Introduction to Numerical Weather Prediction

Introductory Remarks

How can "numerical weather prediction" be defined? Let us consider the dictionary definitions for each term in the phrase, "numerical weather prediction"…

- **Numerical**: using data or numbers; in reference to the methods used (i.e., the solving of mathematical equations).

- **Weather**: the state of the atmosphere with respect to wind, temperature, moisture, pressure, etc.; also referring to the time scale (e.g., in contrast to climate).

- **Prediction**: an act of declaration of a future event; more generally, a forecast of a past, present, or future event.

Thus, we define numerical weather prediction as the use of numerical methods to solve equations governing atmospheric motions and processes to obtain a forecast on the weather spatiotemporal scales. Note that there exist statistical models that may be used to obtain, refine, or otherwise improve upon a forecast. These aren't the types of models considered in this class, however; rather, we consider models that solve dynamical equations.

Numerical (or NWP) models are developed based upon accepted *theories* and rely upon *observations* to provide both initial and verification data so to conduct and verify a forecast. But, how do we solve the equations that govern atmospheric motions and processes? How do we provide the model with observations to initialize (or start) the model forecast? These questions seem as they should have simple answers, but in reality they do not. There are multiple ways of answering each question, each of which has its own strengths and weaknesses. Indeed, the more one comes to find out just how many approximations or simplifications comprise a numerical model, the more one may become surprised at just how well they can perform!

Therein lies a key point – how well they **can** perform. There is an adage that states that you can obtain **any** forecast that you desire using a numerical model, and to some extent that is true. It is altogether possible to run a model and obtain a forecast. The proliferation of desktop computers that can run a numerical model, coupled with the proliferation of freely-available model codes and initialization data, ensure that this occurs many thousands of times per day across the globe. It is far more difficult, however, to obtain a ***robust*** (or reasonably accurate) forecast. For one, doing so requires you to know how the model is formulated, identify the best model configuration options for the forecast challenge at hand, and how to best initialize the model.

Of course, this is all complicated by two factors:

- Despite several decades' worth of advances, we still do not know – and probably never will know – what is *the* best model configuration. It is not immediately clear that one does or should exist.

- Even if we could identify a 'best' configuration, all models provide imperfect forecasts that, even if they verify well, are likely obtaining a reasonable solution for at least partially the wrong reasons. In other words, *the ability to predict does not necessarily imply predictability*!

In this class, it is my goal to expose how models are constructed; the numerical methods used to obtain forecasts and some of the shortcomings inherent to those methods; why and how certain physical processes are parameterized; how numerical models are initialized and (for limited-area models) provided with data on the outermost extent of their simulation domains; and teach you to appropriately configure, run, and analyze the output from a numerical model.

To first order, the primitive equations form the underlying equation set for NWP. It is not as simple as solving the primitive equations in the form that we are most familiar with so as to obtain a forecast, however. We will begin to directly illustrate this in our next lecture. Now, we wish to introduce some general considerations underlying NWP. Much of the remainder of the semester will be spent examining both the fundamental and practical implications of these considerations in the context of NWP for both research and operational forecast purposes.

Discretizing the Fluid Atmosphere

Let us consider the atmosphere and its properties to be a fluid. Indeed, atmospheric dynamics is a subset of fluid dynamics. This fluid, as we know, is comprised of many molecules. How many?

- **Total atmospheric mass**: approximately 5.148×10^{21} g, per the study of Trenberth and Smith (2005, *J. Climate*).
- **Mass of 1 mol of dry air**: 28.97 g
    - **Total mol of air**: 5.148×10^{21} g divided by 28.97 g mol^{-1} = 1.78×10^{20} mol.
- **Number of molecules in a mol**: Avagadro's number, or 6.022×10^{23} mol^{-1}.
    - **Total molecules of air**: 1.78×10^{20} mol multiplied by 6.022×10^{23} mol^{-1} = 1.07×10^{44} molecules.

Even if there were to exist observations of pressure, temperature, wind, and so on at the molecular level globally, we simply cannot solve for this many elements in a reasonable amount of time.

Rather, the atmosphere must be **discretized**, or divided into a smaller number of finite elements. This discretization occurs in all three spatial dimensions and, as we will see, time. The two most popular means of spatial discretization are **grid-based** and **spectral** methods. Grid-based methods discretize the atmosphere into a set of adjacent three-dimensional grid boxes, where the boxes are most commonly in the form of cubes (with alternative formulations of hexagonal or icosahedral prisms also in use in some state-of-the-art models). The latter discretizes the atmosphere into the superposition of a finite number of waves (O(500-1500)) of varying wavelength. Small grid boxes, or a larger number of waves, provide for higher-resolution (or finer discretization) at the cost of greater computational expense. We focus primarily on grid-based methods in this class, although many of the concepts to be covered apply to spectral-based models as well.

Let us consider an example of a grid-point model, the UK Met Office's Unified Model. This model is run four times daily over a global domain. The horizontal grid spacing, or distance between individual grid points in the *x* and *y* dimensions, is approximately 17 km. The model uses 70 non-evenly-spaced levels in the vertical direction between the surface and approximately 80 km above the surface. As a result, the Unified Model discretizes the atmosphere into 1536 x 1152 x 70, or 1.24×10^8 grid points. Indeed, this is a lot of grid points, requiring a supercomputer valued in the tens of millions of dollars on which to run and obtain a forecast in a reasonably short amount of time. However, it is 36 orders of magnitude smaller than the total number of air molecules within the atmosphere.

Model Resolution and Parameterizations

Whether using a grid point, spectral, or other discretization, the atmosphere is discretized to ensure that the smallest meteorological feature of interest – for the desired application of the model and considering available computational resources and time constraints – is adequately **resolved**. Consequently, in the context of NWP, there exist two scales of interest: resolved and unresolved.

What does it mean to resolve a feature? Let's consider this in the context of a grid-point discretization, as that is the most straightforward way in which to do so. Think about shortwaves and longwaves in the geopotential height field, the former with a wavelength of ~500-1000 km and the latter with a wavelength of up to 10000 km. Let us discretize the atmosphere such that there is one grid point every 100 km, as in Fig. 1. For the shortwave, this means that there is a total of about six grid points spanning east to west across the wave – barely enough to identify, crudely, that there is a trough and ridge. For the longwave, however, there are roughly ten times as many grid points spanning east to west across the wave. Thus, the structure of the wave will be better represented, or resolved, for the longwave than for the shortwave.

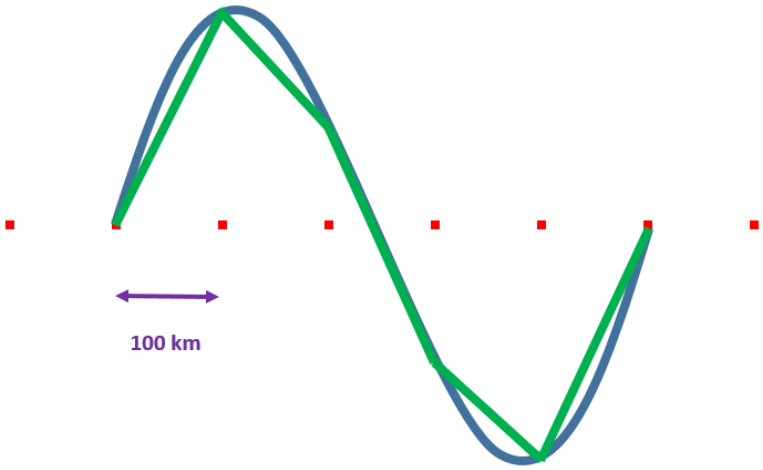

Figure 1. The amplitude of a shortwave trough of wavelength $\lambda = 500$ km (blue) is represented as it would be discretized on a one-dimensional horizontal grid (red) with horizontal grid spacing $\Delta x = 100$ km (green).

For NWP, we solve the primitive equations formally only for the resolved scales of motion. But, though we cannot resolve them, the unresolved scales are not unimportant. Indeed, they are, and on those scales include processes such as turbulent vertical mixing in the planetary boundary layer, water substance phase changes within clouds, and (in many cases) deep, moist convection. Further, the unresolved scales evolve within, and feedback upon, the resolved scales. We account for what occurs on the unresolved scales with **parameterizations**. A parameterization approximates the unresolved scales as a function of known or hypothesized relationships with resolved scales. We will consider parameterizations again briefly later in this lecture and in more detail mid-semester.

For a spectral model, resolved features must have a wavelength longer than the shortest wave retained by the spectral discretization. In grid point models, generally speaking, at least seven grid points are needed to adequately resolve a feature, whether it is wave-like (such as a shortwave) or not. This allows us to establish a distinction between **resolution** and **grid spacing**. These terms, though often used interchangeably, are distinct:

- **Grid spacing**: the distance between two grid elements or points.

- **Resolution**: the scale of the smallest wave/feature that can be resolved on the model grid.

For modern grid point or similarly-constructed models, such as the grid-point-based WRF-ARW (Skamarock 2004, *Mon. Wea. Rev.*) and the unstructured-grid-based finite-volume MPAS (Skamarock et al. 2014, *J. Atmos. Sci.*) models, the **effective resolution** in the horizontal direction is approximately $6\Delta x$ to $7\Delta x$, where Δx is the horizontal grid spacing. In other words, it takes seven to eight grid points to effectively resolve an atmospheric feature, where effective resolution is defined here as the smallest horizontal scale at which the model's kinetic energy spectrum matches that of theory and observations (Fig. 2). Later in the semester, we'll discuss non-physical considerations that exert a control upon a model's effective resolution.

Consider the example of the lake breeze front. Let us assume that the leading edge of the lake breeze front is located just east of Waukesha, WI, approximately 28 km west of Lake Michigan. In mid-spring, the temperature can vary by up to 10-15°C between Waukesha and the Lake Michigan shoreline; the wind direction will also vary in such a scenario, from south-southwest at Waukesha to east-southeast along the Lake Michigan shoreline. This variability in temperature and wind direction is non-uniform, with the sharpest gradients found along the leading edge of the lake breeze front that progressively grow weaker toward the lakeshore.

At a horizontal grid spacing of 4 km, there are eight grid points between Waukesha and the Lake Michigan shoreline. This can crudely resolve the lake breeze. What about a horizontal grid spacing of 28 km? In such a model, the contrast between warmer temperatures with south-southwesterly winds inland and colder temperatures with east-southeasterly winds at the lakeshore would be represented, but the finer-scale detail in between is lost. At even coarser grid spacings, the ability to resolve the lake breeze front is lost.

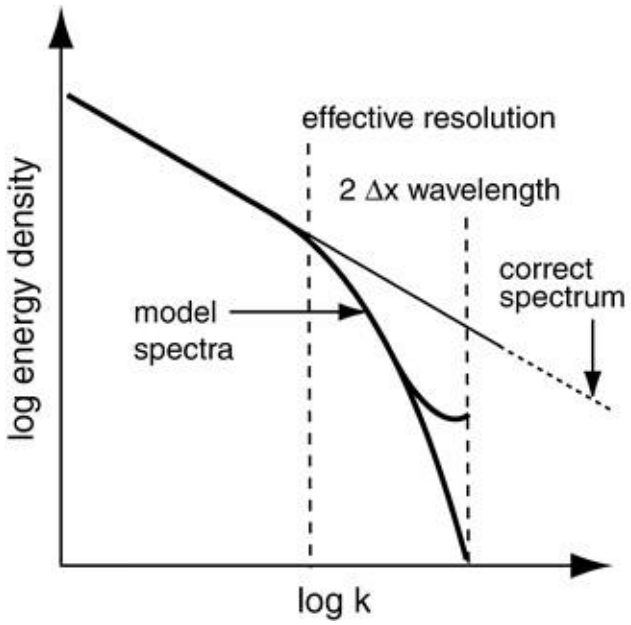

Figure 2. Schematic depicting the kinetic energy spectra for two sets of model simulations: those conducted at horizontal grid spacings in the mesoscale (thick black line with upward-curved tail) and those conducted at horizontal grid spacings on the micro- or synoptic-scale (thick black line that decays exponentially). The *x*-axis depicts the natural logarithm of the wavenumber *k,* such that higher values of log *k* denote smaller spatial scales. The *y*-axis depicts the natural logarithm of the spectral energy density; higher values denote greater kinetic energy. In these examples, the model spectra match the correct spectrum up to the effective resolution but are thereafter damped. Figure obtained from Skamarock (2004, *Mon. Wea. Rev.*), their Fig. 10.

Note that these concepts of discretization, grid spacing, and resolution are, for now, not connected to the available observations used to help initialize a model forecast. Indeed, model discretizations are not identical to those of the observations. How we initialize a model, both generally speaking and in the context of the observations, will be discussed briefly later in this lecture and in more detail over the last 4-5 weeks of the semester.

Discretization can be non-uniform in terms of distance in both the horizontal and the vertical. Variable horizontal discretization, such as that depicted in Fig. 3, is typically employed to provide higher resolution over a region of interest, whether to avoid non-physical influences upon the forecast due to the use of lateral boundary conditions and nested grids or to avoid the computational expense associated with running the model at high resolution over a very large area. We will discuss lateral boundary conditions in more detail both later in this lecture and later in the semester. Certain grid discretizations, such as the latitude-longitude grid, inherently provide variable resolution, with coarser resolution (1° longitude = 111 km) near the Equator and finer resolution (1° longitude < 111 km) at higher latitudes, but also present numerical challenges near the poles.

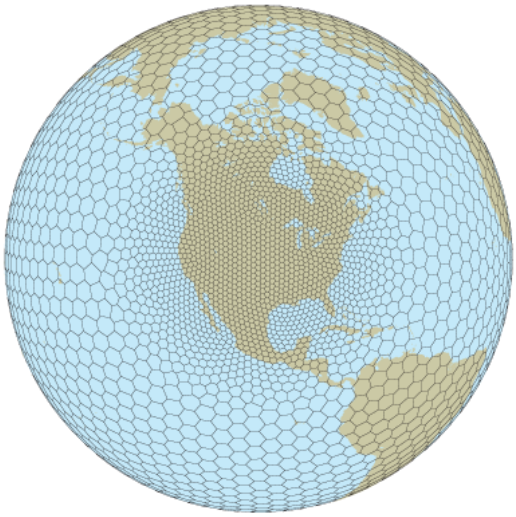

Figure 3. An example of a variable resolution model grid discretization, with higher resolution across North America and lower resolution elsewhere, as employed in the Model for Prediction Across Scales, or MPAS. Image obtained from https://mpas-dev.github.io/.

Most modern models use variable vertical resolution, with finer discretization near the surface and tropopause and coarser discretization in the middle troposphere and stratosphere. This is motivated by the sharp vertical gradients in meteorological quantities typically found between the planetary boundary layer and free atmosphere and along the tropopause. Most modern models utilize some form of a hybrid terrain-following vertical coordinate. Near the surface, the associated vertical levels follow (or contour above) terrain. Starting in the middle to upper troposphere, vertical levels follow constant height, pressure, or potential temperature surfaces depending upon the model; the effect of topography is sufficiently small to not need to use the terrain-following coordinate aloft. An example is given in Fig. 4.

Independent of which vertical coordinate is utilized, the model equations must be transformed from their conventional forms – with height or pressure as the vertical coordinate – to be applicable on the model's chosen vertical coordinate surfaces. We will discuss vertical coordinate systems, including why the height and pressure coordinates are not well-suited for modern NWP, in more detail later this semester.

Modern NWP models use anywhere between 30 and 100 vertical levels, with those having finer horizontal discretization also having finer vertical discretization. As in the horizontal, it takes more than two vertical levels to truly resolve a feature in the vertical, such as an inversion, front, or jet. Another consideration that must be addressed is the definition of the model top. Nominally, this should be at the top of the atmosphere, but in practice the top of the atmosphere is far above the level(s) of meteorological interest on meteorological time scales. The model top should thus be placed above the uppermost level of meteorological interest – generally within the stratosphere, especially on longer time scales – but not so high as to add unneeded computational expense.

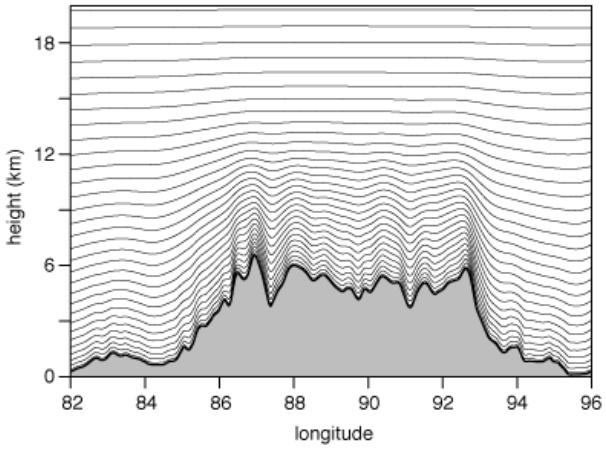

Figure 4. An example of a hybrid terrain-following vertical coordinate over a region of sloped topography. In this example, representative of the vertical coordinate employed in the Model for Prediction Across Scales, vertical levels follow the terrain below approximately 12 km above sea level. At higher altitudes, vertical levels closely resemble constant height surfaces. Image obtained from https://mpas-dev.github.io/.

Numerical Methods

Once we have discretized the atmosphere, whether using a grid, the spectral method, or some other means, how do we solve the primitive equations for that discretization? For spectral models, Fourier series or Fourier-Legendre transforms are used to transform horizontal variability (partial derivatives) between physical and spectral/wave space.

For grid-based models, finite difference approximations to partial derivatives are used to solve the primitive equations. Mathematically speaking, the partial derivative of some generic field *f* with respect to some generic variable *x* can be expressed as:

$$\frac{\partial f}{\partial x} = \lim_{\Delta x \to 0} \frac{\Delta f}{\Delta x} \tag{1}$$

In other words, $\frac{\partial f}{\partial x}$ is equal to the value of $\frac{\Delta f}{\Delta x}$ as Δx approaches (but does not equal) zero. Thus, for small (or finite) values of Δx, we can approximate $\frac{\partial f}{\partial x}$ by $\frac{\Delta f}{\Delta x}$.

Taylor series expansions of a function are used to obtain finite difference approximations. The Taylor series expansion of *f(x)* about $x = b$, where *b* is some generic point, is given by:

$$f(x) = f(b) + f'(b)(x-b) + \frac{f''(b)}{2!}(x-b)^2 + \frac{f'''(b)}{3!}(x-b)^3 + \ldots \tag{2}$$

In other words, *f(x)* is equal to the value of *f(x)* at $x = b$ plus a series of higher-order terms, each of which has a different derivative (primes), exponent on $x - b$, and factorial (!) order.

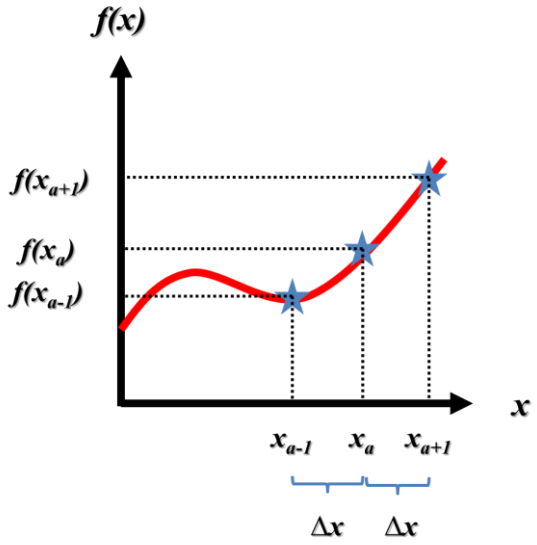

Figure 5. Graphical depiction of a generic function *f(x)* evaluated at three points.

Let us consider the case where $x = x_{a+1}$ and $b = x_a$, as depicted in Fig. 5. The distance $x - b$, or $x_{a+1} - x_a$, is equal to Δx. Conversely, let us consider the case where $x = x_{a-1}$ and $b = x_a$. The distance $x - b$, or $x_{a+1} - x_a$, is equal to $-\Delta x$. Making use of this information, we can expand (2) for each of these two cases:

$$f(x_{a+1}) = f(x_a) + f'(x_a)\Delta x + \frac{f''(x_a)}{2!}(\Delta x)^2 + \frac{f'''(x_a)}{3!}(\Delta x)^3 + \ldots \quad (3)$$

$$f(x_{a-1}) = f(x_a) - f'(x_a)\Delta x + \frac{f''(x_a)}{2!}(\Delta x)^2 - \frac{f'''(x_a)}{3!}(\Delta x)^3 + \ldots \quad (4)$$

Note the similar appearance of (3) and (4) apart from the leading negative signs on the first and third order terms in (4). These arise because $x - b = -\Delta x$ here, as noted above.

From (3) and (4), we are interested in the value of $f'(x_a)$. This is equivalent to $\frac{\partial f}{\partial x}$. We can use (3) and (4) to obtain an expression for this term; we simply need to subtract (4) from (3). Doing so, we obtain the following:

$$f(x_{a+1}) - f(x_{a-1}) = 2f'(x_a)\Delta x + \frac{2f'''(x_a)}{3!}(\Delta x)^3 + \ldots(\text{odd order terms}) \ldots \quad (5)$$

Note how the zeroth and second order terms in (3) and (4) cancel out in this operation. If we rearrange (5) and solve for $f'(x_a)$, we obtain:

$$f'(x_a) = \frac{f(x_{a+1}) - f(x_{a-1})}{2\Delta x} - \frac{f'''(x_a)}{3!}(\Delta x)^2 + \ldots \quad (6)$$

At this point, we wish to neglect all terms higher than the first order term from (6). Doing so, we are left with:

$$f'(x_a) = \frac{f(x_{a+1}) - f(x_{a-1})}{2\Delta x} \quad (7)$$

Equation (7) is what is known as a *centered finite difference*. This expression is said to be *second-order accurate* because terms involving $(\Delta x)^2$ and higher-order powers are dropped, or **truncated**, as being small relative to the retained lower-order terms.

We are likely also already familiar with first-order accurate forward (8) and backward (9) finite differences. The forward difference is obtained by solving (3) for $f'(x_a)$, whereas the backward difference is obtained by solving (4) for $f'(x_a)$:

$$f'(x_a) = \frac{f(x_{a+1}) - f(x_a)}{\Delta x} \quad (8)$$

$$f'(x_a) = \frac{f(x_a) - f(x_{a-1})}{\Delta x} \quad (9)$$

Higher-order (i.e., more accurate) finite difference expressions can be obtained similarly. These involve using even more points to calculate the finite difference approximate to the partial derivative. Modern NWP models typically use fourth-, fifth-, or even sixth-order accurate finite difference approximations. Note that the methods applied for finite differences in time often differ from those for finite differences in space. Further, fully or partially flow-following methods for evaluating advection terms in grid-based models exist.

In the context of finite difference methods, however, each approximation has a unique impact upon the forecast. The most straightforward of these impacts is the concept of **truncation error**, such as that depicted in Fig. 6. Dropping higher-order terms, though they are comparatively small, reduces the accuracy of the approximation. As NWP model forecasts can be quite sensitive to very small differences in initial or forecast values – e.g., chaos theory; Lorenz (1963) and later works – this can impact forecast quality. For grid-based models, truncation error is particularly problematic for features that can be represented but not resolved by the model, or those lying between the grid spacing and effective resolution. For spectral models, truncation error manifests for features with wavelengths smaller than that of the smallest wave represented by the model.

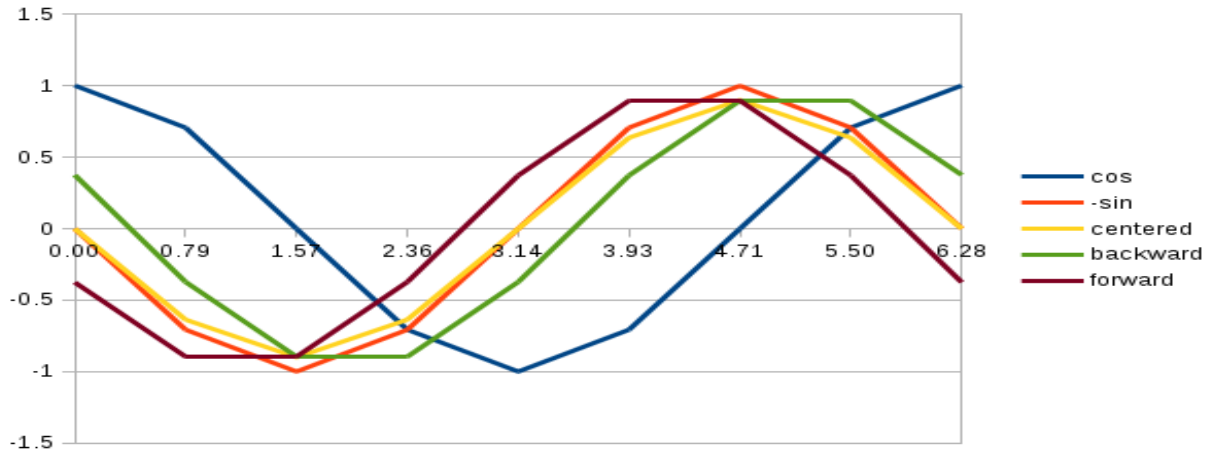

Figure 6. An example of truncation error for the function $f(x) = \cos(x)$ over a grid with nine grid points between 0 and 2π. The function $f(x) = \cos(x)$ is plotted in blue. Its first partial derivative, or $-\sin(x)$, is plotted in red; this is the **exact** solution for $f'(x)$. Three finite difference approximations for the first partial derivative, the centered, forward, and backward approximations, are plotted in yellow, brown, and green, respectively. Note that none provide exact solutions, with the centered approximation best resembling the exact solution. The extent to which each approximation differs from the exact solution can be thought of as the truncation error.

To this point, we have discussed spatial discretization. What about in time, however? Whether for grid point or spectral models, finite differencing is used to advance from one time step to another. As with spatial finite differencing, there exist multiple methods for temporal finite differencing, each with its own accuracy and numerical considerations. Let us start by considering a centered finite differencing scheme in time, known as the "leapfrog" scheme; we will discuss other, more accurate methods later in the semester.

Here, the partial derivative of some function f with respect to time, at time t, is expressed as:

$$\frac{\partial f}{\partial t} = \frac{f_{t+1} - f_{t-1}}{2\Delta t} \tag{10}$$

where this term is then equal to the sum of all forcing terms that act to influence the local value of f. By solving that equation for f_{t+1}, the value of f at the next time can be obtained: here, equal to f_{t-1} plus $2\Delta t$ times the sum of the forcing terms.

How does this work in practice? Consider the schematic in Fig. 7. At $t = 0$, a forward difference is used to go from $t = 0$ to $t = 1$. The value of f at $t = 1$ is equal to that at $t = 0$ plus the forcing upon f (at $t = 0$) multiplied by the timestep Δt. Then, the leapfrog scheme (10) is used to advance to $t = 2$. The value of f at $t = 2$ is equal to that at $t = 0$ plus the forcing upon f (at $t = 1$) multiplied by twice the timestep $2\Delta t$. The value of f at $t = 3$ is then equal to that at $t = 1$ plus the forcing upon f (at $t = 2$) multiplied by twice the timestep $2\Delta t$, and so on.

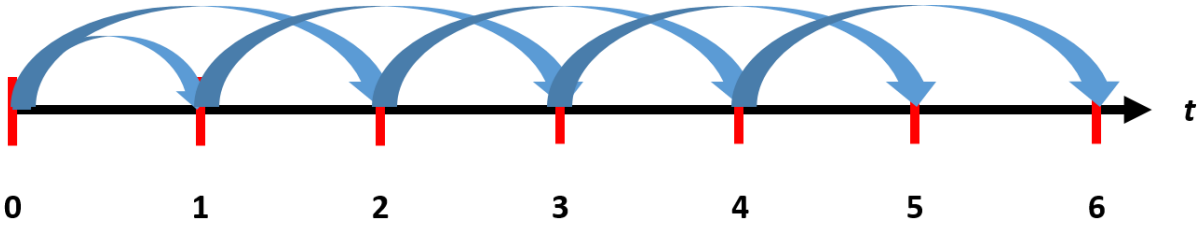

Figure 7. Graphical representation of the leapfrog temporal differencing scheme. Please refer to the text above for details.

Since the model equations are not solved at every possible time, temporal discretization is also a source of truncation error. Atmospheric processes that evolve on shorter timescales than those that the model time step can handle cannot directly be represented by the model equations.

Numerical Stability: The Courant Number and CFL Condition

How is the temporal discretization determined? For spatial and temporal differencing schemes that produce numerically-stable solutions – ones that evolve physically rather than grow exponentially with time during the model integration – there exists a limiting value of the **Courant number**. In its most general of forms, the Courant number is expressed as:

$$C = \frac{U\Delta t}{\Delta x} \tag{11}$$

The Courant number is non-dimensional, where U is the translation speed of the fastest feature or wave on the model grid, Δx is the horizontal grid spacing, and Δt is the model time step. Generally speaking, U is determined by the prevailing meteorology while Δx is determined by the horizontal scales of the features that one desires to resolve on the model grid. Thus, the only free parameter in the Courant number is Δt.

The limiting value of the Courant number is known as the **CFL condition**, representing the maximum value of the Courant number that permits numerically-stable model solutions. The exact value of the CFL condition varies depending on the spatial and temporal finite difference schemes utilized; however, a general guideline is that $C \leq 1$. Conceptually, this implies that an air parcel cannot travel more than one grid distance in one model time step. Given that U and Δx are not free parameters, the CFL condition limits the length of the model time step that may be used.

Though there exist meteorological solutions to the primitive equations, there also exist non-meteorological solutions such as acoustic waves that can move quite rapidly (large U). In modern NWP, however, these waves are either filtered out of the primitive equations (e.g., cannot exist as solutions to the primitive equations) or, more often, are dealt with in an alternative manner. As a result, the value of U is controlled by the prevailing meteorology (e.g., the maximum wind speed within an upper tropospheric jet streak).

Let us consider the example of a jet streak with U = 100 m s^{-1}. On a model grid with horizontal grid spacing of 4 km, for $C \leq 1$, $\Delta t \leq 40$ s. In other words, we must solve the primitive equations across the model grid once every 40 s or less in order to maintain numerical stability. For a one day forecast, this involves 86400 s / 40 s = 2160 separate times that the primitive equations are solved. Over a continental United States domain, with approximately 1150 x 750 x 50 grid points for Δx = 4 km,

- **Number of model grid points**: 1150 x 750 x 50 = 43,125,000
- **Number of model time steps**: 2,160
    - **Total calculations for one variable**: 43,125,000 x 2,160 = 9.315 x 10^{10}.

Contrast this to a model configuration with 12 km grid spacing. For $C \leq 1$, $\Delta t \leq 120$ s, requiring that the primitive equations be solved only 720 times for a one-day forecast. Over the same continental United States domain,

- **Number of model grid points**: 383 x 250 x 50 = 4,787,500
- **Number of model time steps**: 720
    - **Total calculations for one variable**: 4,757,500 x 720 = 3.447 x 10^{9}.

This number is 27 times fewer than that at 4 km. This aside helps to illustrate the limits exerted upon both Δx and Δt by available computational power. Simply going from Δx = 12 km to Δx = 4 km requires 27 times as many computations – three times as many grid points in the x direction multiplied by three times as many grid points in the y direction multiplied by three times as many time steps due to the CFL condition.

Lateral Boundaries

A numerical model represents the atmosphere in three spatial dimensions. If the model is a **global model**, then the horizontal dimensions wrap back upon themselves. This is one instance of what are known as *periodic boundary conditions*, where what goes out one side comes back in on the other side. Other models simulate only a portion of the globe; such models are known as **limited-area models**. The east-west and north-south extents of such models are limited; thus, for such models, we need to know what is coming into and going out of the domain.

This is accomplished using *specified boundary conditions* on the lateral boundaries (where lateral refers to sides). These are generally drawn from the forecast and/or analysis of a global or larger area model – possibly even one you run yourself! Such lateral boundary conditions are typically coarser than the horizontal grid spacing of the limited-area model and, as they are drawn from a different model, typically do not exactly agree with the forecast from the limited-area model. Thus, a transition and damping buffer zone between the lateral boundaries and the interior of the limited-area model domain is generally applied, such as that depicted in Fig. 8.

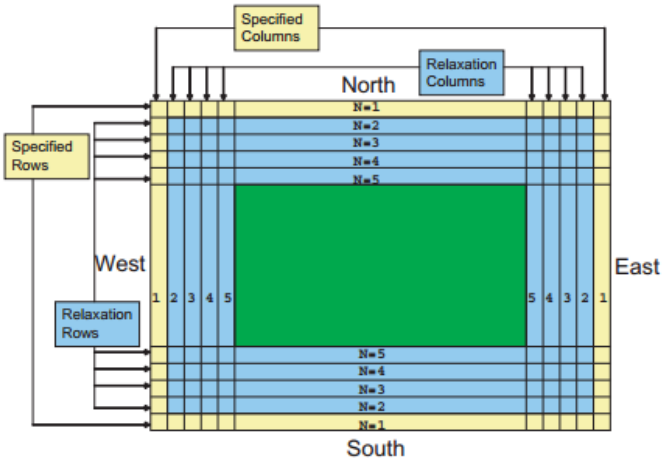

Figure 8. Representation of a five-point transition or buffer zone (blue) between the lateral boundaries of the limited-area model domain (yellow; generally interpolated from coarser model data to the spacing of the finer limited-area model domain) and the interior of the limited-area domain (green). Image obtained from http://www2.mmm.ucar.edu/wrf/users/docs/arw_v3.pdf.

Because of the use of a transition zone near the limited-area model domain's lateral boundaries, the lateral boundaries are typically placed far away from the feature(s) and/or region(s) of most interest within the limited-area model domain.

Upper and Lower Boundaries

For both limited-area and global models, there exist both upper and lower boundaries. The upper boundary is typically specified by adding a layer above the topmost model level in which realistic profiles of model variables from the topmost model level to the top of the atmosphere are specified via linear interpolation. This works reasonably well when the topmost model level is at relatively low pressure (e.g., 2-20 hPa) and less well when it is at relatively high pressure (e.g., 50-100 hPa, where atmospheric conditions vary substantially from those at the top of the atmosphere. See also Cavallo et al. (2011, *Mon. Wea. Rev.*) for more details. There also typically exists a damping layer over the uppermost few kilometers of the model domain to dampen vertically-propagating waves so that they and their energy do not escape the model domain.

The bottom boundary is more challenging and, given that it is where we all live, arguably more important to accurately represent on meteorological time scales. At a bare minimum, accurate topography and land use information (e.g., land versus water, grassland versus forecast, urban versus rural, etc.) must be provided. Heat and moisture transport between the surface and model atmosphere must be accurately handled, as must the effects of surface friction. There exist other considerations for both land and water surfaces. Typically, modern NWP models are **coupled** to land-surface models to effectively represent the bottom boundary over land, although these models themselves are also imperfect. There also exist surface-layer parameterizations to represent the effects of what occurs at the lowermost level – e.g. centimeters – above the surface itself. Some

models are also coupled to ocean models, which can represent how atmospheric forcings modify the oceanic state (and thus influence the atmospheric forecast). This is most important when air-sea interaction is important, such as for tropical cyclones, although most models assume a fixed or climatological oceanic state evolution. We will discuss land-surface modeling later in the semester.

Initial Conditions

A numerical model cannot run if it does not know the values of its variables (wind, temperature, moisture, etc.) at its initial time – its **initial conditions**. In general, initial conditions should:

- Be as accurate (close to observations, acknowledging that observations themselves have uncertainty and error) as possible. Despite the non-linearity of the atmosphere, forecast quality is directly proportional to initialization quality.
- Represent features at and coarser than the model's effective resolution. If they only represent coarser features, the so-called **cold start**, then finer-scale features must "spin up" over the first 6-12 h of the model forecast. Those initial conditions that do represent such finer-scale features are known as **warm start** (partially resolved) or **hot start** (fully resolved) initial conditions.
- Adhere to an appropriate balance condition. For vertical motions, especially for coarser models, this is typically hydrostatic balance, even if the model itself is non-hydrostatic. For horizontal motions, this could be geostrophic, gradient, or some more complex non-linear form of balance. Note that not all models will balance or insist on the existence of balanced initial conditions. However, it is a good idea that they be balanced so as to mitigate the generation of artificial inertia-gravity waves (to attempt to establish or restore balance) early in the model forecast.

While it is theoretically possible to apply some form of objective analysis routine to observations to obtain initial conditions, this is typically not done. Instead, **data assimilation** is used, wherein some "first guess" of the initial conditions is corrected via some means of assimilating available observations. The "first guess" is typically drawn from the 1-, 3-, 6-, or 12-h forecast from an earlier model run, though it can also be drawn from other sources such as climatology. Obtaining initial conditions in such a way is an example of **cycling**, as depicted in Fig. 9.

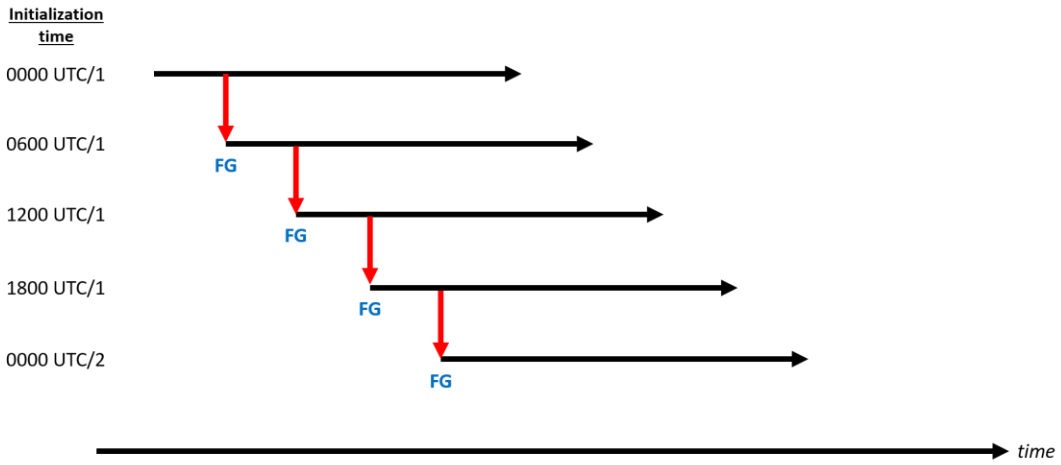

Figure 9. A schematic of forecast cycling, where in this case the 6-h forecast from the previous model run is used as the "first guess" (FG) for the next model run. Observations at the initialization time are assimilated to update and correct the "first guess" before the forecast is launched. This schematic represents a manifestation of three-dimensional data assimilation as temporal variability of the assimilated observations for a given model cycle is not considered.

Most limited-area models use initial conditions generated from another model's initialization. It is possible, however, for model users to initialize forecasts with their own cycled model + data assimilation system. Coupling a model like WRF-ARW to a data assimilation package such as the Data Assimilation Research Testbed (DART) toolkit is one common manifestation of this.

Despite each being founded upon the principle of minimizing the error in the initial conditions relative to observations, there exist multiple ways to assimilate observations. As you might expect, each of these methods have their own strengths and weaknesses. We will discuss these methods further toward the end of the semester.

Model Parameterizations

There exist certain scales that are **unresolved** by a numerical model, even though what happens on such scales may be quite important both societally and to model forecast quality. Such processes must be **parameterized** by the numerical model. To first order, parameterization involves the representation of a process in terms of its known relationships to dependent variables resolved on the model grid. An example of this is turbulent motions, which are a function of wind shear (mechanical turbulence) and buoyancy/stability (buoyant turbulence). Given model-resolved wind shear and stability, a parameterization can, by some means, infer the properties of and resolved-scale effects of unresolved, or **sub-grid-scale**, turbulent motions. Another example is clouds: given a profile of moisture and stability, a parameterization can infer the properties of and resolved-scale effects of unresolved clouds – or determine whether they should be present.

For any parameterized process, the parameterization is but an approximation and thus a major contributor to model error. Naturally, there are many ways that each parameterized process or set

of processes can be parameterized, each with its own assumptions, strengths, and weaknesses, not to mention level of complexity. For example, WRF-ARW version 4.1, released in August 2019, includes thirteen planetary boundary layer, twenty-nine cloud microphysics, fourteen cumulus convection, and eight shortwave and longwave radiation parameterizations!

There exist three primary reasons to employ a parameterization:

- **Scale**: The process or set of processes cannot be resolved on the model grid.
- **Complexity**: We can resolve the process or set of processes on the model grid but doing so would be computationally expensive.
- **Knowledge**: We do not know enough about the relevant process or set of processes to be able to develop solvable equations to resolve the process or set of processes.

Of these, the first – scale – is by far the primary reason for parameterization within modern models. While the last – knowledge – is relevant, it generally is encapsulated within and superseded by the first. Not many processes fall within the second – complexity – and, even if they did, the improved forecast quality that typically results from explicit rather than parameterized treatment of such processes typically far outweighs the added computational cost.

The most common processes that are parameterized by modern NWP models include the planetary boundary layer, surface layer, shallow and deep cumulus convection (the latter particularly for coarser models), cloud microphysics, and radiation. Different land-surface and ocean models may also be viewed as parameterizations. There also exist other types of parameterizations: stochastic parameterizations applied to initial and lateral boundary conditions or physical parameterization tendencies, orographic drag parameterizations, and so on. We will cover the underlying physics and parameterization methods for physical parameterizations in detail later in the semester.