

Where,

- h_0: background fluid height (See Appendix 9.5)
- Δh: maximum cyclone height depression (See Appendix 9.9)
- σ: radial scale of the cyclone (See Appendix 9.9)
- r: great-circle distance from cyclone center (See Appendix 9.9)
- $f_c = 2\Omega \sin(\varphi_c)$: Coriolis parameter at cyclone center (See Appendix 9.4)
- g: gravitational acceleration
- u is the zonal wind (east-west, along constant latitude) (See Appendix 9.2)
- v is the meridional wind (north-south, along constant longitude) (See Appendix 9.2)
- A: mountain amplitude (height) (See Appendix 9.3)
- (φ_m, λ_m): location of mountain center (See Appendix 9.3)
- δ: width parameter of the mountain (See Appendix 9.3)
- $b(\varphi,\lambda)$: topographic elevation (See Appendix 9.3)

3.4. Diagnostic Equations (Post-Processing)

These formulas are not part of the core equations of motion, but are calculated at every simulation step to help track the system's behavior over time. They're useful for understanding how well the model conserves key physical quantities and how the cyclone evolves (See Appendix 9.10).

Total Mass

$$M = \iint h \cos\varphi \, d\varphi \, d\lambda$$

This measures the total amount of fluid in the system. In theory, if there's no external input or loss, the mass should stay constant throughout the simulation.

Total Energy

$$E = \iint \left[\frac{1}{2} h(u^2 + v^2) + gh \left(b + \frac{1}{2} h \right) \right] \cos\varphi \, d\varphi \, d\lambda$$

This is the sum of kinetic and potential energy across the globe.

Potential Enstrophy

$$Z = \iint \left(\frac{\zeta + f}{h} \right)^2 h \cos\varphi \, d\varphi \, d\lambda$$

Potential enstrophy is a measure of how twisted or rotational the fluid is, taking into account both local vorticity and the planetary Coriolis effect. It's especially important in large-scale atmospheric dynamics and helps reveal whether the cyclone is getting more organized or chaotic.

Angular Momentum

$$L = \iint h(xv - yu) \cos\varphi \, d\varphi \, d\lambda \quad \text{with } x = \cos\varphi \cos\lambda, \quad y = \cos\varphi \sin\lambda$$

Angular momentum gives us a sense of the overall "spin" of the fluid around the Earth's axis.

Cumulative Solar Heating Energy

$$E_{\text{solar}}(t) = \int_0^t \left(\iint \Phi(\varphi, \lambda, t') Q(\varphi, \lambda, t') \cos\varphi \, d\varphi \, d\lambda \right) dt'$$

This tracks the total amount of energy added to the system through localized solar heating over time. The heating varies dynamically with the Sun's position.

Cumulative Relaxation Sink Energy

$$E_{\text{relax}}(t) = \int_0^t \left(\iint \Phi(\varphi, \lambda, t') \left(-\varepsilon (h(\varphi, \lambda, t') - h_0) \right) \cos\varphi \, d\varphi \, d\lambda \right) dt'$$

This measures the total energy loss caused by a relaxation process that tends to return the fluid height toward a background state h_0.

Cumulative Drag Dissipation Energy

$$E_{\text{drag}}(t) = \int_0^t \left(\iint \left(-\gamma h(\varphi, \lambda, t') (u(\varphi, \lambda, t')^2 + v(\varphi, \lambda, t')^2) \right) \cos\varphi \, d\varphi \, d\lambda \right) dt'$$

This accounts for the total energy removed from the system due to atmospheric drag, which slows down the winds through frictional effects.

Where,

- $h(\varphi, \lambda, t)$ is the fluid height (See Appendix 9.5)
- φ is the latitude (radians) (See Appendix 9.1)
- λ is the longitude (radians) (See Appendix 9.1)
- u, v are the zonal and meridional wind components (See Appendix 9.2)
- g is the gravitational acceleration
- b is the topographic height (See Appendix 9.3)
- $\Phi = g(h + b)$ is the geopotential (gravitational + topographic energy per unit mass)
- $Q(\varphi, \lambda, t)$ is the solar heating forcing term
- γ is the drag coefficient
- ζ is the relative vorticity, defined as:

$$\zeta = \frac{1}{R\cos\varphi}\frac{\partial v}{\partial \lambda} - \frac{1}{R\cos\varphi}\frac{\partial}{\partial \varphi}(u\cos\varphi)$$

- $f = 2\Omega\sin\varphi$ is the Coriolis parameter (See Appendix 9.4)
- R is the Earth's radius

4. Numerical Method

With the governing shallow water equations fully defined, attention turns to the methods by which we solve them in practice. In the following section, we detail the numerical scheme, discussing both the choice of time-stepping method and the implementation details that ensure stable and accurate solutions

The system evolves under gravity, planetary rotation (Coriolis force), pressure gradients, and solar forcing. The equations include damping (friction), height relaxation, and optional heating forcing. In *validation mode*, forcing is disabled; in *study mode*, all effects are included.

This simulation includes Coriolis effects, surface topography, radiative forcing, frictional damping, and a height relaxation term.

As a reminder, the initial height field includes a Gaussian depression centered at (φ_c, λ_c), with velocity fields derived from geostrophic balance:

$$h(\varphi, \lambda, 0) = h_0 - \delta h \cdot e^{-r^2/\sigma^2}, \quad \text{with } r^2 = (\varphi - \varphi_c)^2 + (\lambda - \lambda_c)^2.$$

4.1. Forcing and Smoothing

To simulate more realistic atmospheric dynamics — and to avoid numerical instabilities — the model incorporates several external and internal adjustments. These include artificial heating to drive motion, spatial filtering to remove grid-scale noise, and safety clamps on velocity magnitudes to prevent blow-ups.

- **Solar Heating:** In the real atmosphere, differential heating from the Sun drives large-scale circulation by creating spatially varying pressure gradients. In this model, we simulate that effect through a time-dependent heating term $Q(\varphi, \lambda, t)$ that mimics the motion of the Sun's energy across the globe.

  The heating is applied using a two-dimensional Gaussian function:

  $$Q(\varphi, \lambda, t) = Q_{\text{amp}} \cdot \exp\left(-\left[\frac{(\varphi - \varphi_s(t))^2}{\delta_\varphi^2} + \frac{(\lambda - \lambda_s(t))^2}{\delta_\lambda^2}\right]\right)$$

  This function centers the heating at a moving "solar subpoint" $(\varphi_s(t), \lambda_s(t))$ that oscillates north-south (seasonal variation) and moves longitudinally (daily rotation), while δ_φ and δ_λ determine the latitudinal and longitudinal width of the heating patch.

  This forcing term gets added to the continuity equation and modifies the height field directly. Physically, it increases fluid height (i.e., local pressure) in heated areas, which then produces a pressure gradient that drives fluid motion. It plays a vital role in triggering cyclonic motion from an otherwise steady state.

- **Polar Smoothing:** One of the challenges in working with spherical grids is the clustering of grid points near the poles (a result of latitude-longitude geometry). This leads to much smaller grid spacing at high latitudes, which can cause numerical instabilities due to high gradients and poorly resolved dynamics.

  To mitigate this, we apply Gaussian smoothing to the h, u, and v fields near the poles. Specifically, the model identifies points where $|\varphi| > 60°$ and smooths them with a Gaussian filter of kernel size (e.g., 5). Mathematically, this is similar to applying a low-pass filter that damps out short-wavelength fluctuations — the kind that are most likely to blow up due to finite differencing.

Without this filter, the simulation would often suffer from unphysical oscillations or "ringing" near the poles, especially as the cyclone approaches higher latitudes. Polar smoothing thus acts as a stabilizing layer that allows the simulation to run longer and remain physically meaningful.

- **Velocity Clamping:** Another critical safeguard is a maximum wind speed limit. Because the simulation uses a forward-stepping integration scheme (Symplectic Euler), a single time step with large velocity values can create massive errors or lead to total divergence.

  To avoid this, the model computes the speed at every grid point:

  $$\text{speed} = \sqrt{u^2 + v^2}$$

  and if the speed exceeds a certain threshold (e.g., 0.2 in nondimensional units), it scales both u and v down proportionally:

  $$u \to u \cdot \frac{v_{\max}}{\text{speed}}, \quad v \to v \cdot \frac{v_{\max}}{\text{speed}}$$

  This essentially caps the wind speed at a physically reasonable value and prevents instability without introducing artificial diffusion or damping. It's a numerical "emergency brake" that ensures the model doesn't crash even during moments of rapid cyclone intensification.

  While this approach slightly breaks conservation (e.g., of energy or momentum), it enables longer simulations and better visual interpretability — a worthwhile tradeoff for educational and exploratory models like this one.

4.2. Symplectic Euler

To evolve the state of the system forward in time, we need a numerical integration method. While the simplest option is Forward Euler — which updates all quantities using their rates of change at the current time step — that method can be unstable for long-running physical simulations. In particular, Forward Euler tends to drift in energy and fails to conserve important quantities like momentum and total mass [24], especially when dealing with oscillatory systems like waves and rotating fluids.

Instead, we use the **Symplectic Euler method**, a time-stepping scheme designed for systems governed by conservation laws. It's especially useful in Hamiltonian systems — where the dynamics follow from energy or momentum principles — and while the shallow water equations aren't strictly Hamiltonian, they behave similarly enough that symplectic integration makes a big difference in stability and realism [6].

We begin by discretizing time into steps of size Δt, so the n-th time step occurs at time $t^n = n \cdot \Delta t$. We denote the key fields at each step as:

$$u^n = u(t^n), \quad v^n = v(t^n), \quad h^n = h(t^n)$$

representing the zonal wind, meridional wind, and fluid height, respectively.

The update procedure for Symplectic Euler is:

1. **Compute tendencies at the current time:** First, we calculate the derivatives of each field using the current values:

$$\frac{\partial u}{\partial t}^n, \quad \frac{\partial v}{\partial t}^n, \quad \frac{\partial h}{\partial t}^n$$

   These tendencies come from the right-hand sides of the momentum and continuity equations, incorporating pressure gradients, Coriolis force, friction, and solar forcing.

2. **Update velocities using current tendencies:**

$$u^{n+1} = u^n + \Delta t \cdot \frac{\partial u}{\partial t}^n, \qquad v^{n+1} = v^n + \Delta t \cdot \frac{\partial v}{\partial t}^n$$

   Here, we only use information from the current step to compute the new velocities. This avoids having to solve a coupled system and keeps the method explicit.

3. **Update height using the new velocities:**

$$h^{n+1} = h^n + \Delta t \cdot \frac{\partial h}{\partial t}^{n+1}$$

   This is the "symplectic" part — instead of updating h using its own current derivative, we recompute the continuity equation with the *new* velocities (u^{n+1}, v^{n+1}). This staggered update ensures that energy and momentum are more accurately preserved, and reduces artificial oscillations and blow-ups.

This scheme is particularly important for our cyclone simulation because the system contains both fast (gravity wave) and slow (geostrophic) dynamics. Using Symplectic Euler ensures that:

- Oscillatory modes (like wave propagation) don't artificially grow or decay.

- Long-term quantities like total energy and angular momentum remain bounded.
- The simulation can run for thousands of steps without "blowing up."

It's also worth noting that while more advanced integrators (e.g., Runge-Kutta, semi-implicit schemes) exist, Symplectic Euler offers a good tradeoff between accuracy, conservation, and computational simplicity — making it great for our exploratory modeling purposes.

5. Validation

To ensure the accuracy and physical realism of the model, we track a series of diagnostic quantities throughout the simulation. These metrics are computed at each time step and provide insight into conservation properties, energy dynamics, and cyclone behavior. In particular, they help validate the model in both *study mode* (with solar forcing and damping) and *validation mode* (idealized, with no external forcing or dissipation).

5.1. Diagnostic Quantities

Total Mass Total mass is a fundamental conserved quantity in ideal shallow water systems. In the absence of external forcing or sinks/sources, the total fluid mass should remain constant over time. Any deviation may indicate numerical diffusion, instability, or implementation errors (See Appendix 9.10).

$$M = \iint h \cos\varphi \, d\varphi \, d\lambda$$

Where h is the fluid height, and $\cos\varphi$ accounts for the spherical surface area element.

Validation: In validation mode (no forcing or relaxation), total mass should remain nearly constant. Small fluctuations may occur due to numerical discretization. In study mode, slight long-term trends may emerge due to heating or damping effects.

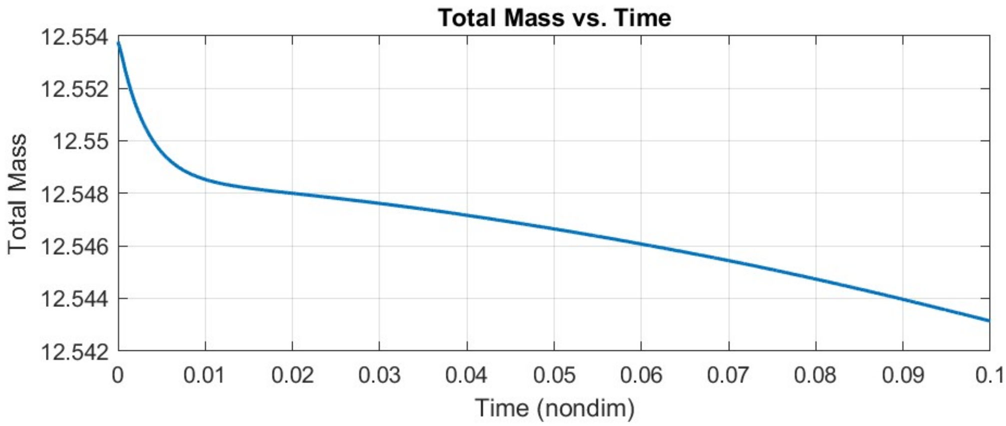

Figure 1: Time series of total mass in study mode

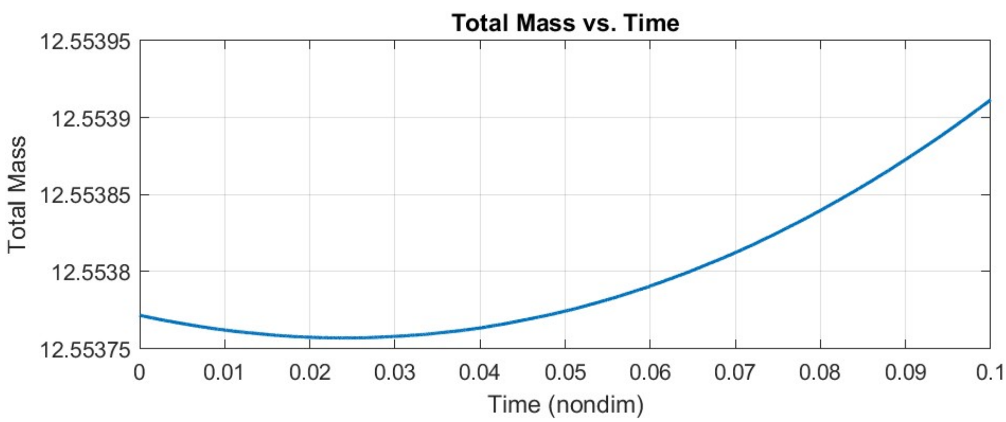

Figure 2: Time series of total mass in validation mode

Total Energy Total energy consists of kinetic and potential components. The kinetic term represents wind motion, while the potential term includes gravitational and topographic contributions. Monitoring energy reveals whether external forcing (like solar heating) is balanced by damping, or if numerical methods introduce artificial loss/gain (See Appendix 9.10).

$$E = \iint \left[\frac{1}{2} h(u^2 + v^2) + gh \left(b + \frac{1}{2} h \right) \right] \cos\varphi \, d\varphi \, d\lambda$$

Where u, v are wind velocities, g is gravity, and b is topographic height.

Validation: In validation mode, total energy should remain almost constant. In study mode, energy may fluctuate due to solar input and dissipative losses.

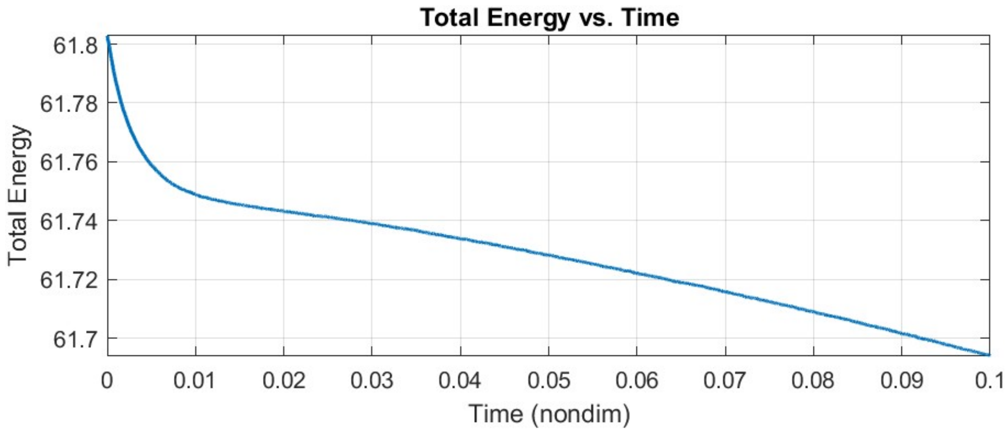

Figure 3: Time series of total energy in study mode

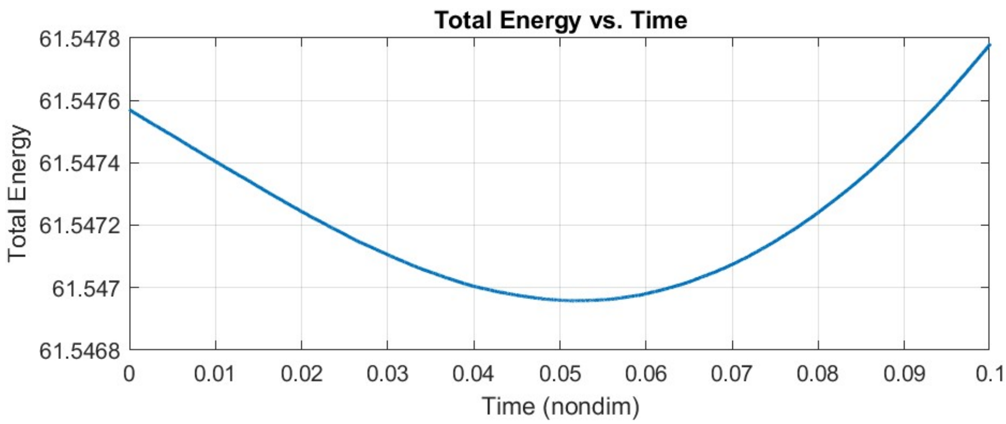

Figure 4: Time series of total energy in validation mode

Potential Enstrophy Potential enstrophy quantifies the rotational and vorticity-based energy content of the fluid. It helps detect changes in cyclone coherence or emergence of turbulent structures. It's especially sensitive to small-scale numerical noise (See Appendix 9.10).

$$Z = \iint \left(\frac{\zeta + f}{h}\right)^2 h \cos\varphi \, d\varphi \, d\lambda$$

$$\text{with } \zeta = \frac{1}{R\cos\varphi}\frac{\partial v}{\partial \lambda} - \frac{1}{R\cos\varphi}\frac{\partial}{\partial \varphi}(u\cos\varphi)$$

Validation: In the absence of dissipation, potential enstrophy should be approximately conserved. Small drifts are common due to vorticity calculation sensitivity.

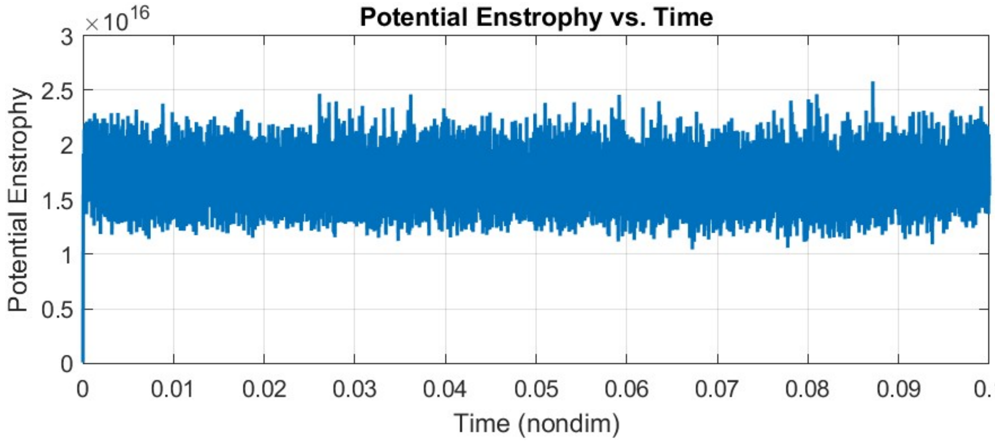

Figure 5: Time series of potential enstrophy in study mode

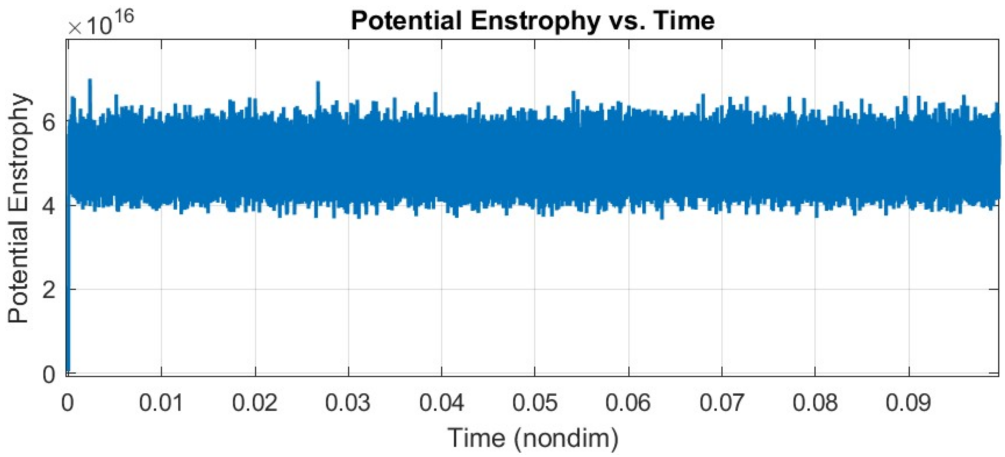

Figure 6: Time series of potential enstrophy in validation mode

Angular Momentum Angular momentum tracks the system's net rotation and helps verify that the model correctly incorporates the Earth's spherical geometry. It's useful in identifying artificial torques or unintended drift in wind circulation (See Appendix 9.10).

$$L = \iint h(xv - yu)\cos\varphi\, d\varphi\, d\lambda \quad \text{with } x = \cos\varphi\cos\lambda, \quad y = \cos\varphi\sin\lambda$$

Validation: Angular momentum should be nearly conserved in validation mode. In study mode, it may drift due to asymmetric solar heating or topographic effects.

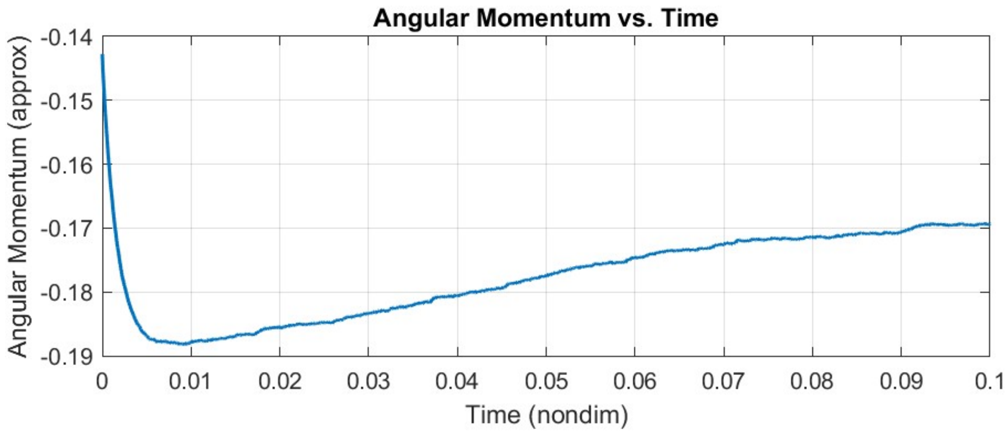

Figure 7: Time series of angular momentum in study mode

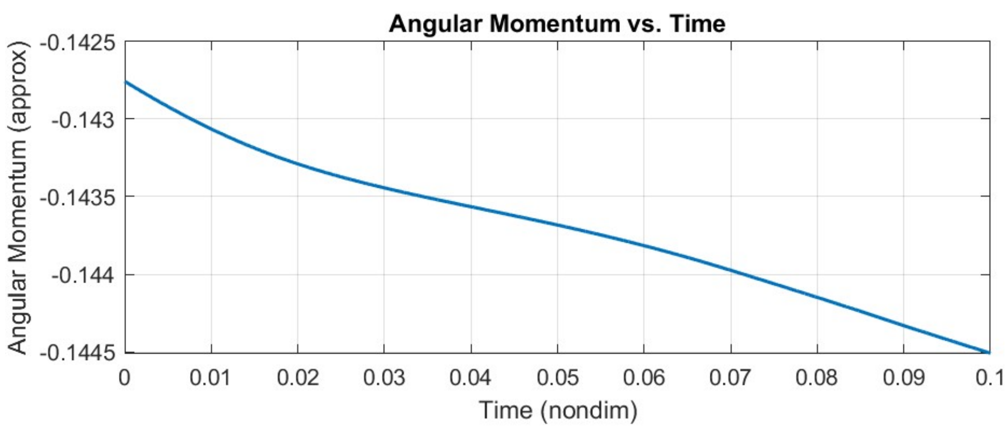

Figure 8: Time series of angular momentum in validation mode

Cumulative Solar Heating, Relaxation, and Drag Energies In study mode, external sources and sinks modify the total energy of the system over time. We track cumulative energy added by solar heating, and energy removed through relaxation and drag dissipation (See Appendix 9.10).

The cumulative energy contributions are computed as:

$$E_{\text{solar}}(t) = \int_0^t \left(\iint \Phi(\lambda, \varphi, t') Q(\lambda, \varphi, t') \cos\varphi \, d\lambda \, d\varphi \right) dt'$$

$$E_{\text{relax}}(t) = \int_0^t \left(\iint \Phi(\lambda, \varphi, t') \left(-\text{relax\_rate}(h(\lambda, \varphi, t') - h_0) \right) \cos\varphi \, d\lambda \, d\varphi \right) dt'$$

$$E_{\text{drag}}(t) = \int_0^t \left(\iint \left(-\gamma h(\lambda, \varphi, t') (u(\lambda, \varphi, t')^2 + v(\lambda, \varphi, t')^2) \right) \cos\varphi \, d\lambda \, d\varphi \right) dt'$$

Where $\Phi = g(h + b)$ is the geopotential, Q is the solar heating forcing, and γ is the drag coefficient.

Validation:

- In study mode, cumulative solar heating should steadily rise (positive energy input), while cumulative relaxation and drag should steadily become more negative (energy removal).

- In validation mode, with no forcing or dissipation active, all cumulative energy contributions should remain near zero.

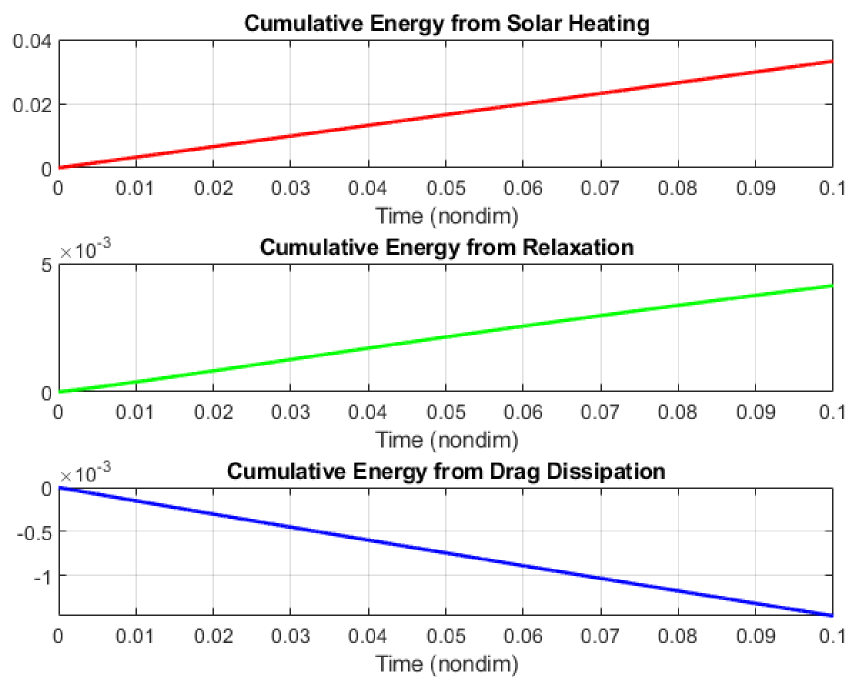

Figure 9: Cumulative energy contributions from solar heating, relaxation sink, and drag dissipation over time in study mode. Solar heating injects energy (positive), while relaxation and drag remove energy (negative).

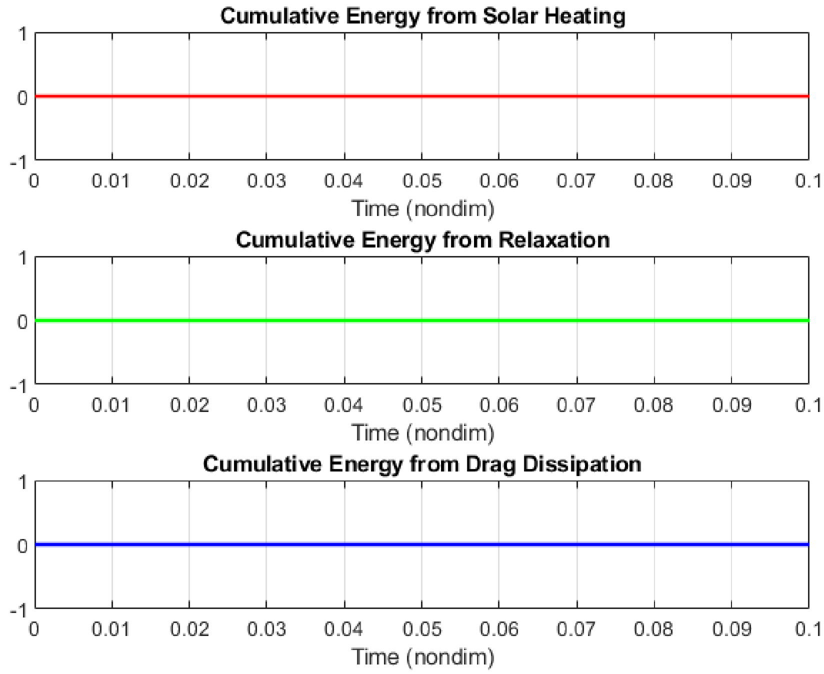

Figure 10: Cumulative energy contributions from solar heating, relaxation sink, and drag dissipation over time in validation mode. All contributions remain near zero, indicating absence of external forcing.

5.1.1 Diagnostics Summary

At each simulation step, we compute the following metrics:

- **Total mass** — to verify conservation under ideal conditions.
- **Total energy** — to monitor energetic exchanges due to forcing or dissipation.
- **Potential enstrophy** — to measure vortex strength and track cyclone coherence.
- **Angular momentum** — to assess spin conservation and the effect of topographic or heating asymmetries.
- **Cumulative energy contributions (solar heating, relaxation sink, drag dissipation)** — to evaluate the external energy budget and energy removal processes.

These diagnostics validate the physical and numerical consistency of the model and highlight the distinct behaviors that emerge under study versus validation modes. Together, they form a robust set of tools for evaluating long-term cyclone evolution and model stability.

6. Implementation and Code Considerations

This section presents a detailed implementation and explanation of our spherical shallow water simulation that models cyclone dynamics. The model incorporates friction, solar heating, height relaxation, and polar/midlatitude smoothing. The simulation includes diagnostics for physical conservation and storm intensity.

6.1. User Settings and Mode Configuration

The simulation allows toggling between two modes:

- **Study Mode:** Enables solar heating, friction, and height relaxation.
- **Validation Mode:** Disables external forcing and dissipation for benchmarking conservation properties.

Listing 1: User Mode Configuration

```
study_mode = true;
params.validation_mode = ~study_mode;
params.gamma = 1;
params.relax_rate = 1;
params.Q_amp = 1;
params.h0 = 1;
```

Explanation: These parameters define whether damping and solar heating are active, as well as set the fluid background depth.

6.2. Grid and Physical Constants Setup

We initialize the Earth's parameters, grid resolution, and time stepping. The simulation uses a latitude-longitude grid.

Listing 2: Grid and Constants

```
R = 1;  % nondimensional Earth radius
g = 9.81;  % gravitational acceleration
Omega = 7.2921e-5;  % Earth's angular velocity
dLat = deg2rad(2); dLon = deg2rad(2);
dt = 0.0001; nSteps = 1000; skipFrame = 1;
[latVec, lonVec] = deal(deg2rad(-90:2:90), deg2rad(0:2:358));
[LAM, PHI] = meshgrid(lonVec, latVec);
```

Explanation: The spherical grid covers the globe with 2-degree resolution. Time step and number of steps define simulation duration.

6.3. Topography and Cyclone Initialization

A Gaussian mountain is embedded optionally, followed by initializing a balanced cyclone using the shallow water equilibrium condition.

Listing 3: Topography and Cyclone Setup

```
lat0 = deg2rad(30); lon0 = deg2rad(90);
b = 0.1 * exp(-((PHI - lat0).^2 + (LAM - lon0).^2)/0.1);
if params.validation_mode, b = 0*b; end

[u, v, h] = balanced_cyclone_initial_condition(PHI, LAM, h0, 0.05, 0.3, deg2rad(35),
    deg2rad(100), Omega, g);
```

Explanation: Balanced cyclone conditions ensure geostrophic balance initially. Topography is excluded in validation mode.

6.4. Derivative Operators and Coriolis Parameter

Numerical derivatives are defined using centered differences, and the Coriolis parameter is amplified slightly for display.

Listing 4: Derivatives and Coriolis Force

```
f = 2 * Omega * sin(PHI);
d_dlambda = @(A) (circshift(A,[0,-1]) - circshift(A,[0,1])) / (2*dLon);
d_dphi    = @(A) (circshift(A,[-1,0]) - circshift(A,[1,0])) / (2*dLat);
```

Explanation: We define operators to compute derivatives across latitude and longitude, used in advection and pressure gradients.

6.5. Diagnostic Preallocation and Display Setup

Quiver plots of velocity vectors on the globe are used to track flow visually. Arrays for mass, energy, and other diagnostics are initialized.

Listing 5: Diagnostics and Display Initialization

```
diag_mass = zeros(nSteps,1);
diag_energy = zeros(nSteps,1);
diag_enstrophy = zeros(nSteps,1);
diag_momentum = zeros(nSteps,1);
diag_max_speed = zeros(nSteps,1);
```

Explanation: These arrays will hold global quantities for each time step, allowing validation of physical principles.

To also monitor external energy inputs and dissipation, we preallocate arrays to store cumulative energy contributions from solar heating, relaxation, and drag:

Listing 6: Cumulative Energy Preallocation

```
power_solar = zeros(nSteps,1);
power_relax = zeros(nSteps,1);
power_drag  = zeros(nSteps,1);
energy_solar = zeros(nSteps,1);
energy_relax = zeros(nSteps,1);
energy_drag  = zeros(nSteps,1);
```

Explanation: At each timestep, we calculate the instantaneous power injected (or removed) by different physical processes, and cumulatively integrate them to track how the total energy budget evolves over time.

6.6. Main Loop I: Recording the Cumulative Energies

At each step, we compute the instantaneous power and update cumulative energy integrals:

Listing 7: Instantaneous Power and Cumulative Energy Update

```
Phi = g * (h + b);  % Geopotential

% Solar forcing calculation
if params.validation_mode
    Qf = 0;
else
    ss = 2*pi/5000;
    lonS = mod(ss*(it*dt),2*pi);
    latS = deg2rad(10)*cos(2*pi*(it*dt)/20000);
    Qf = params.Q_amp * exp(-(((PHI-latS).^2)/deg2rad(15)^2 + ((LAM-lonS).^2)/deg2rad(15)
    ^2));
end

% Power at this timestep
```

```
power_solar(it) = sum(sum((Phi .* Qf) .* cosPHI)) * area_elem;
power_relax(it) = sum(sum((Phi .* (-relax_rate .* (h - params.h0))) .* cosPHI)) *
    area_elem;
power_drag(it)  = sum(sum((-gamma .* h .* (u.^2 + v.^2)) .* cosPHI)) * area_elem;

% Cumulative energy
if it == 1
    energy_solar(it) = power_solar(it) * dt;
    energy_relax(it) = power_relax(it) * dt;
    energy_drag(it)  = power_drag(it) * dt;
else
    energy_solar(it) = energy_solar(it-1) + power_solar(it) * dt;
    energy_relax(it) = energy_relax(it-1) + power_relax(it) * dt;
    energy_drag(it)  = energy_drag(it-1) + power_drag(it)  * dt;
end
```

Explanation: We calculate solar energy input, relaxation sink, and drag dissipation separately at each timestep. Their cumulative integrals allow us to quantify the full energy balance over time.

6.7. Main Loop II: Momentum and Continuity Update

Each step computes momentum tendencies, forcing terms, and height changes using a symplectic Euler update scheme.

Listing 8: Main Update Loop Excerpt

```
dudt = -adv_u - (1./(R*cosPHI)) .* gradPhi_lambda + f .* v - geo_u - gamma .* u;
dvdt = -adv_v - (1./R) .* gradPhi_phi - f .* u - geo_v - gamma .* v;
u_new = u + dt .* dudt;
v_new = v + dt .* dvdt;
```

Explanation: Velocity is updated by summing advection, pressure gradients, Coriolis, and friction. Stability is maintained with clamping.

Listing 9: Height Update and Forcing

```
dhdt = -(1./(R*cosPHI)) .* d_dlambda(h.*u_new.*cosPHI) ...
     - (1./R) .* d_dphi(h.*v_new) + Q_forcing - relax_rate .* (h - h0);
h_new = h + dt .* dhdt;
```

Explanation: The height field changes according to the continuity equation, with source terms from heating and relaxation.

6.8. Arrow Display and Reset Logic

To visualize velocity flow, quiver arrows are moved with smoothed velocities and reset when drifting too far.

Listing 10: Quiver Arrow Updating

```
smooth_uq = smooth_factor * prev_uq + (1 - smooth_factor) * uq;
px = px + move_factor * smooth_uq; ... % similar for py, pz
```

Explanation: Arrows give a dynamic, smooth indication of wind vectors on a rotating sphere. Arrows are normalized and reset periodically.

6.9. Balanced Cyclone Initial Condition

```
function [u, v, h] = balanced_cyclone_initial_condition(PHI, LAM, h0, delta_h, sigma,
    lat_c, lon_c, Omega, g)
    f_c = 2 * Omega * sin(lat_c);
    [nLat, nLon] = size(PHI);
    u = zeros(nLat, nLon);
    v = zeros(nLat, nLon);
    h = zeros(nLat, nLon);
    for i = 1:nLat
        for j = 1:nLon
            dlat = PHI(i,j) - lat_c;
            dlon = LAM(i,j) - lon_c;
            dlon = mod(dlon + pi, 2*pi) - pi;
            r = sqrt(dlat^2 + dlon^2);
            h(i,j) = h0 - delta_h * exp(-(r/sigma)^2);
            if r < 1e-6
                u(i,j) = 0;
                v(i,j) = 0;
            else
                dhdr = (2 * delta_h / sigma^2) * r * exp(-(r/sigma)^2);
                V = (-f_c * r + sqrt((f_c * r)^2 + 4 * g * r * dhdr)) / 2;
                u(i,j) = -V * (dlat / r);
                v(i,j) =  V * (dlon / r);
            end
        end
    end
end
```

This function sets up a balanced vortex (cyclone) on a rotating sphere by initializing the velocity and height fields $u(\phi, \lambda)$, $v(\phi, \lambda)$, and $h(\phi, \lambda)$ such that the pressure gradient, Coriolis, and centrifugal forces are in approximate equilibrium.

- `PHI`, `LAM`: Gridded latitude and longitude arrays.
- `h0`: Mean background fluid depth.
- `delta_h`: Maximum depression at cyclone center.
- `sigma`: Radial scale (spread) of the pressure anomaly.
- `lat_c`, `lon_c`: Latitude and longitude of cyclone center.
- `Omega`, `g`: Planetary rotation rate and gravitational acceleration.

1. Coriolis Parameter at Center We begin by calculating the Coriolis parameter at the cyclone center:

$$f_c = 2\Omega \sin(\phi_c)$$

2. Loop Over Grid We loop through every point (ϕ_i, λ_j) to compute:

- Radial distance r from cyclone center (with longitude wrapping):

$$\Delta\lambda = \mathrm{mod}(\lambda - \lambda_c + \pi, 2\pi) - \pi$$

$$r = \sqrt{(\phi - \phi_c)^2 + (\Delta\lambda)^2}$$

- The fluid depth field is initialized via a Gaussian depression:

$$h(\phi, \lambda) = h_0 - \Delta h \cdot \exp\left(-\frac{r^2}{\sigma^2}\right)$$

- The pressure gradient (related to the radial height derivative) is:

$$\frac{dh}{dr} = \frac{2\Delta h}{\sigma^2} \cdot r \cdot \exp\left(-\frac{r^2}{\sigma^2}\right)$$

- Using gradient wind balance, the tangential velocity V is:

$$V = \frac{-f_c r + \sqrt{(f_c r)^2 + 4gr\frac{dh}{dr}}}{2}$$

- Convert tangential velocity into zonal (u) and meridional (v) components:

$$u = -V \cdot \frac{\Delta\phi}{r}, \quad v = V \cdot \frac{\Delta\lambda}{r}$$

If $r < 10^{-6}$, then set $u = v = 0$ to avoid division by zero.

Outcome: A rotationally balanced depression whose winds spiral cyclonically and decay with distance.

6.10. Interactive Globe Visualization

```
function [globe, lat_rad, lon_rad, x, y, z] = interactive_globe(texture_path)
    lat_res = 0.3;
    lon_res = 0.3;
    lat = -90:lat_res:90;
    lon = 0:lon_res:360 - lon_res;
    [lon_rad, lat_rad] = meshgrid(deg2rad(lon), deg2rad(lat));
    R = 1;
    x = R * cos(lat_rad) .* cos(lon_rad);
    y = R * cos(lat_rad) .* sin(lon_rad);
    z = R * sin(lat_rad);
    if nargin < 1 || isempty(texture_path)
        texture_path = 'textures/earth_reg_10k.jpg';
    end
    try
        earth_img = imread(texture_path);
        if size(earth_img, 3) == 1
            earth_img = repmat(earth_img, 1, 1, 3);
        end
        earth_img = imresize(earth_img, [length(lat), length(lon)]);
        figure('Color', 'k');
        globe = surf(x, y, z, flipud(earth_img), 'EdgeColor', 'none', 'FaceColor', '
    texturemap');
    catch
        figure('Color', 'k');
        globe = surf(x, y, z, 'EdgeColor', 'none', 'FaceColor', [0.3, 0.5, 0.9]);
    end
    shading interp;
    axis equal off;
    light('Position', [1 0 1], 'Style', 'infinite');
    lighting gouraud;
    material dull;
    view(160, 20);
    rotate3d on;
    title('Interactive Earth Globe', 'Color', 'w', 'FontSize', 14);
end
```

This function creates a visual representation of the Earth using a spherical grid and applies a realistic texture map.

- Define latitudes and longitudes from $-90°$ to $90°$ and $0°$ to $360°$ respectively, using fine resolution $(0.3°)$.

- Convert to radians and generate a spherical mesh using:

$$x = R\cos(\phi)\cos(\lambda), \quad y = R\cos(\phi)\sin(\lambda), \quad z = R\sin(\phi)$$

- Load and apply Earth texture (e.g., `earth_reg_10k.jpg`). If texture fails, fallback to a blue-colored sphere.
- Add 3D shading, lighting, and interactive rotation for a polished, immersive globe.

This globe provides the geographical context for the fluid simulation by rendering the Earth as a spherical surface.

6.11. Converting Spherical Velocity to Cartesian Vectors

```
function [vx, vy, vz] = velocity_sphere_to_cart(u, v, lat, lon)
    sinLat = sin(lat);
    cosLat = cos(lat);
    sinLon = sin(lon);
    cosLon = cos(lon);
    ex = -sinLon;
    ey = cosLon;
    ez = zeros(size(lat));
    nx = -cosLon .* sinLat;
    ny = -sinLon .* sinLat;
    nz = cosLat;
    vx = u .* ex + v .* nx;
    vy = u .* ey + v .* ny;
    vz = u .* ez + v .* nz;
end
```

This function transforms the zonal (u) and meridional (v) velocity components defined in spherical coordinates into Cartesian vectors for 3D quiver visualization.

Local Unit Vectors: Define the local tangent vectors on the sphere:

$$\vec{e}_\lambda = (-\sin\lambda, \cos\lambda, 0)$$
$$\vec{e}_\phi = (-\cos\lambda\sin\phi, -\sin\lambda\sin\phi, \cos\phi)$$

Velocity Conversion:

$$\vec{v}_{\text{cartesian}} = u \cdot \vec{e}_\lambda + v \cdot \vec{e}_\phi$$

This gives the final components:

$$v_x = u \cdot (-\sin\lambda) + v \cdot (-\cos\lambda \sin\phi)$$
$$v_y = u \cdot \cos\lambda + v \cdot (-\sin\lambda \sin\phi)$$
$$v_z = v \cdot \cos\phi$$

Purpose: These Cartesian vectors can be directly used with `quiver3` to animate wind flow arrows over the spherical Earth.

6.12. Diagnostics Evaluation

After each step, global quantities are integrated over the sphere using appropriate weights.

Listing 11: Diagnostics Computation Functions

```
function M = compute_total_mass(h, PHI, dLat, dLon)
    M = sum(sum(h .* cos(PHI))) * dLat * dLon;
end
function E = compute_total_energy(h, u, v, g, PHI, dLat, dLon, b)
    KE = 0.5 * h .* (u.^2 + v.^2);
    PE = g * h .* (b + 0.5 * h);
    E = sum(sum((KE + PE) .* cos(PHI))) * dLat * dLon;
end
```

Explanation: These diagnostics ensure that physical principles such as mass and energy are conserved, or change only due to external forcing.

7. Results and Discussion

This section presents the outcomes of systematic parameter variations applied to the shallow water model, with the goal of understanding how key physical parameters affect cyclone evolution, structure, intensity, and conservation behavior. By isolating one variable at a time while holding others fixed, we are able to diagnose the individual impact of each process on the overall system dynamics.

The four parameters explored are:

- Friction coefficient (γ) — surface drag or internal viscosity.
- Relaxation rate (ε) — vertical adjustment toward equilibrium height.
- Solar forcing amplitude (Q_{amp}) — energy injection mimicking solar heating.

- Background fluid depth (h_0) — base state energy level and wave propagation speed.

Each subsection presents a visual comparison, diagnostic time series, and cumulative energy budget for that parameter variation. All simulations are run with 100,000 steps at a time step of $\Delta t = 10^{-6}$.

7.1. Baseline Simulation

We begin by establishing a control case using baseline values:

- $\gamma = 1$
- $\varepsilon = 1$
- $Q_{\text{amp}} = 1$
- $h_0 = 1.0$

In this baseline case, the cyclone is initialized in geostrophic balance as a compact vortex at 35°N latitude. Over time, it responds to weak solar forcing and moderate damping. The system demonstrates visually coherent vortex structure with stable energy and enstrophy levels.

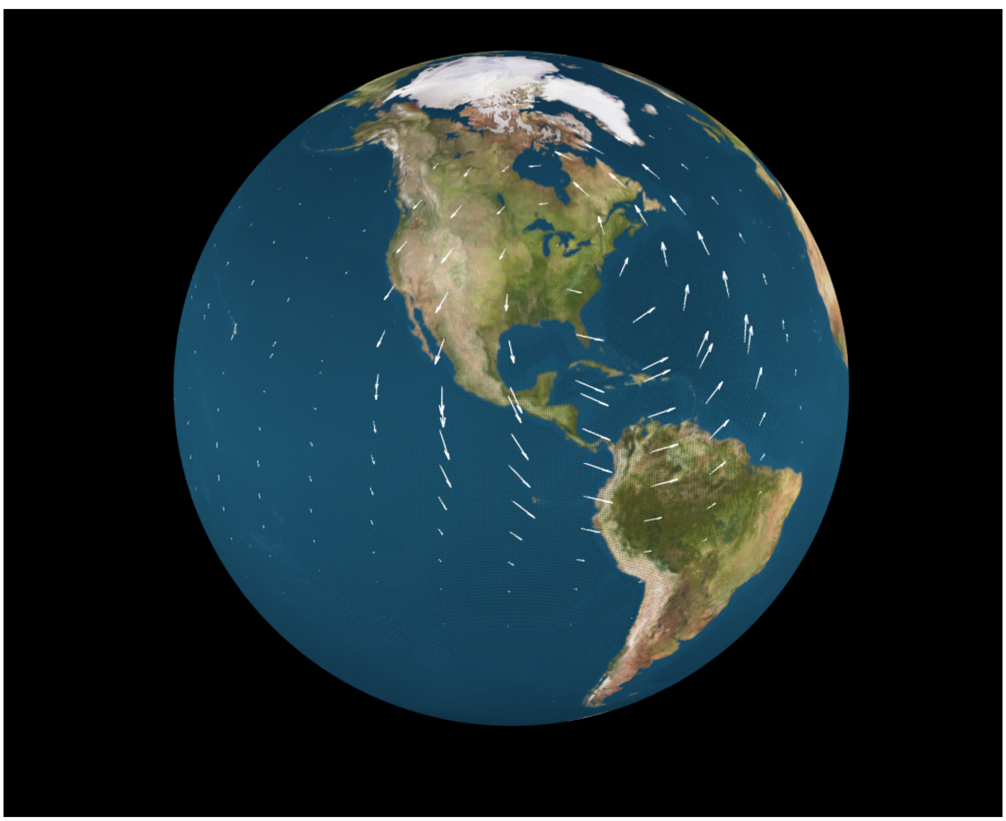

Figure 12: Cyclone evolution at $t = 100{,}000$ for baseline case

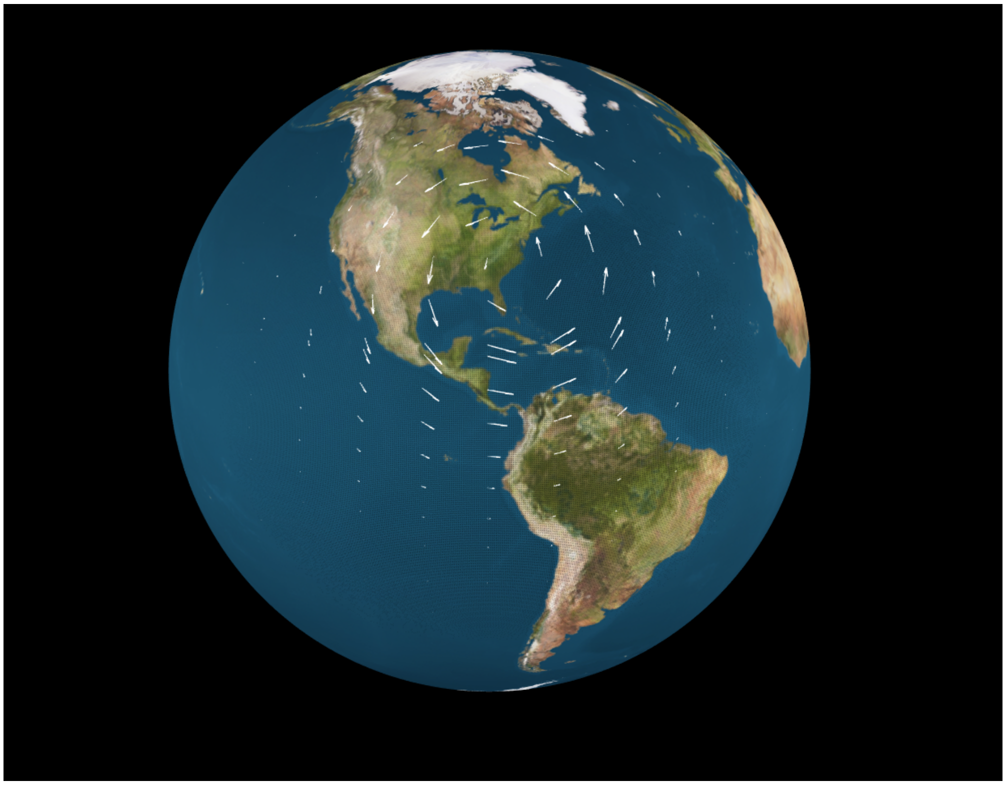

Figure 11: Wind field and height field at $t = 0$ for baseline simulation

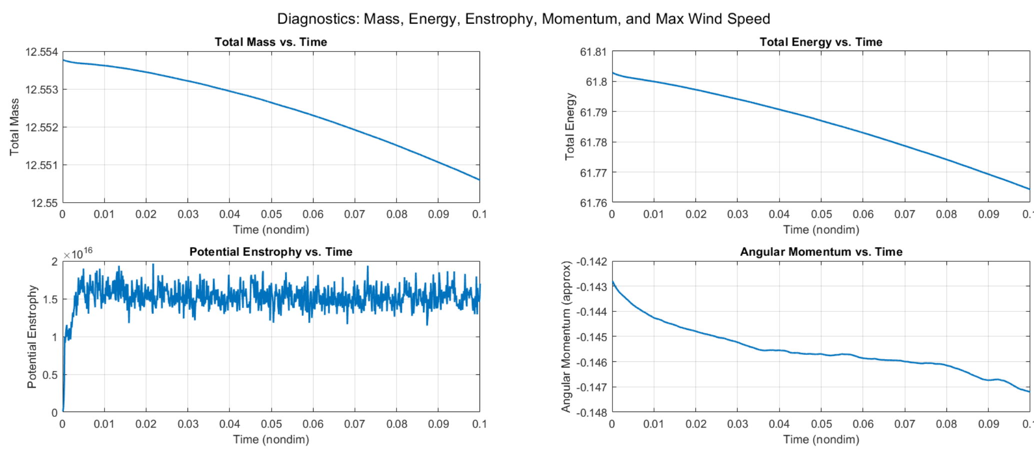

Figure 13: Diagnostics for baseline case: mass, energy, enstrophy, momentum, max wind speed

The system maintains near-constant energy and enstrophy with minimal drift in total mass and momentum. This confirms proper implementation of numerical and physical principles.

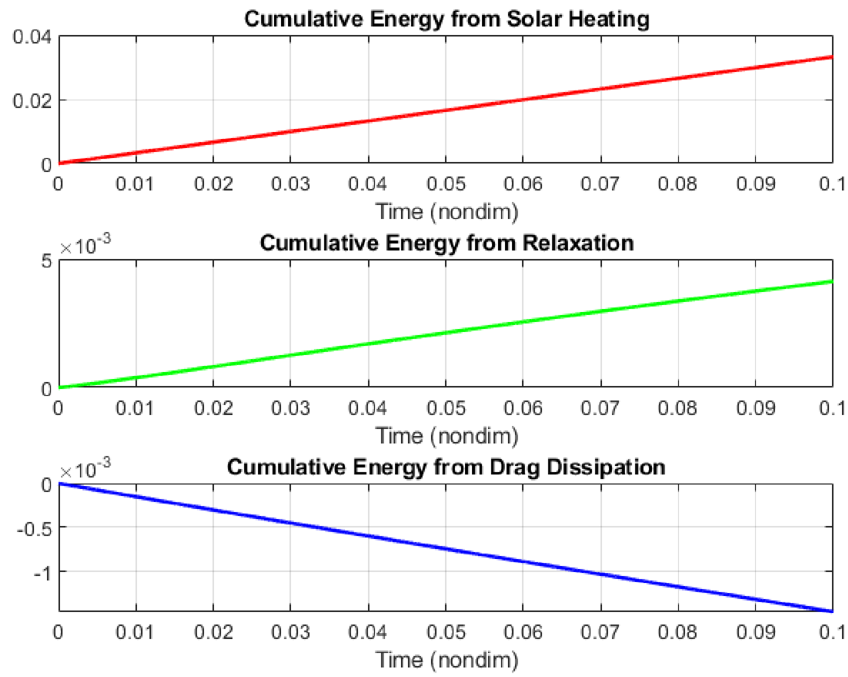

Figure 14: Cumulative energy contributions over time for baseline simulation. Solar heating steadily injects energy, while relaxation and drag act as continuous sinks.

Explanation: Solar heating adds energy to the system at a nearly constant rate, producing the steady upward red line. Both relaxation and drag remain two orders of magnitude smaller; relaxation adds a small amount of potential energy (green), while drag removes a similarly small amount of kinetic energy (blue). Because the sinks almost balance, the net energy growth over the 0.1-unit simulation window is modest, confirming that the baseline setup stays close to energetic equilibrium.

7.2. Effect of Friction Coefficient (γ)

Friction represents momentum loss due to surface drag or internal viscous forces. Increasing γ from 1 to 10 introduces much stronger damping (See Appendix 9.7).

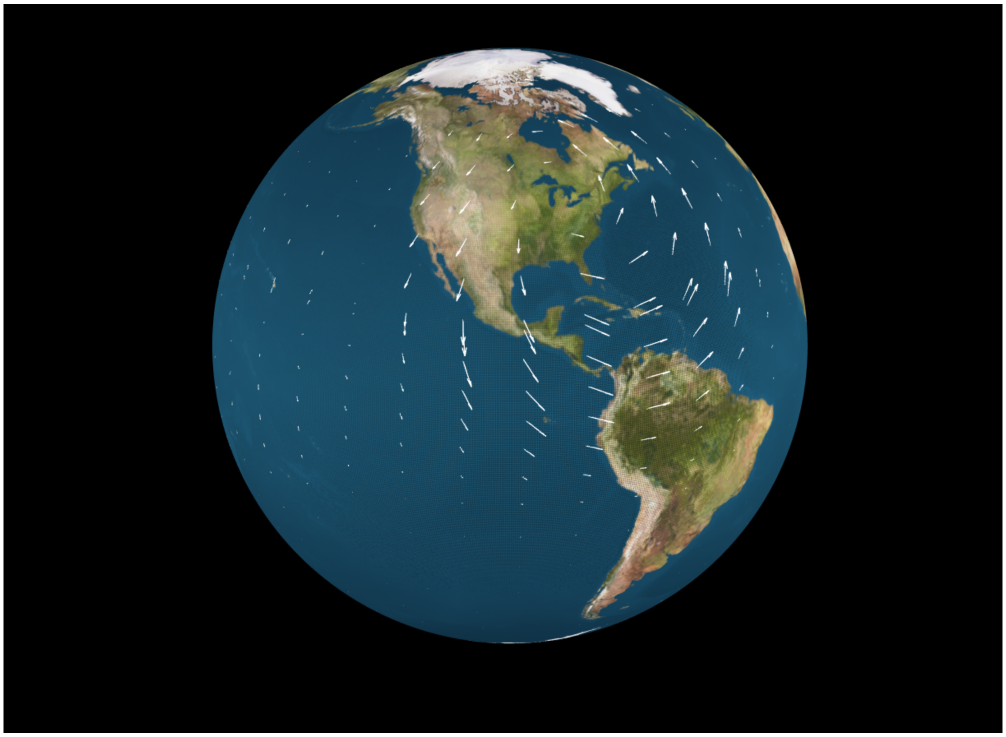

Figure 15: Simulation result with $\gamma = 10$ at final time

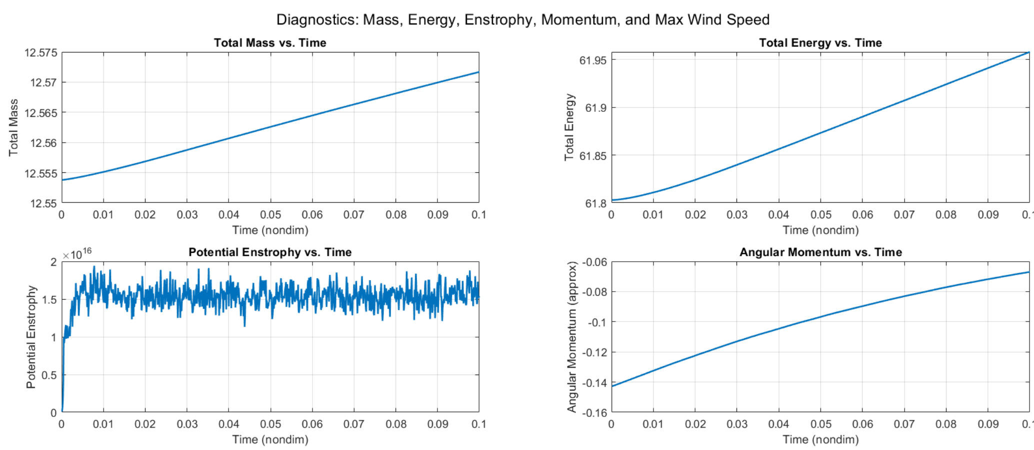

Figure 16: Diagnostics with $\gamma = 10$ at final time

The vortex spreads radially and weakens, as drag converts kinetic energy into heat. The cumulative energy trends shown in Figure 17 illustrate the stronger energy removal associated with increased drag.

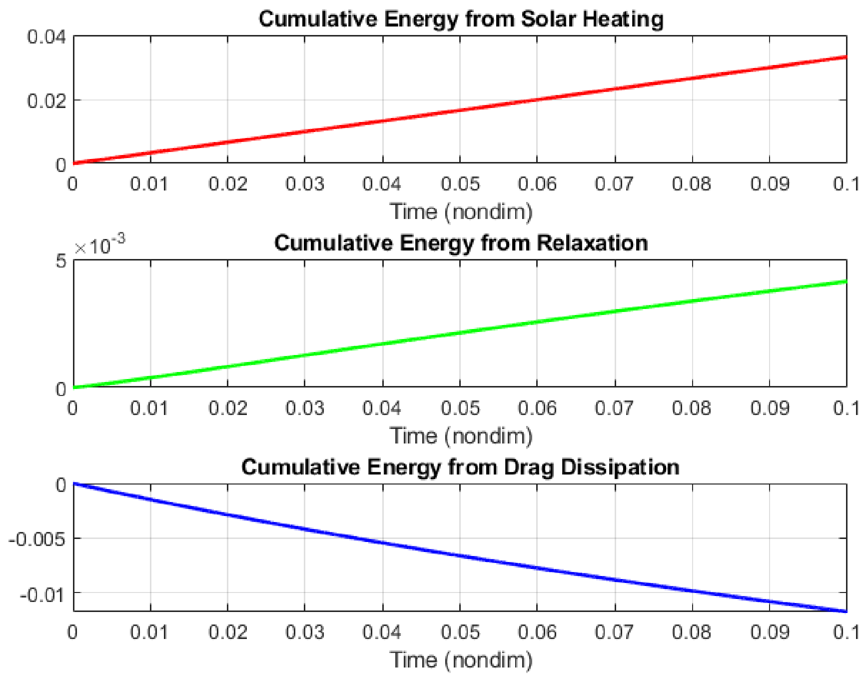

Figure 17: Cumulative energy contributions over time with increased friction ($\gamma = 10$). Drag dissipation dominates, leading to stronger net energy loss.

Explanation: With $\gamma = 10$, the drag curve (blue) steepens markedly, indicating much stronger kinetic-energy dissipation. Solar heating still injects energy at the same rate as the baseline, and relaxation remains small, but the dominant drag sink drives the total budget negative. The stronger drag therefore suppresses cyclone intensity by continuously extracting mechanical energy faster than it is supplied.

7.3. Effect of Relaxation Rate (ε)

The height relaxation rate returns the fluid height toward its equilibrium value (h_0). This simulates vertical damping effects like radiative cooling or convective adjustment [8]. Increasing ε from 1 to 10 accelerates this relaxation process.

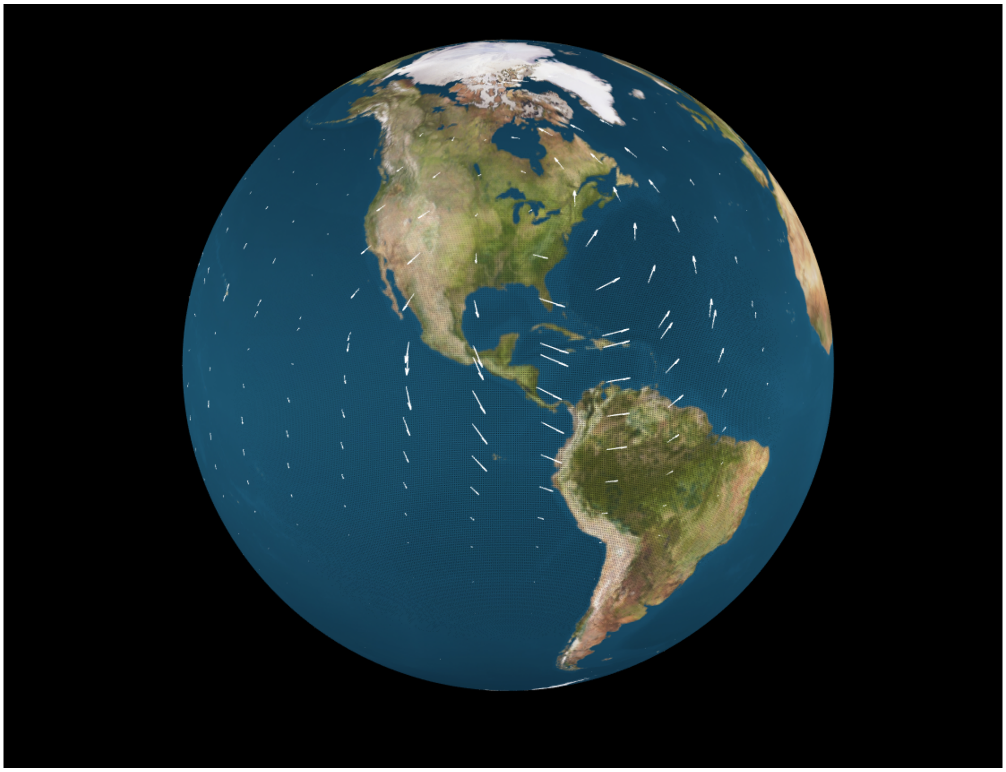

Figure 18: Simulation result with $\varepsilon = 10$ at final time

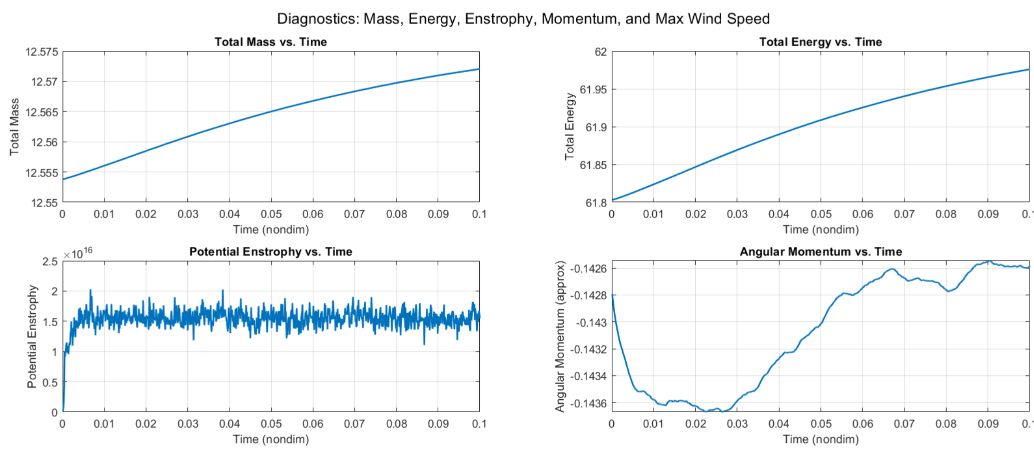

Figure 19: Diagnostics with $\varepsilon = 10$ at final time

Increased relaxation leads to stronger vertical damping and smoother height fields. The corresponding cumulative energy changes are shown in Figure 20.

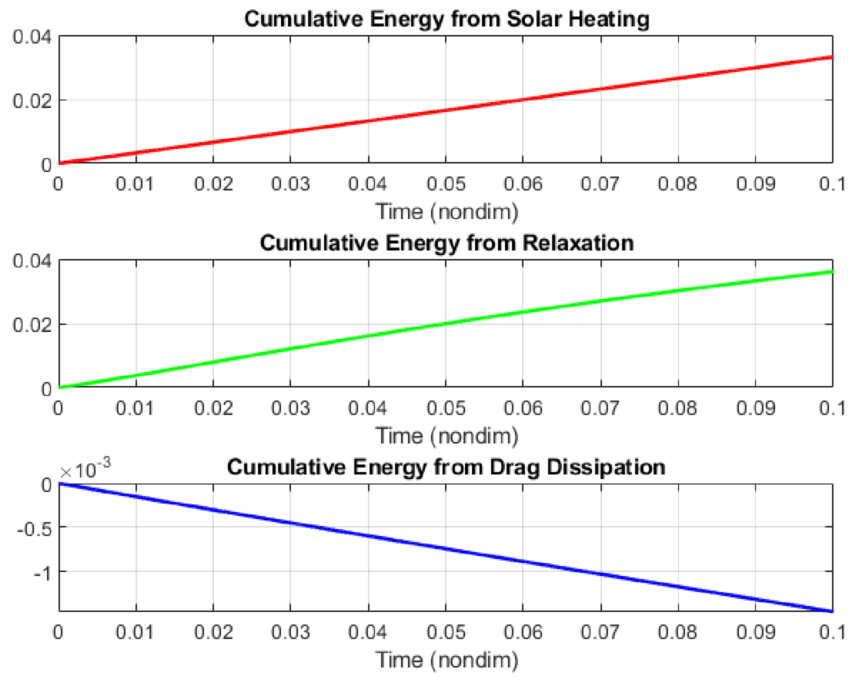

Figure 20: Cumulative energy contributions over time with increased relaxation rate ($\varepsilon = 10$). Relaxation energy sink becomes more significant.

Explanation: Increasing the relaxation rate to $\varepsilon = 10$ makes the relaxation term (green) the largest contributor after solar heating. Here, relaxation *adds* geopotential energy because it continuously restores the fluid column toward the deeper reference height. Drag plays only a minor role, so the overall budget is still positive, but the energy pathway now favors potential (height) adjustments rather than kinetic growth.

7.4. Effect of Solar Forcing Amplitude (Q_{amp})

This parameter governs the amplitude of time-varying solar heating applied as a moving Gaussian. We increase Q_{amp} from 1 to 10 to simulate stronger differential heating across the atmosphere.

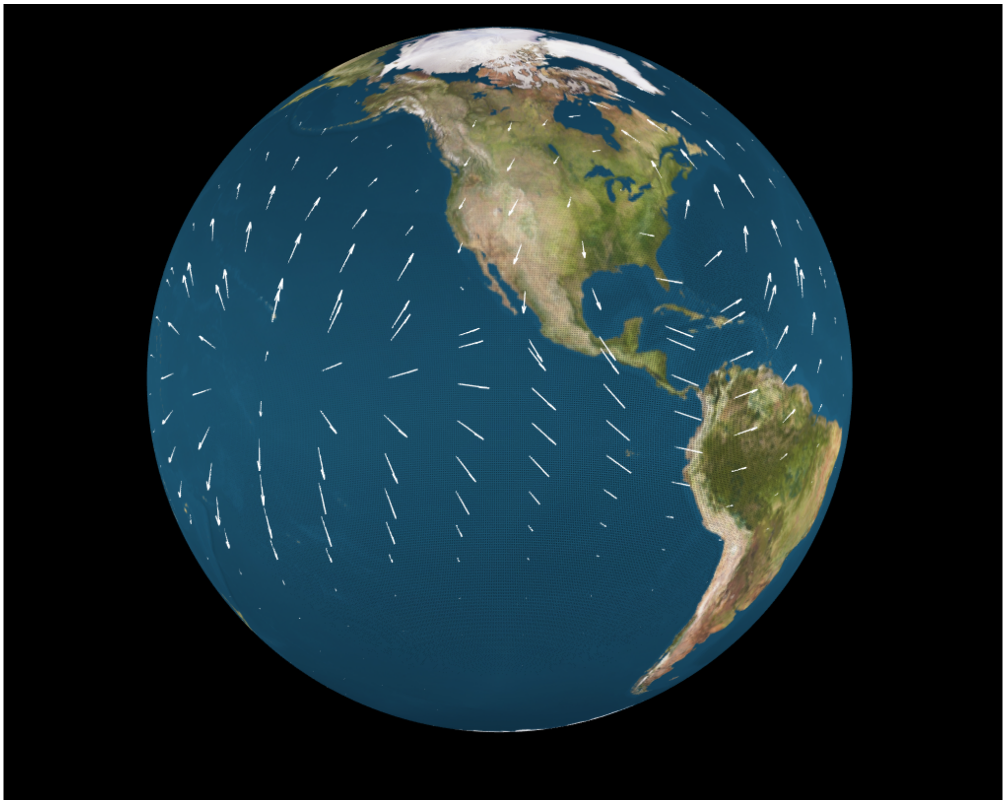

Figure 21: Simulation result with $Q_{\text{amp}} = 10$ at final time

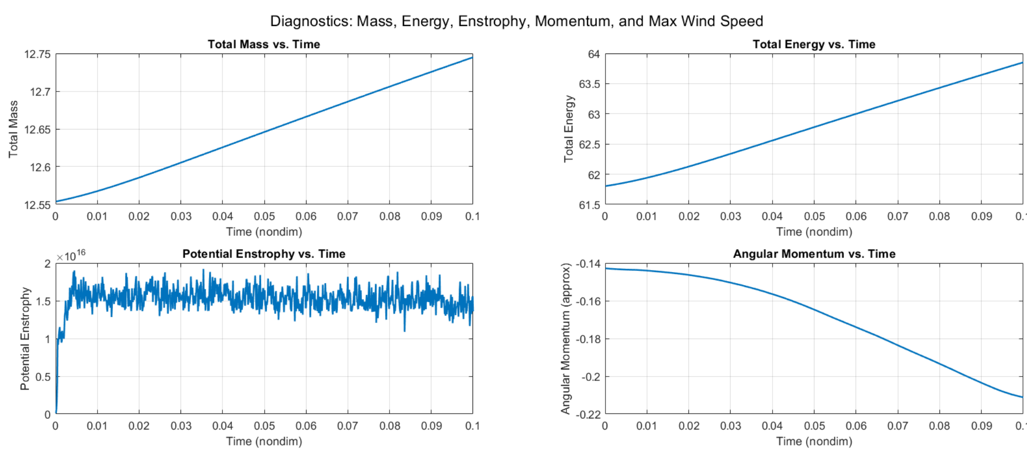

Figure 22: Diagnostics with $Q_{\text{amp}} = 10$ at final time

With stronger solar forcing, cumulative energy input grows much faster, as shown in Figure 23.

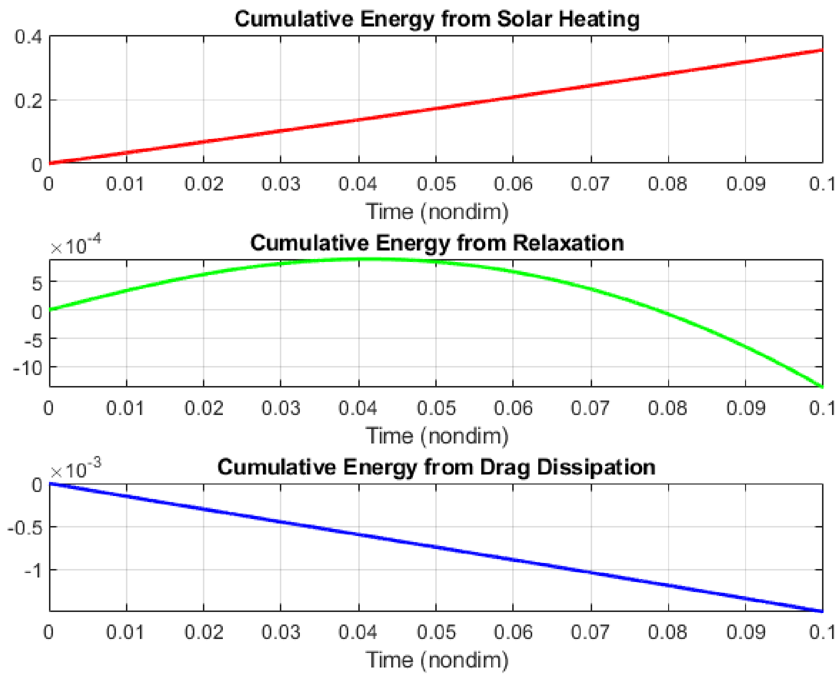

Figure 23: Cumulative energy contributions over time with increased solar forcing ($Q_{\text{amp}} = 10$). Solar heating injection increases substantially.

Explanation: A ten-fold increase in Q_{amp} accelerates solar energy input by an order of magnitude, seen in the steep red line. Relaxation initially rises as the column expands, then turns negative as the scheme begins removing excess height—producing the characteristic hump. Drag remains a comparatively small sink. The large positive area under the solar curve explains the explosive wind growth and loss of vortex coherence noted in the diagnostics.

7.5. Effect of Background Fluid Depth (h_0)

The mean fluid depth h_0 sets the base state energy and modifies wave propagation speed. A deeper fluid supports faster gravity waves and potentially more inertial stability.

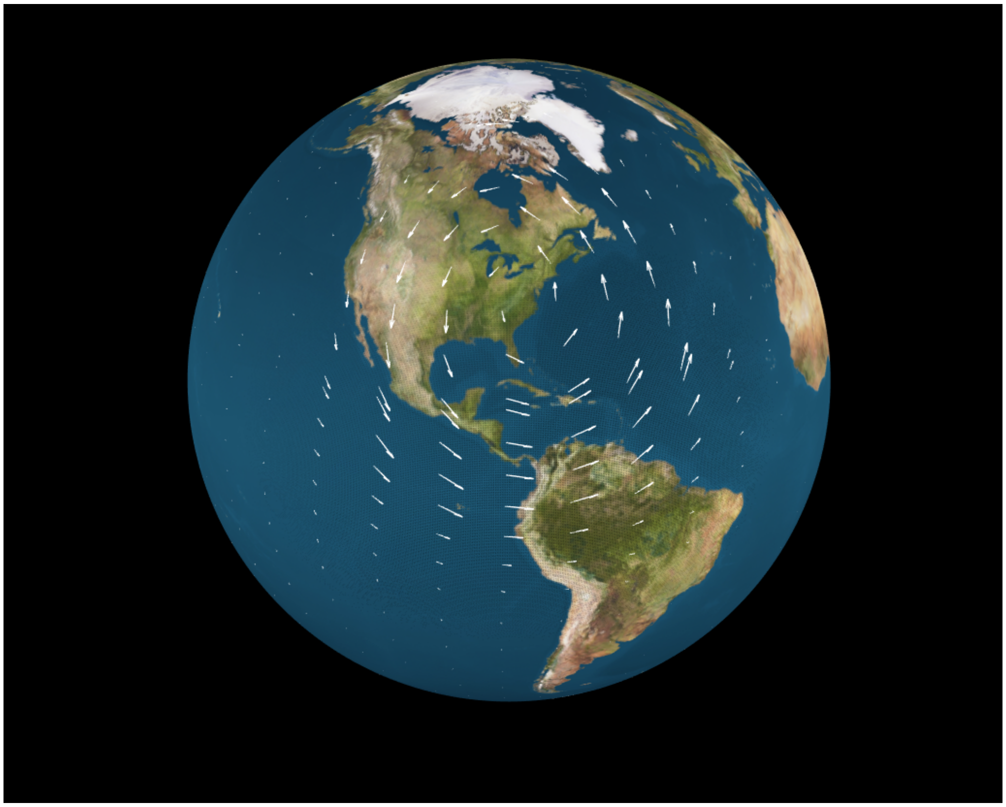

Figure 24: Simulation result with $h_0 = 10$ at final time

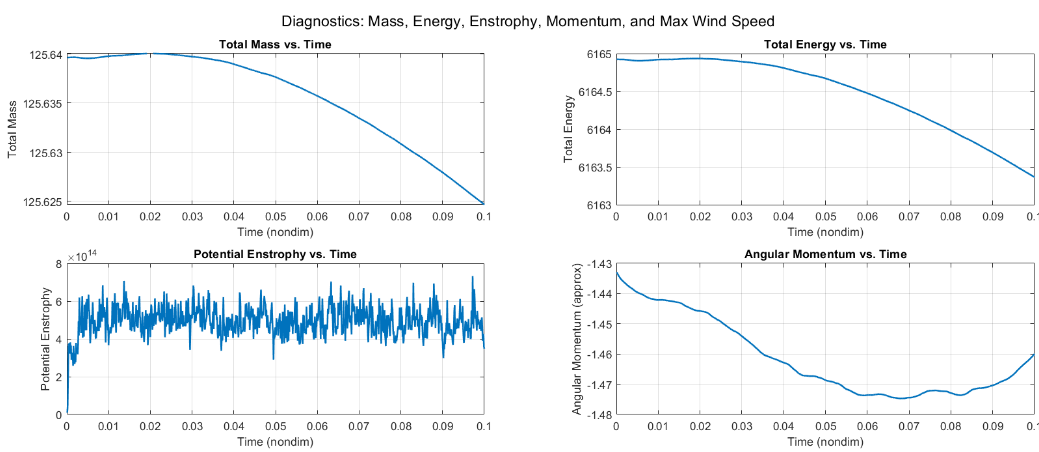

Figure 25: Diagnostics with $h_0 = 10$ at final time

Cumulative energy contributions under deeper fluid conditions are shown in Figure 26.

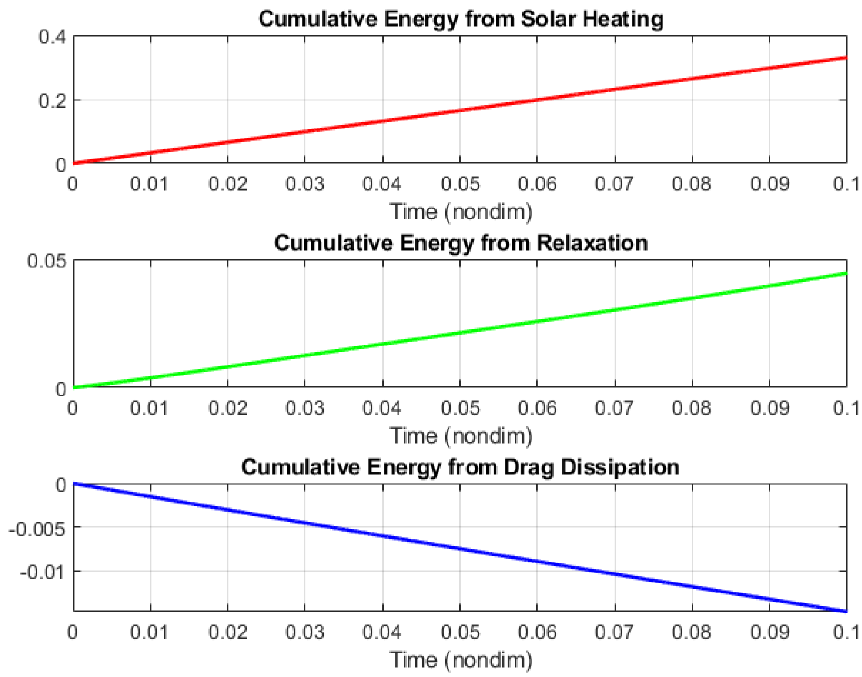

Figure 26: Cumulative energy contributions over time with increased background fluid depth ($h_0 = 10$). Total energy fluxes are moderated by deeper fluid inertia.

Explanation: With a deeper background layer ($h_0 = 10$) the solar term climbs more slowly in *fractional* terms but reaches a larger absolute value because the geopotential of a deeper fluid is higher. Relaxation adds a modest, steady amount of energy, while drag again removes only a small fraction. The gentler slopes reflect the greater inertial stability of a thick layer: energy still accumulates, yet the cyclone's response is buffered by the increased mass.

7.6. Summary of Findings

- **Friction (γ):** Dampens wind intensity and erodes the vortex. Essential for controlling energy growth in study mode.
- **Relaxation (ε):** Flattens height anomalies and stabilizes the energy budget. Helps preserve cyclone shape.
- **Solar Forcing ($Q_{\mathbf{amp}}$):** Injects energy and promotes strong, often chaotic, wind responses. Key for simulating convective activity.
- **Background Depth (h_0):** Increases wave speed and cyclone scale. Deep fluid supports more stable, larger structures.

These results highlight the delicate balance between forcing and dissipation that governs cyclone evolution. The ability to control and predict this balance is central to geophysical fluid modeling.

8. Conclusions

This project demonstrated how a simplified, one-layer shallow water model on a rotating sphere can effectively reproduce essential features of mid-latitude cyclone dynamics. By incorporating key physical ingredients—such as the Coriolis force, pressure gradients, solar forcing, topography, and friction—the model captured the evolution of a balanced cyclone under varying environmental conditions.

Our validation tests confirmed that the model conserves mass and potential enstrophy with high fidelity, and that energy and angular momentum are reasonably well-behaved under study-mode forcing. The use of the Symplectic Euler integration method proved especially important for maintaining numerical stability and long-term realism in simulations involving rotating geophysical fluids.

Systematic parameter sweeps revealed that:

- Increased friction damps cyclonic strength and dissipates kinetic energy.
- Higher relaxation rates smooth out height anomalies and stabilize the system.
- Stronger solar forcing injects energy rapidly, often destabilizing cyclone structure.
- Deeper background fluid supports larger, more stable vortices with slower evolution.

These results highlight the delicate interplay between forcing and damping in atmospheric systems. Despite its simplicity, the shallow water model reveals valuable insights into cyclone behavior, structural integrity, and the importance of balance in large-scale atmospheric flows.

Future directions include extending the model to a multilayer system for better vertical structure, introducing moisture and latent heat release, or coupling to realistic climate datasets for data-driven initialization. With further development, this framework can serve as both an educational tool and a stepping stone toward more comprehensive weather and climate modeling efforts.

9. Appendix

This appendix expands on the mathematical and physical concepts used throughout the simulation.

9.1. Spherical Coordinates for the Earth

In geophysical modeling, the Earth is often idealized as a perfect sphere of radius R, despite its slight equatorial bulge (i.e., being an oblate spheroid). In this approximation, locations on Earth's surface are expressed in terms of spherical coordinates [19]:

$$(x, y, z) = (R\cos\varphi\cos\lambda,\ R\cos\varphi\sin\lambda,\ R\sin\varphi)$$

where:

- φ is the latitude, measuring the angle from the equator (in radians), ranging from $-\frac{\pi}{2}$ (South Pole) to $+\frac{\pi}{2}$ (North Pole),
- λ is the longitude, measuring the angle from the prime meridian, ranging from 0 to 2π (or equivalently from $-\pi$ to π),
- R is the radius of the Earth (typically $\approx 6.371 \times 10^6$ m, though we often normalize $R = 1$ in simulations for nondimensional simplicity).

Latitude and Longitude in Simulations

In computational simulations, latitude and longitude are usually defined over uniform grids:

- Latitude grid: $\varphi_i = -\frac{\pi}{2} + i \cdot \Delta\varphi$, for $i = 0, 1, \ldots, N_\varphi$,
- Longitude grid: $\lambda_j = j \cdot \Delta\lambda$, for $j = 0, 1, \ldots, N_\lambda$,

where $\Delta\varphi$ and $\Delta\lambda$ are angular steps in radians (converted from degrees via `deg2rad()` in MATLAB). For instance:

$$\Delta\varphi = \Delta\lambda = \texttt{deg2rad}(2^\circ)$$

Numerical Considerations

While using spherical coordinates in simulation, several important computational and physical nuances must be addressed:

1. Metric Terms and Coriolis Force

Quantities like velocity divergence, vorticity, and gradients require metric corrections due to spherical geometry. For example:

$$\frac{\partial}{\partial x} \to \frac{1}{R\cos\varphi}\frac{\partial}{\partial\lambda}, \qquad \frac{\partial}{\partial y} \to \frac{1}{R}\frac{\partial}{\partial\varphi}$$

These ensure that distances along the sphere are correctly interpreted, especially since the spacing between longitude lines shrinks near the poles [21].

2. Area Elements

The differential area on the sphere [4] is:

$$dA = R^2 \cos\varphi \, d\lambda \, d\varphi$$

In discretized form, this affects the computation of mass, energy, momentum, and enstrophy integrals [11]:

$$\text{Total mass} = \sum_{i,j} h_{i,j} \cos\varphi_i \cdot \Delta\lambda \cdot \Delta\varphi$$

3. Coordinate Wrapping

Longitudes must wrap around modulo 2π, especially when computing derivatives using circular finite differences. This is managed using:

```
circshift(A, [0, ±1])
```

to enforce periodic boundary conditions in longitude.

4. Polar Singularities

Near the poles ($\varphi \to \pm\frac{\pi}{2}$), certain expressions (e.g., $\frac{1}{\cos\varphi}$) diverge. To handle this:

- Arrows may be downsampled or smoothed near poles,
- Physical variables (like height or velocity) are filtered or blended with adjacent latitudes,
- Some models exclude the exact pole points from the grid entirely.

9.2. Cartesian Projection of the Zonal and Meridional Wind Vectors

In geophysical fluid simulations on a spherical domain, velocity fields are often represented in **spherical coordinates** as:

- u: the **zonal wind component**, directed eastward (along lines of constant latitude, increasing longitude),

- v: the **meridional wind component**, directed northward (along lines of constant longitude, increasing latitude).

However, for 3D visualization (e.g., using `quiver3` in MATLAB), it is necessary to project these components onto the Cartesian coordinate system:

$$(x, y, z) = (R \cos\varphi \cos\lambda,\ R \cos\varphi \sin\lambda,\ R \sin\varphi)$$

where R is the radius of the sphere, λ is longitude, and φ is latitude (both in radians).

Local Tangent Basis Vectors

To project u and v into 3D space, we define two orthonormal tangent vectors at each point on the sphere:

$$\mathbf{e}_\lambda = \begin{bmatrix} -\sin\lambda \\ \cos\lambda \\ 0 \end{bmatrix} \quad \text{(East / Zonal)} \qquad \mathbf{e}_\varphi = \begin{bmatrix} -\cos\lambda \sin\varphi \\ -\sin\lambda \sin\varphi \\ \cos\varphi \end{bmatrix} \quad \text{(North / Meridional)}$$

Conversion to Cartesian Velocity Vectors

Given zonal and meridional components u and v, the corresponding 3D Cartesian velocity vector $\vec{v}_{\text{cart}}$ is computed as:

$$\vec{v}_{\text{cart}} = u \cdot \mathbf{e}_\lambda + v \cdot \mathbf{e}_\varphi$$

This yields:

$$\begin{aligned} v_x &= u \cdot (-\sin\lambda) + v \cdot (-\cos\lambda \sin\varphi) \\ v_y &= u \cdot (\cos\lambda) + v \cdot (-\sin\lambda \sin\varphi) \\ v_z &= 0 + v \cdot (\cos\varphi) \end{aligned}$$

9.3. Modeling Topography with the Use of Gaussian Distributions

In geophysical fluid dynamics, incorporating realistic or idealized topography is essential to simulate terrain effects such as orographic lift, mountain-induced gravity waves, and flow channeling. A common and mathematically convenient approach is to model mountains using a **2D Gaussian bump**.

Gaussian Function for Topography

A two-dimensional Gaussian distribution centered at a given location on the sphere is defined as:

$$b(\lambda, \varphi) = b_0 \exp\left(-\frac{(\varphi - \varphi_0)^2 + (\lambda - \lambda_0)^2}{\sigma^2}\right)$$

where:

- $b(\lambda, \varphi)$ is the topographic height (or deviation from sea level),
- b_0 is the maximum height of the mountain,
- φ and λ are the latitude and longitude (in radians),
- φ_0, λ_0 are the coordinates of the mountain center,
- σ controls the radial spread (width) of the mountain.

This expression produces a smooth, bell-shaped surface that decays rapidly away from the peak, mimicking an isolated mountain or volcano.

Why Gaussian?

The Gaussian shape is used due to several advantages [12]:

- **Smoothness**: It is infinitely differentiable, avoiding sharp numerical discontinuities.
- **Locality**: The values decay quickly to near-zero away from the center, making it ideal for isolated features.
- **Parameter control**: The amplitude and spread can be independently tuned via b_0 and σ.
- **Analytical convenience**: Gaussian functions are straightforward to integrate and differentiate.

9.4. The Coriolis Parameter

The **Coriolis force** is an apparent force that arises due to observing motion from a **rotating reference frame**—specifically, the rotating Earth. In an inertial frame, objects move in straight lines unless acted upon by a real force. However, on a rotating Earth, such objects appear to deviate from straight paths, an effect attributed to the Coriolis force.

Vector Form of the Coriolis Force

The Coriolis force per unit mass is given by:

$$\mathbf{F}_{\text{Coriolis}} = -2\,\boldsymbol{\Omega} \times \mathbf{v}$$

where:

- $\boldsymbol{\Omega}$ is the Earth's angular velocity vector (pointing from the South to North Pole),
- $\mathbf{v}$ is the velocity of the fluid parcel relative to the rotating Earth,
- $\times$ denotes the vector cross product.

This force acts **perpendicular** to both the rotation axis and the velocity vector, inducing a deflection to the right in the Northern Hemisphere and to the left in the Southern Hemisphere.

Coriolis Parameter f

In geophysical fluid dynamics, the vertical component of the Coriolis force is encapsulated in the scalar **Coriolis parameter** [18]:

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

where:

- $\Omega = 7.2921 \times 10^{-5}\,\text{rad/s}$ is the Earth's angular rotation rate,
- φ is the latitude.

Important Properties of f:

- At the equator ($\varphi = 0^\circ$), $f = 0$: the Coriolis force vanishes.
- At the poles ($\varphi = \pm 90^\circ$), $f = \pm 2\Omega$: the Coriolis effect is strongest.
- $f > 0$ in the Northern Hemisphere, $f < 0$ in the Southern Hemisphere.

Coriolis Terms in Shallow Water Equations

In the rotating shallow water equations, the Coriolis force appears in the momentum equations as:

$$\frac{Du}{Dt} = \ldots + fv$$
$$\frac{Dv}{Dt} = \ldots - fu$$

These terms induce a coupling between the zonal velocity u and meridional velocity v, leading to rotational dynamics. In particular, this coupling is responsible for the formation of **cyclonic** and **anticyclonic** flows and for the balance of pressure gradients in geostrophic flow.

Example: Cyclone Rotation

In a low-pressure system:

- Air accelerates toward the low center due to the pressure gradient force.
- The Coriolis force deflects the inflow:
  - To the right in the Northern Hemisphere: **counterclockwise** rotation.
  - To the left in the Southern Hemisphere: **clockwise** rotation.

Summary Table

Symbol	**Meaning**	**Units**
Ω	Earth's rotation rate	rad/s
φ	Latitude	radians
$f = 2\Omega \sin\varphi$	Coriolis parameter	s^{-1}
$\mathbf{F}_{\text{Coriolis}}$	Apparent force in rotating frame	N/kg (acceleration)

Table 1: Summary of Coriolis-related quantities.

9.5. The Shallow Water Equations

Governing Equations

The **shallow water equations (SWEs)** describe the horizontal flow of a thin, incompressible, hydrostatic fluid layer on a rotating surface. They are derived by vertically integrating the Navier–Stokes equations under the assumption that the vertical scale of motion is much smaller than the horizontal scale (i.e., $H \ll L$) [3].

For a rotating sphere of radius R, the SWEs in spherical coordinates (φ, λ), where φ is latitude and λ is longitude, are given by:

$$\frac{\partial u}{\partial t} = -\left(\frac{u}{R\cos\varphi}\frac{\partial u}{\partial \lambda} + \frac{v}{R}\frac{\partial u}{\partial \varphi} - \frac{uv\tan\varphi}{R}\right) + fv - \frac{g}{R\cos\varphi}\frac{\partial \Phi}{\partial \lambda}, \tag{10}$$

$$\frac{\partial v}{\partial t} = -\left(\frac{u}{R\cos\varphi}\frac{\partial v}{\partial \lambda} + \frac{v}{R}\frac{\partial v}{\partial \varphi} + \frac{u^2\tan\varphi}{R}\right) - fu - \frac{g}{R}\frac{\partial \Phi}{\partial \varphi}, \tag{11}$$

$$\frac{\partial h}{\partial t} = -\frac{1}{R\cos\varphi}\frac{\partial (hu\cos\varphi)}{\partial \lambda} - \frac{1}{R}\frac{\partial (hv)}{\partial \varphi} + Q - \beta(h - h_0). \tag{12}$$

Here:

- u: zonal velocity (east–west),
- v: meridional velocity (north–south),
- h: fluid layer thickness,
- $\Phi = g(h + b)$: geopotential, with b representing topography,
- g: gravitational acceleration,
- $f = 2\Omega\sin\varphi$: Coriolis parameter,
- Q: external mass forcing (e.g., solar heating),
- β: height relaxation rate (Newtonian damping toward h_0),
- Ω: planetary rotation rate.

Term-by-Term Breakdown

1. **Advection**: The nonlinear terms such as $u\partial u/\partial\lambda$ and $v\partial v/\partial\varphi$ account for self-transport of momentum.

2. **Coriolis Force**: The terms $\pm fv$ and $\mp fu$ represent the deflective force arising due to Earth's rotation, which causes cyclonic motion.

3. **Geopotential Gradient**: These pressure gradient terms (involving $\partial\Phi/\partial\lambda$, $\partial\Phi/\partial\varphi$) are responsible for driving flow from high to low pressure regions.

4. **Geometric Terms**: The $\tan\varphi$ terms appear due to the spherical coordinate system and ensure conservation of angular momentum.

5. **Continuity Equation**: Governs mass conservation by linking changes in h to fluxes in u and v, and also includes mass source/sink terms Q and relaxation toward a background depth h_0 [11].

Discretization and Implementation in the Cyclone Simulation

In the MATLAB cyclone simulation, the SWEs are discretized on a finite-difference grid. Specifically:

- **Grid Resolution**: The Earth is divided into grid cells of $\Delta\varphi = \Delta\lambda = 2^\circ$. These are converted to radians and used to compute differential operators.

- **Central Differences**: Operators such as $\frac{\partial}{\partial\lambda}$ and $\frac{\partial}{\partial\varphi}$ are approximated by central finite differences using:
$$\frac{\partial A}{\partial x} \approx \frac{A(x+\Delta x) - A(x-\Delta x)}{2\Delta x}$$
with periodic boundary conditions in λ.

- **Momentum Update**: The momentum equations (10) and (11) are advanced using a symplectic Euler scheme for stability:
$$u^{n+1} = u^n + \Delta t \cdot \frac{du}{dt}, \quad v^{n+1} = v^n + \Delta t \cdot \frac{dv}{dt}$$

- **Continuity Update**: The height field is updated by computing the divergence of mass fluxes in λ and φ directions.

- **Coriolis Amplification**: For visualization purposes, the Coriolis parameter f is optionally amplified (by a factor of 3) to accentuate the rotational effects on the sphere.

- **Topography**: Included via a Gaussian mountain term $b(\lambda,\varphi)$, which is zero in validation mode.

- **Mass Forcing Q**: A localized solar heating term that moves longitudinally and oscillates in latitude, modeled using a Gaussian profile centered around a moving "solar sub-point."

Application to Cyclone Modeling

The simulation initializes with a geostrophically balanced cyclone using a depression in height ($h < h_0$) and matching rotational winds. The SWEs then evolve this initial state under various physics [15].

9.6. Geopotential in the Shallow Water Equations

Definition and Physical Meaning

In geophysical fluid dynamics, the **geopotential** Φ represents the gravitational potential energy per unit mass. It reflects how gravitational acceleration g and vertical position (height) combine to influence the potential energy of a fluid parcel. For the shallow water approximation, where vertical structure is collapsed into a single layer, the geopotential is directly related to the fluid column's depth [8]:

$$\Phi = g(h + b), \tag{13}$$

where:

- g is the gravitational acceleration (e.g., 9.81 m/s^2),
- $h(\lambda, \varphi, t)$ is the dynamic fluid depth (i.e., thickness of the water layer),
- $b(\lambda, \varphi)$ is the bottom topography (height of terrain or seafloor).

Thus, Φ captures the total vertical displacement of the fluid surface above a reference level. In your simulation, this total height varies in space and time and is a central quantity driving atmospheric motion.

Role in the Momentum Equations

The horizontal pressure gradient force in the shallow water equations is derived from gradients in geopotential. In spherical coordinates, the momentum equations include:

$$\frac{\partial u}{\partial t} \sim -\frac{1}{R\cos\varphi}\frac{\partial \Phi}{\partial \lambda}, \tag{14}$$

$$\frac{\partial v}{\partial t} \sim -\frac{1}{R}\frac{\partial \Phi}{\partial \varphi}. \tag{15}$$

These terms accelerate fluid from high geopotential (i.e., thicker or elevated regions) to low geopotential (shallower or lower-lying areas), just like how gravity drives flow downhill.

Importantly, the pressure gradient is scaled differently in the zonal and meridional directions due to spherical geometry (with $\cos\varphi$ appearing in the denominator of the zonal term).

Topography's Contribution

The topographic term $b(\lambda, \varphi)$ represents mountain height or other surface elevations. In this simulation, a Gaussian-shaped mountain is added as:

$$b(\lambda, \varphi) = b_0 \cdot \exp\left(-\frac{(\varphi - \varphi_0)^2 + (\lambda - \lambda_0)^2}{\sigma^2}\right),$$

where:

- b_0: peak mountain height (e.g., 0.1 in nondimensional units),
- φ_0, λ_0: location of the mountain center (e.g., 30°N, 90°E),
- σ: width of the mountain's Gaussian profile.

Topography modifies the geopotential field even if the dynamic fluid depth h is flat. This causes air to flow around elevated regions, mimicking real-world behavior such as mountain-induced wave trains, blocking, orographic lift, and vortex shedding [16].

9.7. Atmospheric Friction and the Role of γ

In geophysical fluid dynamics, **atmospheric friction** refers to the dissipative effects that reduce kinetic energy in the atmosphere over time due to interactions between air parcels and surface roughness, turbulent eddies, or other forms of viscous-like drag. In large-scale models, such friction is parameterized rather than resolved directly due to the complexity and scale separation involved [11].

9.7.1 Friction in the Shallow Water Equations

The shallow water equations (SWE) model a thin layer of fluid (e.g., the atmosphere or ocean) and are often used in meteorological and climate simulations. The momentum equations in spherical coordinates with frictional damping are given by:

$$\frac{Du}{Dt} - vf + \frac{1}{a\cos\varphi}\frac{\partial\Phi}{\partial\lambda} = -\gamma u,$$
$$\frac{Dv}{Dt} + uf + \frac{1}{a}\frac{\partial\Phi}{\partial\varphi} = -\gamma v,$$

where:

- u and v are the zonal and meridional velocity components respectively.
- $f = 2\Omega \sin\varphi$ is the Coriolis parameter.
- $\Phi = g(h + b)$ is the geopotential (with topography b).
- λ and φ are longitude and latitude.
- a is the planetary radius.
- γ is the **friction coefficient**, a free parameter in the simulation.

The terms $-\gamma u$ and $-\gamma v$ represent **linear Rayleigh drag**, a common simplification used in atmospheric modeling to simulate frictional dissipation near the surface or in the boundary layer.

9.7.2 Physical Interpretation of γ

The coefficient γ has units of inverse time (e.g., s^{-1}), and it controls the **rate at which momentum is dissipated**. A higher value of γ means stronger damping—wind velocities decay more rapidly over time. This is crucial in regulating the growth of instabilities, preventing numerical blow-ups, and mimicking the effect of unresolved turbulent drag in the real atmosphere.

9.8. Solar Forcing in the Shallow Water Model

Physical Background: What is Solar Forcing?

Solar forcing refers to the impact of incoming solar radiation (shortwave energy) on the Earth's atmosphere and oceans. In real geophysical systems, solar radiation [7]:

- Varies spatially and temporally due to Earth's rotation and orbit,
- Creates thermal gradients that drive atmospheric circulation,
- Primarily heats the surface, causing air to rise and generating vertical and horizontal motion,
- Is strongest in the tropics and migrates seasonally.

In a **shallow water framework**, where vertical resolution is compressed into a single fluid layer, we cannot model radiative-convective physics directly. Instead, we translate the heating effects into changes in the fluid layer depth $h(\lambda, \varphi, t)$, representing vertical expansion due to thermal forcing.

Simplified Representation: Forcing Term in Continuity Equation

In this simulation, solar forcing appears as a source term $Q(\lambda, \varphi, t)$ in the continuity equation for fluid height:

$$\frac{\partial h}{\partial t} + \nabla \cdot (h\mathbf{u}) = Q(\lambda, \varphi, t) - \gamma_h (h - h_0), \tag{16}$$

where:

- h is the fluid thickness,
- $\mathbf{u} = (u, v)$ is the velocity field,
- $Q(\lambda, \varphi, t)$ is the heating term (i.e., solar input),
- γ_h is a linear relaxation rate toward background height h_0.

The term Q acts as a local source of "mass" in the fluid column, effectively mimicking vertical expansion from heating. This then alters the geopotential $\Phi = g(h + b)$, and thus drives pressure gradient forces in the momentum equations.

Mathematical Model of Q

The heating term is modeled as a 2D moving Gaussian function on the spherical surface:

$$Q(\lambda, \varphi, t) = Q_0 \exp\left[-\left(\frac{\lambda - \lambda_s(t)}{\sigma_\lambda} \right)^2 - \left(\frac{\varphi - \varphi_s(t)}{\sigma_\varphi} \right)^2 \right], \tag{17}$$

with:

- Q_0: amplitude of heating (controls intensity),
- $\lambda_s(t)$: solar longitude center (moves with time),
- $\varphi_s(t)$: solar latitude center (oscillates with time),
- $\sigma_\lambda, \sigma_\varphi$: widths of the Gaussian envelope.

This formulation ensures that:

- The solar heat source is spatially localized (like a sunlit patch),
- The solar longitude drifts steadily over time (mimicking Earth's rotation),
- The solar latitude oscillates periodically (mimicking seasonal variation).

9.9. Balancing Cyclone Initial Conditions in the Shallow Water Model

Purpose of the Initial Condition

In geophysical fluid dynamics, a **balanced initial condition** represents a flow that is in near-equilibrium, meaning it satisfies a physical force balance — typically geostrophic or gradient wind balance — between [7]:

- The pressure gradient force,
- The Coriolis force,
- And, optionally, centrifugal or curvature terms (especially for strong vortices).

Starting from a balanced configuration prevents the sudden generation of spurious inertia–gravity waves that can distort the dynamics. For cyclones, this balance ensures the simulated system evolves from a realistic, rotating low-pressure system.

Equation: Gradient Wind Balance on a Sphere

For an axisymmetric cyclone, the relevant balance (in latitude–longitude coordinates) is the **gradient wind balance**, expressed in polar coordinates as [21]:

$$\frac{v^2}{r} + fv = \frac{1}{\rho}\frac{dP}{dr}, \tag{18}$$

which, in shallow water form, becomes:

$$\frac{v^2}{r} + fv = g\frac{dh}{dr}, \tag{19}$$

where:

- v: tangential wind speed,
- r: radial distance from cyclone center,
- f: Coriolis parameter at cyclone center $f_c = 2\Omega \sin \varphi_c$,
- h: fluid height,
- g: gravitational acceleration.

This nonlinear equation is solved at each grid point to compute the corresponding wind field for a given depression in h. It is assumed that the cyclone has radial symmetry for analytical tractability.

Height Field: Gaussian Depression

The cyclone is seeded by defining a fluid height field with a smooth depression:

$$h(r) = h_0 - \delta h \exp\left(-\left(\frac{r}{\sigma}\right)^2\right), \tag{20}$$

where:

- h_0: background (mean) height of fluid,
- δh: depth of the depression (maximum deficit at the center),
- σ: radial scale or "radius" of the depression (in radians),
- r: great-circle distance from cyclone center on the sphere.

This depression mimics a low-pressure region in Earth's atmosphere. The smooth Gaussian ensures no discontinuities in ∇h, which helps numerical stability.

Radial Derivative of Height

The pressure gradient term needed for balance is obtained from the radial derivative [21]:

$$\frac{dh}{dr} = \frac{2\delta h}{\sigma^2} r \exp\left(-\left(\frac{r}{\sigma}\right)^2\right). \tag{21}$$

This derivative is substituted into the gradient wind equation to find the wind speed V tangential to the cyclone.

Solving for Tangential Velocity

The velocity V is derived from solving the quadratic equation:

$$\frac{V^2}{r} + f_c V = g\frac{dh}{dr},$$

which yields:

$$V = \frac{-f_c r + \sqrt{(f_c r)^2 + 4gr\frac{dh}{dr}}}{2}, \tag{22}$$

where the positive root is taken to ensure cyclonic rotation (counterclockwise in Northern Hemisphere) [18].

9.10. The Diagnostic Equations

To assess the accuracy, physical realism, and energy-conserving behavior of the shallow water model on a rotating sphere, we compute a set of diagnostic quantities at every timestep. These quantities include the total mass $\mathcal{M}$, total energy $\mathcal{E}$, potential enstrophy $\mathcal{Z}$, approximate vertical angular momentum $\mathcal{L}$, maximum wind speed $U_{\max}$, and cumulative energy changes from solar heating, relaxation, and drag processes [21].

9.10.1 Total Mass $\mathcal{M}$

The total fluid mass is defined by:

$$\mathcal{M} = \iint h(\lambda, \varphi) \cos\varphi \, d\lambda \, d\varphi$$

Discretized on the grid as [13]:

$$\mathcal{M} \approx \sum_{i,j} h_{i,j} \cos\varphi_i \, \Delta\lambda \, \Delta\varphi$$

This formula ensures the area elements on the sphere are correctly scaled by $\cos\varphi$ [22].

9.10.2 Total Energy $\mathcal{E}$

The total energy includes kinetic and potential contributions [21]:

$$\mathcal{E} = \iint \left[\frac{1}{2} h(u^2 + v^2) + gh \left(b + \frac{1}{2} h \right) \right] \cos\varphi \, d\lambda \, d\varphi$$

Where:

- u and v are the zonal and meridional velocities.
- b is topographic height.

9.10.3 Potential Enstrophy $\mathcal{Z}$

Defined as [1]:

$$\mathcal{Z} = \iint \left(\frac{\zeta + f}{h} \right)^2 h \cos\varphi \, d\lambda \, d\varphi$$

Where the relative vorticity ζ is [23]:

$$\zeta = \frac{1}{R\cos\varphi} \frac{\partial v}{\partial \lambda} - \frac{1}{R\cos\varphi} \frac{\partial (u\cos\varphi)}{\partial \varphi}$$

9.10.4 Angular Momentum $\mathcal{L}$

The approximation of vertical angular momentum is [21]:

$$\mathcal{L} = \iint h(\lambda, \varphi)(xv - yu) \cos\varphi \, d\lambda \, d\varphi$$

Where:

$$x = \cos\varphi \cos\lambda, \quad y = \cos\varphi \sin\lambda$$

9.10.5 Cumulative Solar Heating Energy E_{solar}

The energy input from localized solar heating over time is given by [21]:

$$E_{\text{solar}}(t) = \int_0^t \left(\iint \Phi(\lambda, \varphi, t') Q(\lambda, \varphi, t') \cos\varphi \, d\lambda \, d\varphi \right) dt'$$

Where:

- $\Phi = g(h + b)$ is the geopotential.
- Q is the spatially and temporally varying solar heating forcing.

9.10.6 Cumulative Relaxation Sink Energy E_{relax}

The total energy removed through relaxation toward the background fluid height h_0 is [21]:

$$E_{\text{relax}}(t) = \int_0^t \left(\iint \Phi(\lambda, \varphi, t') \left(-\varepsilon (h(\lambda, \varphi, t') - h_0) \right) \cos\varphi \, d\lambda \, d\varphi \right) dt'$$

This diagnostic helps monitor artificial stabilization introduced into the model to maintain system balance.

9.10.7 Cumulative Drag Dissipation Energy E_{drag}

The cumulative energy loss due to surface or atmospheric drag is [21]:

$$E_{\text{drag}}(t) = \int_0^t \left(\iint \left(-\gamma h(\lambda, \varphi, t')(u(\lambda, \varphi, t')^2 + v(\lambda, \varphi, t')^2) \right) \cos\varphi \, d\lambda \, d\varphi \right) dt'$$

Where γ is the drag coefficient. This represents kinetic energy removed by frictional forces.

9.10.8 Significance of Diagnostics

- **Mass** conservation validates numerical flux consistency.
- **Energy** trends show input/dissipation and method stability.
- **Enstrophy** spikes highlight turbulent activity or instabilities.
- **Momentum** helps analyze rotational evolution (e.g., cyclone spin).
- **Cumulative Energies** from solar, relaxation, and drag help track the external energy budget of the system and diagnose balance between forcing and dissipation.

References

[1] Vladimir I. Arnold and Boris A. Khesin. Topological methods in hydrodynamics. *Annual Review of Fluid Mechanics*, 34:145–166, 1982.

[2] Tony Battistel. Understanding the coriolis force. `https://phys420.phas.ubc.ca/p420_12/tony/Coriolis_Force/Home.html`. Accessed: 2025-04-04.

[3] Manuel J. Castro, Sergio Ortega, and Carlos Parés. Well-balanced methods for the shallow water equations in spherical coordinates. *Computers Fluids*, 157:196–207, 2017.

[4] Steven W. Ellingson. Spherical coordinates. In *Electromagnetics I*. LibreTexts, 2019. Accessed: 2025-04-04.

[5] Encyclopaedia Britannica. Latitude and longitude, n.d. Accessed: 2025-04-04.

[6] Ernst Hairer. Lecture 2: Symplectic integrators, 2010. Accessed: 2025-04-04.

[7] Isaac M. Held and Arthur Y. Hou. Nonlinear axially symmetric circulations in a nearly inviscid atmosphere. *Journal of the Atmospheric Sciences*, 41(1):56–77, 1984.

[8] James R. Holton. *An Introduction to Dynamic Meteorology*. Elsevier Academic Press, 4th edition, 2004. See Section 2.6 and Chapter 10 on simplified atmospheric models and damping mechanisms.

[9] National Geographic Society. Coriolis effect, 2023. Accessed: 2025-04-04.

[10] Past Global Changes (PAGES). Solar forcing. `https://pastglobalchanges.org/science/wg/former/solar-forcing/intro`. Accessed: 2025-04-04.

[11] Paul J. Phlips. The shallow water equations. Technical report, Geophysical Fluid Dynamics Laboratory, 2000.

[12] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[13] Mark A. Ringer and Paul R. Williams. Mass and energy conservation in shallow water equations. *International Journal for Numerical Methods in Fluids*, 33(6):751–772, 2000.

[14] Science Education Resource Center, Carleton College. Insolation, differential heating and storm generation, 2016. Accessed: 2025-04-04.

[15] ScienceDirect Editors. Shallow water equation. `https://www.sciencedirect.com/topics/earth-and-planetary-sciences/shallow-water-equation`, n.d. Accessed: 2025-03-26.

[16] Ronald B. Smith. Hydrostatic airflow over mountains. *Advances in Geophysics*, 31:1–41, 1989.

[17] Study.com Contributors. Cyclone formation types — what is a cyclone?, 2022. Accessed: 2025-04-04.

[18] Cornell University. Coriolis force, n.d. Accessed: 2025-04-04.

[19] Mount Royal University. Cylindrical and spherical coordinates, n.d. Accessed: 2025-04-04.

[20] Geoffrey K. Vallis. Atmospheric and oceanic fluid dynamics, 2006. See Chapter 9 and 10: Vorticity Dynamics and Waves in Geophysical Fluids.

[21] Geoffrey K. Vallis. Equations of motion in spherical coordinates. In *Atmospheric and Oceanic Fluid Dynamics: Fundamentals and Large-scale Circulation*, chapter 2.2, pages 61–66. Cambridge University Press, 2017.

[22] James C. Warner and W. Rockwell Geyer. Numerical modeling of momentum dynamics on a rotating sphere. *Ocean Modelling*, 8(1):81–102, 2005.

[23] Vladimir Zeitlin. Nonlinear dynamics of rotating shallow water: Methods and advances. *Applied Mathematical Modelling*, 31(11):2407–2416, 2007.

[24] Michael Zeltkevic. Forward and backward euler methods, 1998. Accessed: 2025-04-04.