AOS 452 – Lab 13 Handout
Upper-Level Frontogenesis and Sawyer-Eliassen Circulations

Introduction

As we discussed in class, fronts are locations at which we cannot ignore the effects of ageostrophy. Furthermore, we've discovered that an ageostrophic secondary circulation must accompany a change in the magnitude of the horizontal temperature gradient. It turns out that these ageostrophic secondary circulations can be diagnosed directly via the **Sawyer-Eliassen Circulation Equation**. This equation emerges from a consideration of across-front ageostrophic motions and is applicable to a wide variety of atmospheric phenomena, including jets and frontal boundaries. This lab will explore the process of **upper-level frontogenesis** as analyzed from the Sawyer-Eliassen perspective.

Sawyer-Eliassen Circulation Equation

A particularly useful way to interrogate the vertical circulations associated with jet-front structures is afforded by employing the Sawyer-Eliassen Circulation Equation (Sawyer 1956, Eliassen 1962):

$$(-\gamma \frac{\partial \theta}{\partial p}) \frac{\partial^2 \psi}{\partial y^2} + (2 \frac{\partial M}{\partial p}) \frac{\partial^2 \psi}{\partial p \partial y} + (-\frac{\partial M}{\partial y}) \frac{\partial^2 \psi}{\partial p^2} = Q_g - \gamma \frac{\partial}{\partial y} (\frac{d\theta}{dt})$$

where M is the absolute geostrophic momentum (M=U_g-fy), ψ is the Sawyer-Eliassen streamfunction, Q_g is the geostrophic forcing function, and γ is a thermodynamic constant. The coordinate system is defined such that the x-axis points in the along-front direction and the y-axis points in the across-front direction into the colder air. The final term on the right-hand side of the equation is the diabatic forcing term and will not be included in our discussion for this lab. Such a term is important, however, when considering the Sawyer-Eliassen circulation in the vicinity of heavy precipitation. The geostrophic forcing function is the sum of both the shearing deformation term ($Q_{SH} = 2\gamma[(\partial U_g / \partial y)(\partial \theta / \partial x)]$) and the stretching deformation term ($Q_{ST} = 2\gamma[(\partial V_g / \partial y)(\partial \theta / \partial y)]$) and the winds associated with the Sawyer-Eliassen circulation can be determined directly from the streamfunction, ψ, such that $v_{ag} = -\partial \psi / \partial p$ and $\omega = dp / dt = \partial \psi / \partial y$ within a cross-sectional plane perpendicular to the frontal boundary.

Alright, now to make sense of all that ugly math! Ultimately, the way the above equation works is similar to how we've looked at any Laplacian operator, such as in the QG-omega equation. For example, if the forcing function is positive, then the left hand side must be positive as well. So, if the sum of these second derivative terms is positive, then ψ must be a minimum! The opposite holds true if the forcing function is negative. Once we've identified the streamfunction field, we can easily determine the winds associated with that field using the expressions for v$_{ag}$ and ω listed above. As we explored in class, positive

(negative) values for the forcing correspond to frontogenesis (frontolysis) and thermally direct (indirect) secondary circulations.

Lastly, if you take a close look at the left-hand side of the equation, you'll notice three terms in parentheses. These coefficients are constants for each point on the globe at a given time and represent the static stability, baroclinicity, and a portion of the absolute vorticity (if you substitute for the expression for M), respectively. Interestingly, these are all three of the characteristics of frontal boundaries that we discussed earlier in the semester!

Successive Over-Relaxation

Now, in order to find the Sawyer-Eliassen streamfunction, one must solve the above equation for ψ. At first glance, the equation looks pretty formidable to solve by hand! Luckily, with the help of computers and centered finite differencing techniques, we are able to solve the equation rather easily and painlessly if we are given a gridded dataset of model data.

If you think all the way back to the first weeks of your dynamics classes, you learned that you could approximate the value of a function by performing a Taylor series expansion (Don't worry, I won't make you do it!). After some algebra and mental strife, you can obtain the following expressions for the first and second derivatives of a function (see Martin pg. 10-11):

$$f'(x_0) \approx \frac{f(x_0 + \Delta x) - f(x_0 - \Delta x)}{2\Delta x}$$

$$f''(x_0) \approx \frac{f(x_0 + \Delta x) - 2f(x_0) + f(x_0 - \Delta x)}{(\Delta x)^2}$$

where x_0 is the point at which the derivative is being taken and Δx is the distance between x_0 and the adjacent points.

It is then possible to plug in these expressions for each of the derivatives in the Sawyer-Eliassen Circulation Equation. This allows us to calculate a value for the forcing terms at each point in our gridded dataset, as well as each of the coefficient terms in the parentheses on the left-hand side of the equation. We can also plug in the approximation for a second derivative for each of the terms with ψ in them.

It is at this point we run into a problem. How can we figure out the value of these second derivatives if we don't know what psi is!? This is where we use a technique called Successive Over-Relaxation.

The first step is to take the form of the Sawyer-Eliassen Circulation Equation with all the derivative approximations substituted. Then, perform some algebra to rearrange the equation into an expression for ψ at x_0 (e.g. f(x_0)) in terms of everything else. This means there will be some terms on the right-hand side that include the value of ψ at other points other than x_0!

Once we have this expression, we can apply the successive over-relaxation technique. Successive over-relaxation is an iterative technique (e.g. it runs over-and-over until a solution is obtained) and, as part of the solution process, gives calculated values an extra nudge towards the final solution during each iteration. This is done so that the solution process is completed in a more rapid fashion (over-relaxation).

The way this works is as follows:
      1) A value of 0 for ψ is initially assigned to each value in the domain for which you want to solve the equation. On the boundaries of this domain, the values will remain 0 throughout the entire process. This is our **boundary condition** that permits a solution.
      2) Then, you progress through each point one at a time on the interior of the domain and solve your expression for ψ at x_0 for each point. Once the value is calculated, you use the following expression to "push" your value for ψ closer to a final solution:

$$\psi = \psi_{old} + \alpha * (\psi_{new} - \psi_{old})$$

where the updated ψ becomes the sum of the old ψ value and the product of an over-relaxation coefficient (α) with the difference between the updated ψ value and the old ψ value. α is typically between 1.0 and 2.0, accelerating the process of convergence to a solution.
      3) A solution is typically obtained once the difference between the calculated ψ value and the old ψ value is less than a specified amount (usually 1% of the magnitude of ψ) at all points in the domain.

The Sawyer-Eliassen Equation Solver

Now that we've been able to obtain a way to solve the Sawyer-Eliassen Circulation Equation using iterative techniques, it can be employed in numerous situations to diagnose the strength and sense of ageostrophic secondary circulations. For this lab, Andrew Winters (former 452 TA and all-around genius) created a program that solves the Sawyer-Eliassen Circulation Equation within any cross section that you specify. There are two scripts you will need to download from Learn@UW under the Lab13 Module:

      SEextract452.csh
      SEsolver452.py

OR

Copy them from my directory using the following syntax:

   cp /ef5/raid6/class/fall11/mbreeden/Desktop/SEextract452.csh   .
    cp /ef5/raid6/class/fall11/mbreeden/Desktop/SEsolver452.py   .

(the period at the end will copy the files to your present working directory)

Here is the process that you'll need to follow to solve for the circulations:

       1) Once you have taken a cross section, exit gdplot and open up gdcross (do not type gpend). (Hint: When you are taking your cross section, make sure it is perpendicular as possible to the thickness contours and covers a large latitudinal/longitudinal distance.)

       2) You'll see a parameter called CXSTNS within your parameter list with the bounds of the cross section you've selected.

       3) Use these values as an input for the GAREA in the SEextract452.csh script and **ROUND EACH VALUE TO THE NEAREST WHOLE NUMBER**. The format you will use is as follows:

       lowest latitude; westernmost longitude; highest latitude; easternmost longitude
            ex: 16;-100;45;-70

       4) Change the GDATTIM and GDFILE to the correct inputs.

       5) Run the SEextract452.csh script after making it executable (this will create text files that you will use to solve the Sawyer-Eliassen equation in the next script).

       6) Open up the SEsolver452.py script in gedit

       7) Scroll down slightly to the first stop sign, as indicated in the script, and enter in the same latitude and longitude values used in SEextract452.csh where specified.

       8) Indicate the "tilt" of the cross section you've drawn. If the cross section is oriented from NW to SE, type "westward". If it's oriented from NE to SW type "eastward".

       9) Scroll down to the next stop sign that controls the forcing function. Select which forcing terms to include in the solution by commenting or uncommenting the correct lines.

       10) Scroll all the way to the bottom to select which variables you want to plot in your solution by commenting and uncommenting lines within that section of the script.

       11) Save everything and run the SEsolver452.py script after making it executable.

       12) You should see a plot appear after the script is run. You can save and print this off if you so choose.

There you go! In about 5 minutes you will have solved that nasty looking equation without even breaking a sweat!

One last tip to remember when looking at your plot of streamfunction: The winds associated with the circulation flow with **higher** streamfunction values to their **left** in this particular plot, per the relationship discussed earlier.

Theoretical Sawyer-Eliassen Circulations

As we've discussed in class, the structure of ageostrophic circulations in a straight jet streak with no geostrophic temperature advection resembles the four-quadrant model with a thermally direct circulation in the jet entrance region and a thermally indirect circulation in the jet exit region (Panel A).

The introduction of along-jet geostrophic temperature advection, however, acts to shift the location of these circulations. The figure below shows that, within a regime of geostrophic cold air advection, the thermally direct (indirect) circulation is shifted towards the equatorward (poleward) side of the jet such that descent is present along the length of the

jet axis (Panel B). This is extremely favorable for the production of upper-tropospheric fronts via tilting frontogenesis.

Within a regime of geostrophic warm air advection, the thermally direct (indirect) circulation is shifted towards the poleward (equatorward) side of the jet, such that ascent is present throughout the length of the jet axis (Panel C).

These theoretical circulations, which result as the sum of both the shearing and stretching deformation forcing terms, will be helpful to remember when analyzing the resultant circulations produced from the Sawyer-Eliassen solver.

Upper Troposphere