Koninklijk Nederlands
Meteorologisch Instituut
Ministerie van Infrastructuur en Milieu

The dynamic role of the cross-frontal circulation

Marten Blaauw

De Bilt, 2011 | Stageverslag

Koninklijk Nederlands
Meteorologisch Instituut
Ministerie van Infrastructuur en Milieu

The dynamic role of the cross-frontal circulation

Versie 1.0

Datum 20 oktober 2011
Status Definitief

Abstract

Atmospheric fronts form and intensify continuously within extra-tropical cyclones, a process named frontogenesis. The distribution of (anti)-cyclones provides the two main ingredients for frontogenesis: two different airmasses and a wind deformation field that brings them together. In response, a cross-frontal circulation (CFC) develops.

The main goal of this study is to examine the dynamic role of the CFC in the upper-troposphere. The CFC is responsible for initiating tropopause folds. Such events result in the intrusion of stratospheric air into the troposphere, a process that is important for the forecast of severe weather as well as for ozone distribution studies.

Previous studies on CFCs are based on simple balanced models. In Hoskins [1982], an overview is given of the use of quasi-geostrophic as well as semi-geostrophic theory in modeling deformation frontogenesis. A common approach is the use of a geostrophic coordinate transformation in order to get around the nonlinearity of the problem.

We take a different approach here. We examine tropopause folding using Cartesian coordinates. We will use numerical techniques that are capable of solving the nonlinear semi-geostrophic set of equations in a PV framework. We restrict ourselves to two-dimensional frontogenesis in order to explain the basic characteristics of a tropopause fold evolution.

Using this approach, we discuss the differences between quasi-geostrophic and semi-geostrophic theory in describing a tropopause fold evolution. Furthermore, we discuss the contribution of deformation frontogenesis in the vertical development of a tropopause fold. A case study is included to compare our model results with ECMWF operational data archive.

Contents

1 Introduction

Fronts are transition areas: transitions between warm and cold air masses, dry and humid air, sunny and cloudy skies, calm and windy conditions. At the Earth's surface, such transitions may be associated with severe weather phenomena like large hail, lightning activity and severe wind gusts. In the upper troposphere, frontal zones are also formed, especially at the tropopause. The focus on frontogenetic processes shifted increasingly towards the tropopause region since upper air observations became available in the mid-fifties of the previous century.

Frontal zones near the surface and the tropopause are embedded in larger synoptic scale systems of low and high pressure areas. Their spatial distribution create favorable conditions for front formation. In return, fronts disturb the atmosphere not only locally but also on synoptic scales. These nonlinear relations between fronts and their environments make it difficult for both forecaster and researcher, who would like to understand physical processes in terms of 'cause and effect'.

The cause and effect of fronts can be studied by (i) the use of operational numerical weather prediction models and (ii) the use of off-line theoretical atmospheric models. The forecaster will mostly rely on (i) and use his experience to predict (severe) weather associated with fronts. But much can be learned too from the second approach where under ideally created conditions one can understand cause and effect in terms of direct mathematical relations. This is the approach that we shall take here.

We aim to contribute to understanding cause and effect in fronts by zooming in on one aspect of a frontal system, namely the cross-frontal circulation (hereafter CFC) and its dynamical role. The CFC forms simultaneously with an intensifying front and affects the evolution of the frontal system in return. Using a theoretical model, we shall simulate the formation of the front and the CFC. The nonlinear aspect of this problem shall be examined in the modern framework of potential vorticity (hereafter PV). Its advantages are extensively described in Hoskins et al. [1985] in which paper it is emphasized that the PV framework can be used when:

(I) the atmosphere is assumed to be in a condition of *balance* (well-chosen w.r.t. time and spatial scales) so that the corresponding balanced equations can be used;

(II) a reference state is specified that describes the background temperature distribution θ;

(III) the PV inversion problem is solved on the whole specified domain with proper attention to the boundary conditions.

We will study the dynamical role of the CFC and its effect on the tropopause structure using a two-dimensional stretching deformation model (i) without the use of geostrophic coordinates (ii) with the use of numerical relaxation methods for *non*linear PV inversion problems.

Our results are based on results of others in the past. We first start with a historic overview on the study of fronts. Piece by piece a conceptual model of a frontal system will arise from which we will formulate our main research questions at the end of the introduction.

1.1 Historic overview

The first important achievements in frontal studies were made by the Bergen school under V. Bjerknes after World War I. In their most important papers, Bjerknes [1919] and Bjerknes and Solberg [1922] describe the Norwegian model for cyclogenesis where a cyclone forms *as a result* of instability of a polar front, the border between polar and tropical air masses. In figure 1, taken from an excellent overview of the history on frontogenesis given by Reed [1990], the life cycle of cyclones according to the Norwegian model is shown. It provided an explanation of the formation of the warm and cold front, how they occlude and how eventually the cyclone cuts off from the polar front. This model was based on very limited data.

Not much was known about the upper atmosphere in those days. But in the 1940s new observational techniques, like radiosonde networks, began to reveal the three-dimensional structure of the atmosphere. The discovery of the jet stream and the studies on Rossby waves drew the attention to the upper atmosphere and its connection with cyclogenesis.

The theory of baroclinic instability by Charney and Eady (Charney [1947], Eady [1949]), introduced end 1940s, was another milestone and it became widely accepted as the explanation for cyclogenesis. The baroclinic instability theory predicted normal modes of growing baroclinic waves which in terms of growth rates and characteristic wavelengths ($\approx$ 1000km) could explain the process of cyclogenesis well.

In this dynamical view, fronts are formed in growing baroclinic wave structures and are the *result* of cyclogenesis and not the cause. It was shown however that fronts have a large influence on cyclogenesis in return. Moreover, fronts can aid the formation of new 'satellite'-cyclones (Eliassen [1966]).

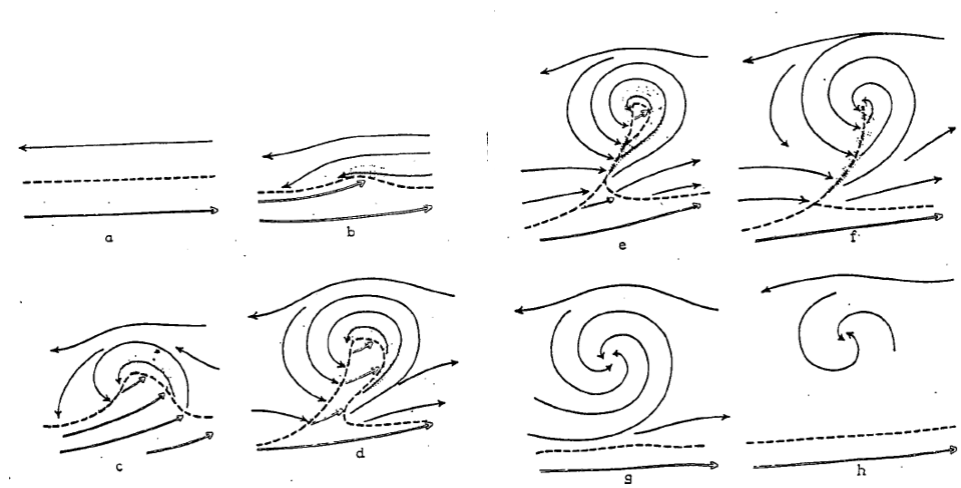


Figure 1 – The life cycle of cyclones (Bjerknes and Solberg [1922])

1.2 Conceptual view of front dynamics

The kinematic studies by Bergeron [1928] and Miller [1948] provided the basis for developments in frontogenesis studies later on. They examined the velocity and temperature fields associated with frontogenesis. The related shear and confluence environments, provided by cyclogenetic processes on larger scales, are still used in current simple two-dimensional numerical studies of frontogenesis.

But how do we actually define a front? Practically, it is difficult for operational meteorologists to define fronts which come in different flavors in terms of activity and precipitation. Moreover, fronts form on a variety of time and spatial scales. In order to relate it to cyclogenesis and distinguish from local sea-breeze fronts or convective outflow boundaries, we follow Hoskins [1982] in his definition of a front (slightly paraphrased):
A front is considered to be a region whose length scale is comparable with the radius of deformation in the along-front direction but much less in the cross-front direction, with in the cross-front direction significant changes in buoyancy and velocity with gradients tending to become very large in a finite time.

The dynamical picture was extended when one started to realize that there had to be a response to the frontogenetic forcing in the form of a cross-frontal circulation (hereafter CFC). Based on previous physical ideas, Sawyer derived an equation for this circulation and Eliassen modified it later on such that it became known as the Sawyer-Eliassen equation (Sawyer [1956], Eliassen [1962]). This circulation completes our conceptual view on front dynamics (shown in figure 2 as green thermal-direct circles). In the years after, the role of the CFC was widely studied and its effect on atmospheric dynamics shall be our main topic here. The influence of the CFC is largest in two distinct areas, namely near the tropopause (1.2.1) and at the Earth's surface (1.2.2).

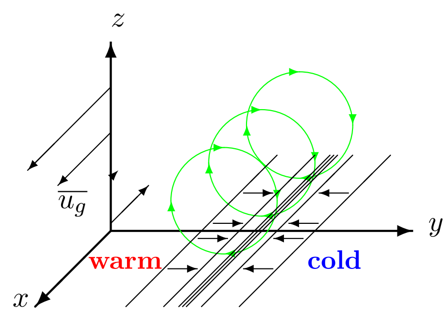


Figure 2 – Conceptual model of frontogenesis.

1.2.1 Tropopause folding

One of the first observations of tropopause folding was done through aircraft measurements in the 1960s. Nuclear weapon tests were common and radioactivity was sent into the stratosphere and remained there, serving as a tracer for stratospheric air. This is illustrated by figure 3, taken from Danielsen [1968]. The figure includes three flight levels through an upper air frontal zone where an increase in radioactivity of β radiation was measured.

Both extensions indicate intrusion of stratospheric air. It shows that besides radioactivity of β radiation (or another example: ozone), PV can also be used as a tracer of stratospheric air. These extensions of PV are recognized as tropopause folds and the formation and dissipation of these phenomena has been subject of many studies over the last few years, see Davies and Rossa [1998], Wandishin et al. [2000], Moore [1993] and Thorpe [1997]. The justification is the growing importance of understanding how ozone at mid latitudes is distributed and how its distribution changes dynamically. The role of tropopause folding in injecting ozone from the stratospheric reservoir is studied by Shapiro [1980], who estimates that 50% of the ozone stays in the troposphere once it is injected by tropopause folds.

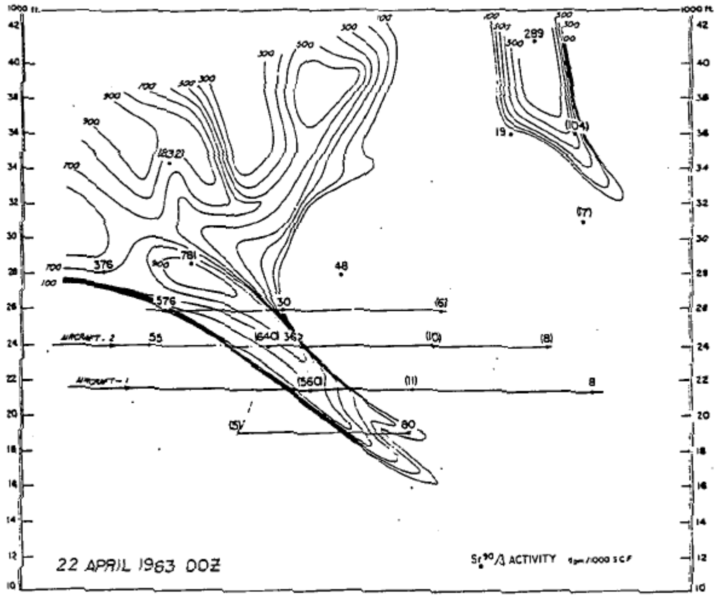
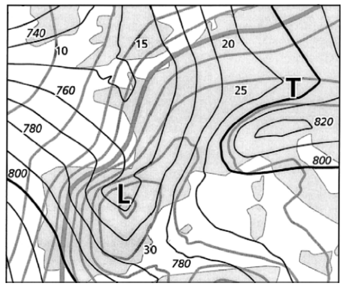


Figure 3 – Potential vorticity divided by the gravitational acceleration (each 100×10^{-9}m s K kg^{-1}) on the left side, radioactivity (β radiation) of strontium-90 (dpm/KSCF) in the upper right corner and three flight paths of the WB-50 aircraft. (Danielsen [1968])

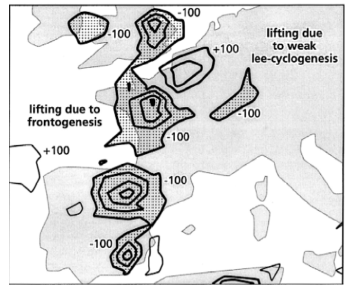
1.2.2 Surface boundary

From a forecaster's point of view, surface fronts are interesting because they come along with precipitation and other, possibly severe, weather phenomena. The vertical motion field, part of the CFC, plays an important role in triggering (deep moist) convection. One can find the rising branch ahead of cold fronts above lines of convergence, where the CFC and the environmental winds meet. Potential unstable air is forced to lift to the level of free convection where the energy of the potentially unstable air can be released. This results in (organized) thunderstorms with severe weather such as hail, excessive rainfall or damaging wind gusts. Developing thunderstorms may organize themself into squall lines along lines of convergence in the warm sector. The satellite images in figure 4 show an example of a typical severe thunderstorm setting above Europe on the 20 July 1992, where areas of forced rising air are located ahead of a strong cold front.

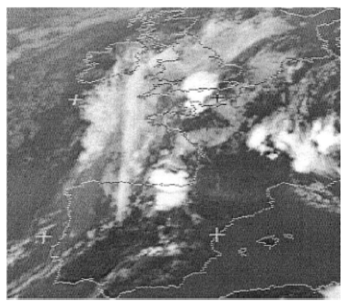
To understand when and where such severe weather phenomena develop, it is important to know more about how a front-CFC coupling evolves. Early numerical models did not include surface boundary effects and were simply based on a prescribed deformation field and the quasi-geostrophic equations (see [Hoskins, 1982, QG part]). It was successful in producing a sharp front near the surface in time, but did not have the other characteristics of a front,



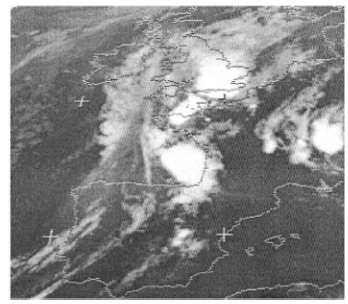
(a) 1800 UTC



(b) 1800 UTC



(c) 1730 UTC



(d) 1930 UTC

Figure 4 – A synoptic thunderstorm setting on 20 July 1992 around 1800 UTC showing (a) height (labeled in units of m) and temperature (labeled in °C) at 925 hPa with the cold front approximately located on the $20°$ isotherm, (b) Vertical velocity at 700 hPa (labeled in units of hPa per hour), (c) Meteosat satellite image at 1730 UTC and (d) at 1930 UTC (van Delden [2001])

like a slope and a realistic vorticity field. As a result, vertical motion fields and static stabilities were not realistic either.

With the introduction of the semi-geostrophic approach in models (Hoskins and Bretherton [1972]), one could produce a more realistic frontal slope, a stronger CFC resulting in a more realistic vertical velocity field. Its important role was realized in the papers by Hoskins and others and frontogenesis became a two-step process where the geostrophic deformation field concentrates the isotherms into a frontal zone and the induced CFC adds to this by intensifying and tilting the front towards the warm sector.

In reality, surface fronts are subject to a variety of secondary (surface boundary) effects like: (i) turbulent diffusion (Williams [1974]), (ii) surface friction Eliassen [1959] and (iii) latent heat release Hoskins and Bretherton [1972] unlike their upper atmospheric counterpart. For fronts in the upper atmosphere, these processes may intensify or weaken the front during frontogenesis, but are not of first-order importance.

1.3 Research questions

In this thesis, we focus on a numerical approach to study frontogenesis and the cross-frontal circulation using a PV framework. Our main goal is to understand the dynamic influence of the cross-frontal circulation on the structure of the tropopause, in particular the tropopause folding process.

An important aspect of our model is our choice of Cartesian coordinates. We wish to simulate a realistic tropopause folding process without the use of geostrophic coordinates. This requires us to build a two-dimensional model that solves a nonlinear problem in time. We will therefore also focus here on the development of the appropriate numerical techniques.

Both goals are the motivation behind the following research questions:

1. What does the semi-geostrophic approach add to the quasi-geostrophic approach in the representation of a cross-frontal circulation and a tropopause fold?

2. Can we simulate a realistic tropopause folding event with a two-dimensional semi-geostrophic model based on stretching deformation?

The first part of this thesis shall be concerned with background theory where the balanced equations and related theory shall be derived and explained. We start with discussing the *cause* of fronts in chapter 2. After that, the dynamical *effect* of fronts shall be explored, starting from general theory in chapter 3 and describing the balanced equations (i) in quasi-geostrophic form in chapter 4 and (ii) in semi-geostrophic form in chapter 5.

The second part shall discuss the numerical aspects starting with a general model overview in chapter 7, including the initial configurations of three model run results. Then follows a technical discussion on the model algorithm and the used numerical techniques in chapter 8. Then in the third part, we will discuss the results of the three model runs in chapter 9, followed by a description of a case-study and its results in chapter 10. After the summary & conclusions part, the reader can find in the Appendix the Fortran code (F) which can be used to reproduce the results of the given model runs.

Part I
Theory

2 Frontogenesis ingredients

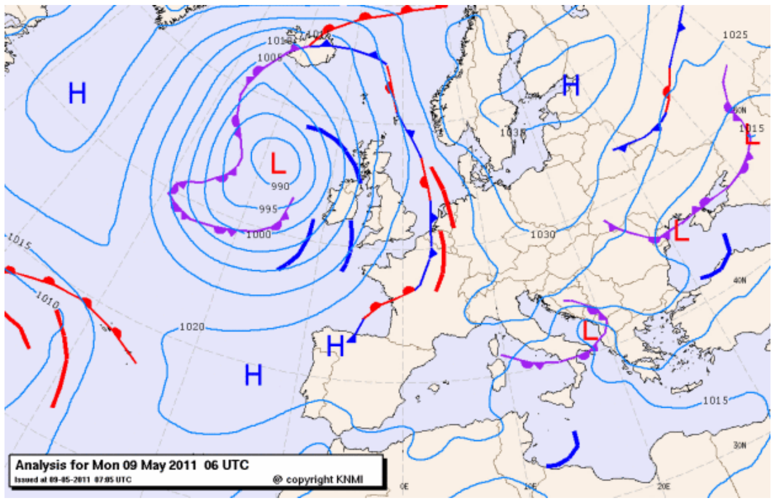


Figure 5 – Synoptic situation on the 9th of May 2011, showing a south-north quasi-stationary front over the low countries. A large low pressure area is located northwest of UK and a smaller one over the Adriatic Sea. High pressure is dominant over Scandinavia and the Iberian landmass.

In this section, we will examine the ingredients of frontogenesis more closely. A nice example of frontogenesis is shown in figure 5. The synoptic setting on this day is such that the flow pattern is almost symmetric with respect to the quasi-stationary front over Western Europe. In the western sector, cooler maritime air is transported in northeastern direction over NW-France and Belgium. On the other side of the front, warm continental air is transported in northwestern direction resulting in a so-called confluence wind field which favors frontogenesis. Such a synoptic setting (e.g. two-dimensional confluence) would be ideal for a detailed two-dimensional study of frontal dynamics.

2.1 Initial temperature gradient

Horizontal gradients in air temperature can be initiated by physical processes on many spatial scales. Some examples on how such gradients form shall be given here.

The most obvious example is differential heating meaning that air on one location warms up quicker with respect to a nearby location. Such gradients in temperature could form over areas where there are differences in heat capacity (e.g. land and sea), incoming radiation on global scale (poles and equator), cloudiness, latent heat release or in radiative cooling during night. In this paper, we will use the gradient in *potential* temperature to include the adiabatic warming and cooling effect for vertical displaced air.

Gradients in *equivalent* potential temperature θ_e can be found near dry lines, which are fronts with a large gradient in relative humidity or in dew point temperature (dew point lines are also a common name for such fronts). Dry lines are typically found in America where on the Great plains warm humid air (dense) from Mexico meets warm dry air (light) from the Golf of Mexico forming a gradient in air density, i.e. the dry line. This gradient can reverse during the night.

Differences in humidity of two air masses on both sides of a front can have an enhancing or weakening effect on frontogenesis. If humidity plays a significant role then θ may be replaced by its equivalent part θ_e where it is possible that $\delta\theta_e > \delta\theta$ in case of enhancement. For example, in case of a typical summer day synoptic setting over Europe where warm humid air is driven out by dry and cooler air behind an east moving cold front. However, the humidity does not play a crucial role in the theory of frontogenesis. It's influence on frontogenetic processes is described by others in Markowski and Richardson [2010, Ch. 5.3] and shall not be considered here.

2.2 Wind deformation field

Once a temperature gradient is present, a front can intensify when the wind field is favorable to increase an initial temperature gradient, i.e. when it advects two different air masses towards each other. We will study several *deformation* fields in this section.

In the following part, we will start from the basics and investigate the effect of an arbitrary local wind field on a line segment, which could be visualized as an isotherm along a front. This line will be deformed by the wind field indicated by the wind vectors in figure 6. This approach is based on the detailed notes by Smith [2007] and references herein.

Figure 6 shows an arbitrary velocity field in which the deformation of an arbitrary line segment PQ ($\vec{\delta x}$) will be considered in two-dimensional space (x, y).

Let P be at (x, y) and its velocity given by (u_0, v_0). Let Q be at $(x + \delta x, y + \delta y)$ and its velocity given by $(u_0 + \delta u, v_0 + \delta v)$. Now we consider the relative motion between P and Q to see if the line will be stretched, contracted, rotated, etc.

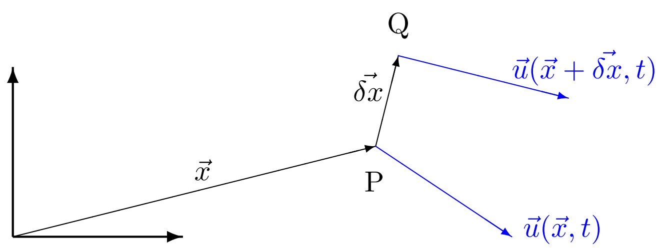


Figure 6 – Deformation of a line segment PQ by a velocity field $u(\vec{x},t)$.

Note that we can easily expand figure 6 to a 3D space and investigate how the line segment PQ is deformed in the vertical too. But for the purpose of the 2D model that is developed, it is enough to consider horizontal deformation at a certain height.

The terms $(\delta u, \delta v)$ can be expanded as:

$$\delta u = \frac{\partial u}{\partial x}\delta x + \frac{\partial u}{\partial y}\delta y \tag{1a}$$

$$\delta v = \frac{\partial v}{\partial x}\delta x + \frac{\partial v}{\partial y}\delta y \tag{1b}$$

Which can be put in matrix form:

$$\begin{pmatrix}\delta u & \delta v\end{pmatrix} = \underbrace{\begin{bmatrix}\frac{\partial u}{\partial x} & \frac{\partial u}{\partial y}\\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y}\end{bmatrix}}_{M}\begin{pmatrix}\delta x\\ \delta y\end{pmatrix} \tag{2}$$

Any matrix can be the sum of a symmetric matrix S and an antisymmetric matrix T. Such a split is common in fluid dynamics and will be done here as follows:

$$M = \begin{bmatrix}\frac{\partial u}{\partial x} & \frac{\partial u}{\partial y}\\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y}\end{bmatrix} = S + T \tag{3}$$

In the following procedure a shorter notation will be used for the partial derivatives. Both S and T now can be written as:

$$S = \begin{bmatrix}\frac{1}{2}(u_x + u_x) & \frac{1}{2}(u_y + v_x)\\ \frac{1}{2}(v_x + u_y) & \frac{1}{2}(v_y + v_y)\end{bmatrix} \tag{4a}$$

$$T = \begin{bmatrix}\frac{1}{2}(u_x - u_x) & \frac{1}{2}(u_y - v_x)\\ \frac{1}{2}(v_x - u_y) & \frac{1}{2}(v_y - v_y)\end{bmatrix} \tag{4b}$$

The terms in the matrix reflect the effects of: (i) divergence D, (ii) Stretching deformation A, (iii) shearing deformation F and (iv) vorticity ζ:

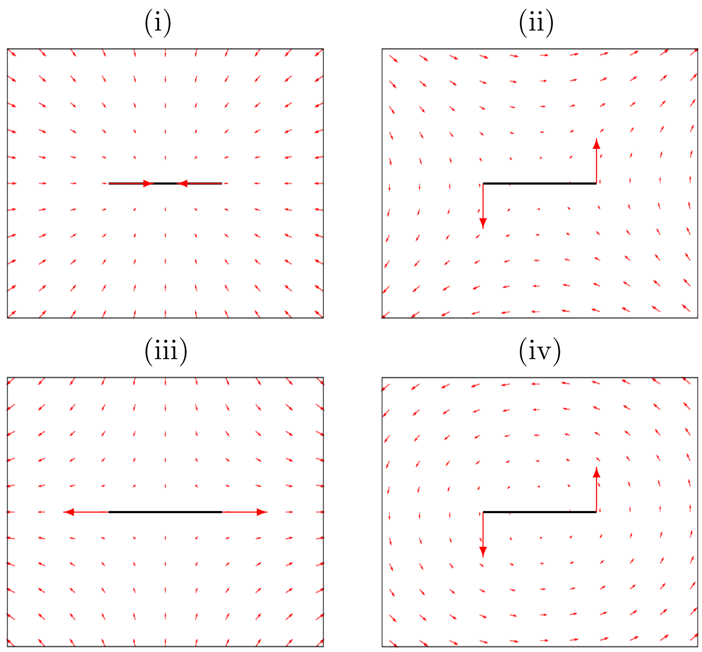


Figure 7 – Decomposition of an arbitrary velocity field into four categories: (i) convergence , (ii) stretching deformation, (iii) shearing deformation and (iv) rotation.

(i) $D = u_x + v_y$

(ii) $A = u_x - v_y$

(iii) $F = v_x + u_y$

(iv) $\zeta = v_x - u_y$

The velocity fields are illustrated in figure 7.

The above definitions can be rewritten in terms of (u_x, u_y, v_x, v_y) as:

$$u_x = \frac{1}{2}(D+A), \quad u_y = \frac{1}{2}(F-\zeta), \quad v_x = \frac{1}{2}(F+\zeta), \quad v_y = \frac{1}{2}(D-A), \tag{5}$$

such that (3) changes to:

$$M = S + T = \begin{bmatrix} \frac{1}{2}(D+A) & \frac{1}{2}F \\ \frac{1}{2}F & \frac{1}{2}(D-A) \end{bmatrix} + \begin{bmatrix} 0 & -\frac{1}{2}\zeta \\ \frac{1}{2}\zeta & 0 \end{bmatrix} \tag{6a}$$

$$M = \frac{1}{2}\left[\begin{pmatrix} D & 0 \\ 0 & D \end{pmatrix} + \begin{pmatrix} A & 0 \\ 0 & -A \end{pmatrix} + \begin{pmatrix} 0 & F \\ F & 0 \end{pmatrix} + \begin{pmatrix} 0 & -\zeta \\ \zeta & 0 \end{pmatrix}\right] \tag{6b}$$

which we can write in component form (using (1), $u = u_0 + \delta u$ and $v = v_0 + \delta v$):

$$u = u_0 + \frac{1}{2}(D\delta x + A\delta x + F\delta y - \zeta\delta y) \tag{7a}$$

$$v = v_0 + \frac{1}{2}(D\delta y - A\delta y + F\delta x + \zeta\delta x) \tag{7b}$$

This is the desired result, as it contains much information about the local velocity field near P and Q. An air column of length PQ is deformed in the following way by the four deformation field components:

(i) *pure convergence* $\vec{u} = \frac{1}{2}D(x, y) = \frac{1}{2}D(r\cos\theta, r\sin\theta) = \frac{1}{2}D\vec{r}$ where in the last step the definition of the unit vector $\hat{r} = (\cos\theta, \sin\theta)$ has been used here and that $\vec{r} = r\hat{r}$.
This corresponds to pure radial motion along PQ no matter how it is orientated. Compression occurs along both axes in case of convergence.

(ii) *pure stretching deformation* $\vec{u} = \frac{1}{2}A(x, -y)$. It is very similar to divergence accept for a minus sign such that compression along one axis goes together with stretching along another axis, hence stretching deformation.

(iii) *pure shearing deformation* $\vec{u} = \frac{1}{2}F(y, x)$. Streamlines are given by $y^2 - x^2 = const.$ which is similar to the stretching case only with a rotation of 45° w.r.t. dilatation axis.

(iv) *pure rotation* $\vec{u} = \frac{1}{2}\zeta(-y, u) = \frac{1}{2}\zeta(-r\sin\theta, r\cos\theta) = \frac{1}{2}\zeta r\hat{\theta}$ which corresponds to a solid body rotation with angular velocity $\frac{1}{2}\zeta$.

In frontogenetic studies, (ii) and (iii) are most important and are true frontogenetic deformation fields. Convergence (i) induces additional vertical motion apart from the CFC (e.g. Ekman pumping by convergence as a consequence of surface friction). Pure rotation (iv) corresponds to rotation of isotherms rather than bringing them together.

In the appendix A, we take a small sidestep to show how A and F can be combined to one deformation field and how one could interpret this physically.

2.3 Kinematics of frontogenesis

Now that the velocity field has been separated into four characteristic deformation fields, we consider the evolution of an initial temperature gradient by deformation. In this kinematic view approach, we do not consider forces or balances yet. Our purpose here is to show under what conditions frontogenesis or frontolysis occurs.

The frontogenesis function is the right parameter for this purpose and is defined as:

$$\frac{D|\vec{\nabla}_h\theta|^2}{Dt} = 2\vec{\nabla}_h\theta \cdot \frac{D\vec{\nabla}_h\theta}{Dt} \tag{8}$$

where the second factor is defined as the frontogenetic forcing $\vec{Q}$:

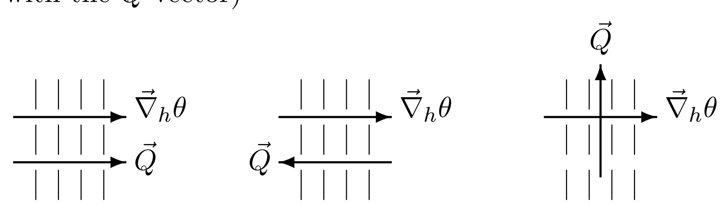
$$\vec{Q} \equiv \frac{D\vec{\nabla}_h\theta}{Dt} \tag{9}$$

The vector $\vec{Q}$ is defined as the rate of change of a horizontal temperature gradient from a Lagrangian point of view (i.e., moving with the flow). The interpretation of the frontogenetic function (8) is that:

(i) *frontogenesis* occurs when the $\vec{Q}$ vector points in the same direction as the temperature gradient $\vec{\nabla}_h\theta$ (from cold to warm)

(ii) *frontolysis* occurs when the $\vec{Q}$ is in the opposite direction of $\vec{\nabla}_h\theta$

(iii) When the $\vec{Q}$ vector is 90° out of phase with $\vec{\nabla}_h\theta$, neither frontogenesis nor frontolysis takes place but instead the front will rotate towards the $\vec{Q}$ vector (i.e., $\vec{\nabla}_h\theta$ will not change in magnitude but it will align itself with the $\vec{Q}$ vector)



(a) Frontogenesis (b) Frontolysis (c) Front rotation

The $\vec{Q}$ vector plays an important role in dynamical meteorology: it is not only helpful in recognizing intensification of fronts on weather maps, but it is also a great tool for indicating regions with uplifting motions near depressions. This has been shown by Hoskins et al. [1978] by deriving an alternative form of the ω-equation using the divergence of $\vec{Q}$, or $\vec{\nabla}_h \cdot \vec{Q}$, as an indicator for vertical motions. Thus, the frontogenetic forcing $\vec{Q}$ is not only helpful on the smaller mesoscales, but also on larger synoptic scales (e.g. depressions) where multiple warm and cold fronts with their opposing $\vec{Q}$ vectors can be used for great insight in the (vertical) dynamics.

A nice example is given in a frame of an animation (figure 8) where areas of downward resp. upward vertical motion correspond to divergence resp. convergence of $\vec{Q}$. Near the southern low pressure area the $\vec{Q}$ point along the isotherms indicating rotation of the isotherms by the velocity field. Further to the south (Mexico), the increasing cross-isotherm component of $\vec{Q}$ indicates an intensification of a cold front.

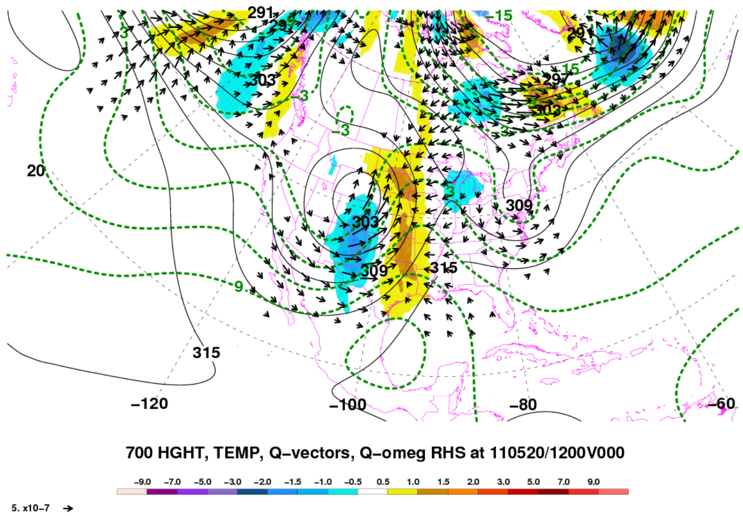


Figure 8 – Application of QG-ω equation to a synoptic scale case showing the 700 hPa geopotential height (solid black contours), 700 hPa temperature (dashed green contours), $\vec{Q}$-vectors (black arrows) and ω (shaded areas) from Galarneau Jr. [2011].

Let us have a closer look at $\vec{Q}$ in (9). The total time derivative can be expanded and rewritten in terms of $\frac{D\theta}{Dt} = \dot{q}$ which is the thermodynamic equation where $\dot{q} \equiv \frac{\theta}{c_p}\dot{Q}$ is the differential heating rate. Both components of $\vec{Q}$ can be rewritten as:

$$
\begin{aligned}
Q_1 = \frac{D}{Dt}\left(\frac{\partial \theta}{\partial x}\right) &= -\frac{\partial v}{\partial x}\frac{\partial \theta}{\partial y} - \frac{\partial u}{\partial x}\frac{\partial \theta}{\partial x} - \frac{\partial w}{\partial x}\frac{\partial \theta}{\partial z} + \frac{\partial}{\partial x}\left(\frac{D\theta}{Dt}\right) \\
&= -\frac{\partial v}{\partial x}\frac{\partial \theta}{\partial y} - \frac{\partial u}{\partial x}\frac{\partial \theta}{\partial x} - \frac{\partial w}{\partial x}\frac{\partial \theta}{\partial z} + \frac{\partial \dot{q}}{\partial x}
\end{aligned}
\tag{10a}
$$

$$
\begin{aligned}
Q_2 = \frac{D}{Dt}\left(\frac{\partial \theta}{\partial y}\right) &= -\frac{\partial u}{\partial y}\frac{\partial \theta}{\partial x} - \frac{\partial v}{\partial y}\frac{\partial \theta}{\partial y} - \frac{\partial w}{\partial y}\frac{\partial \theta}{\partial z} + \frac{\partial}{\partial y}\left(\frac{D\theta}{Dt}\right) \\
&= -\frac{\partial u}{\partial y}\frac{\partial \theta}{\partial x} - \frac{\partial v}{\partial y}\frac{\partial \theta}{\partial y} - \frac{\partial w}{\partial y}\frac{\partial \theta}{\partial z} + \frac{\partial \dot{q}}{\partial y}
\end{aligned}
\tag{10b}
$$

Now consider the terms in (10b), in accordance with the chosen orientation of our front. The first two terms represent (i) wind shear of the along-front wind which is a combination of shearing deformation F and rotation ζ (see (5) and fig. 9(b)) and (ii) confluence of the cross-front wind which is a combination of stretching deformation A and convergence D (see (5) and fig. 9(a)). Both horizontal mechanisms are most important in *surface* frontogenesis.

(a) Confluence of cross-frontal wind (b) Wind shear of along-front wind

The third term describes the tilting effect by differential vertical advection which transfers the vertical temperature gradient into a horizontal one. Its effect is dominant in regions where $\frac{\partial w}{\partial y}$ is sufficiently large, i.e. in the mid and upper troposphere.

Finally, the fourth mechanism represents differential heating which, for example, is responsible for the pole-to-equator temperature gradient on larger scales.

The two horizontal deformation fields are a combination of the velocity fields discussed earlier in 2.2 (in terms of D, A, F, ζ) and the thermal gradient. But these four parameters are not explicitly included, which makes a direct physical interpretation more difficult. An alternative form of the frontogenetic function (8) in terms of D, A, F, ζ is given by (Markowski and Richardson [2010]):

$$\frac{D|\vec{\nabla}_h\theta|^2}{Dt} = |\vec{\nabla}_h\theta|^2(A'\cos 2\phi - D) \tag{11}$$

where A' is total deformation parameter defined as $A'^2 = A^2 + F^2$ (see (125a)), ϕ is the angle between the isotherms and the axis of dilatation (AOD) given by x' in figure A along front where maximum *stretching* occurs and finally convergence given by $D < 0$. The derivation of (11) is given in appendix B.

For the following angles, $\phi = (n\pi)^\circ, \quad n = 0, 1, ...$ frontogenetic forcing $\vec{Q}$ is maximal because it corresponds to situations where maximum stretching (along AOD) occurs along the isotherms and contraction across the isotherms. Although rotation ζ is not included in (11), one can imagine that rotation of isotherms (near the center of a depression for example) can indirectly lead to frontogenesis by reducing ϕ towards zero in time, such that stretching and shearing deformation becomes more effective!

In this thesis, we assume that these four local forcing mechanisms result from large scale balances associated with the distribution of (anti)-cyclones on synoptic scales. On such scales, the effects of *geostrophic* tilting and diabatic heating are neglected. Many authors (Hoskins [1982], Moore [1993], Sawyer [1956], etc.) only assume a confluence pattern in their study to frontogenesis such that it reduces to a two-dimensional problem. Hence, we shall only use the confluence term in our model, i.e. $Q_2 = -\frac{\partial v}{\partial y}\frac{\partial \theta}{\partial y}$ containing stretching deformation only!

2.4 Other front ingredients

Apart from an initial potential temperature gradient and a deformation field, there are many other ingredients on a variety of spatial and time scales that have a frontogenetic or frontolytic effect. These secondary effects were studied by Williams, Hoskins and others. A nice overview is given in Hoskins and Bretherton [1972]

An important ingredient that prevents the front from becoming too intense, is turbulent diffusion. Numerical experiments on frontogenesis including turbulent diffusion by Williams [1974] showed that fronts could form in a day or two and then remained in a quasi-steady state. This indicates a balance between the main frontogenetic processes and the frontolytic effect of turbulent diffusion.

Surface friction has two counteracting effects on frontogenesis. It has an overall dissipating effect on across-front velocities corresponding to a *smaller* frontal convergence. But on the other hand, if surface drag is included for the along-front velocities (which are larger), it will have an additional effect on destroying thermal wind balance and a stronger cross-frontal circulation will form to restore thermal wind balance. This was demonstrated already in the

1950s by Eliassen [1959]. The dynamical role of the cross-frontal circulation results in an *increased* convergence near in the surface boundary layer. This was also found by the models of Hoskins and Bretherton [1972] including Ekman layer suction to represent friction. Their models show convergence near the surface boundary layer and a compensating divergence effect above the surface boundary layer where the effect of friction is frontolytic.

The effects of latent heat release on cross-frontal circulations are studied by Hoskins and Bretherton [1972] and others. Latent heat release typically occurs in the warm sector where conditions are favorable for convection (source of lift). The release of latent heat warms up the atmosphere and therefore strengthens the circulation. Especially for intense cold fronts associated with severe thunderstorms developing along the convergence line in the warm sector, the contribution from latent heat release can become a substantial factor. But the effects of latent heat release can also be frontolytic if convection is mainly observed behind a cold front.

3 Front dynamics

In this section, we will start from a very basic form of the equations of motion (hereafter EOM). Our aim here is to derive a set of equations which is valid for describing the dynamics on spatial and time scales that are typical for fronts. Various approximations have thus to be made. To give a quick overview, we will start from a set of equations that has already been simplified by (i) Boussinesq approximation (hereafter BQ) and (ii) hydrostatic balance (hereafter HB) and will be simplified further using scale analysis. That will result in (iii) geostrophic wind balance (hereafter GWB) and the (iv) geostrophic momentum approximation (hereafter GM).

We start from the EOM (simplified by BQ and HB) also known as the 'primitive equations' (first numerical solutions in Williams [1967], Hoskins and Bretherton [1972]), which are given in Cartesian coordinates by:

$$\frac{Du}{Dt} - fv = -\frac{\partial \phi'}{\partial x} \quad (12a) \qquad \theta = \overline{\theta}(z) + \theta'(y,z,t)$$

$$\frac{Dv}{Dt} + fu = -\frac{\partial \phi'}{\partial y} \quad (12b) \qquad b = g\frac{\theta'}{\theta_0}$$

$$\frac{Db}{Dt} + N^2 w = 0 \quad (12c) \qquad \phi' = \frac{p'}{\rho_0}$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \quad (12d) \qquad N^2 = \frac{g}{\theta_0}\frac{\partial \overline{\theta}}{\partial z}$$

$$b = \frac{\partial \phi'}{\partial z} \quad (12e) \qquad \frac{D}{Dt} = \frac{\partial}{\partial t} + u\frac{\partial}{\partial x} + v\frac{\partial}{\partial y} + w\frac{\partial}{\partial z}$$

These equations are almost of similar form as those given in Hoskins [1975]. The equations in the left column represent: evolution of zonal and meridional momentum, the temperature equation, the continuity equation and HB. Their forms are the result of the Boussinesq approximation.

The Boussinesq approximation yields a splitting of variables into an atmospheric reference state and deviations from this state. In the right column, we see that the potential temperature (defined as $\theta = T(p_r/p)^\kappa$) is split into (I) a reference term, $\overline{\theta}$, and (II) a perturbation term, θ'.

We define the temperature perturbation in terms of buoyancy, which is the relative temperature with respect to the surface temperature, θ_0, expressed in units of g. Buoyancy is propertional to $\frac{\partial \phi'}{\partial z}$, where ϕ' is the pressure perturbation, p', divided by a constant density ρ_0.

The reference term only depends on height and has a constant temperature, θ_0, on the surface. The rate of increase of $\overline{\theta}$ with height determines the static stability of the reference atmosphere, given by N^2. The static stability is part of the temperature equation (12c).

Our temperature equation differs from Hoskins [1975, eq. 2] and needs some explanation. In Hoskins [1975] and Hoskins and Bretherton [1972], they assume an adiabatic reference atmosphere ($\overline{\theta} = \theta_0$) such that the vertical derivative is zero and (12c) reduces to a form $\frac{Db}{Dt} = \frac{D\theta'}{Dt} = 0$. $N^2 = 0$ may be interpreted as a lower limit in terms of static stability.

Another option is to take the temperature T constant which is an other limit in representing a realistic static stability. Such a reference atmosphere would be perfect to model the temperature profile of the stratosphere. The isothermal reference atmosphere has a constant static stability given by $N = \frac{g}{\sqrt{c_p T_0}}$ s^{-1} (derived in eq. (165)).

Our reference atmosphere in the troposphere will also be isothermal, but we choose a lower static stability value of $N \approx 1.2 \times 10^{-2}$ s^{-1}. How realistic is our choice of reference atmosphere? Hoskins and Bretherton [1972, fig. 3] refer to a study on different reference atmosphere models, which are the isothermal, adiabatic and the International Civil Aviation Organization (ICAO) reference atmospheres. They compared the corresponding pressure functions of height, $p(z)$, with each other and with observations. Observations from soundings show curves mostly between the isothermal and adiabatic ones such that the ICAO curve seems to be the most realistic choice. Our reference atmosphere resembles the ICAO closely and is therefore a good choice.

3.1 Scale analysis

This set of equations is very general and not well suited if we wish to describe frontal dynamics. Let us assume a quasi-stationary front from west to east and where the largest temperature gradients are in the north-south direction due to a confluence deformation field (see figure 9).

We will now perform a scale analysis to investigate the importance of the different terms in (12) and to reduce it into a two-dimensional form. The non-dimensional parameters are introduced in table 1.

The dimensionless equations then become:

$$\frac{UV}{l_y}\tilde{D}_t\tilde{u} - fV\tilde{v} = -\frac{\Phi}{l_x}\frac{\partial\tilde{\phi}}{\partial\tilde{x}} \tag{14a}$$

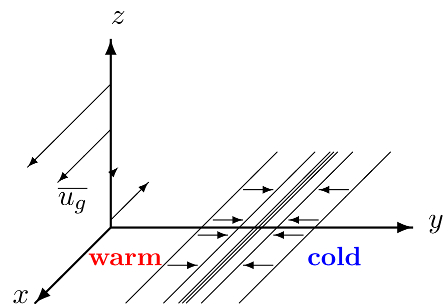


Figure 9 – Two-dimensional front model showing the cross-front component of a confluence deformation field (A) compressing the isotherms together. Thermal wind balance is indicated by the vertical zonal wind shear in accordance with a negative meridional temperature gradient

$x = l_x\tilde{x}$	$u = U\tilde{u}$	$b = B\tilde{b}$
$y = l_y\tilde{y}$	$v = V\tilde{v}$	$\phi' = \Phi\tilde{\phi}$
$z = h\tilde{z}$	$w = W\tilde{w}$	$t = T\tilde{t}$

Table 1 – Definition of non-dimensional parameters

$$\frac{V^2}{l_y}\tilde{D}_t\tilde{v} + fU\tilde{u} = -\frac{\Phi}{l_y}\frac{\partial\tilde{\phi}}{\partial\tilde{y}} \tag{14b}$$

$$\frac{BV}{l_y}\tilde{D}_t\tilde{b} + N^2W\tilde{w} = 0 \tag{14c}$$

$$\frac{U}{l_x}\frac{\partial\tilde{u}}{\partial\tilde{x}} + \frac{V}{l_y}\frac{\partial\tilde{v}}{\partial\tilde{y}} + \frac{W}{h}\frac{\partial\tilde{w}}{\partial\tilde{z}} = 0 \tag{14d}$$

$$B\tilde{b} = \frac{\Phi}{h}\frac{\partial\tilde{\phi}}{\partial\tilde{z}} \tag{14e}$$

where

$$\tilde{D}_t \equiv \frac{l_y}{V}\left(\frac{1}{T}\frac{\partial}{\partial\tilde{t}} + \frac{U}{l_x}\tilde{u}\frac{\partial}{\partial\tilde{x}} + \frac{V}{l_y}\tilde{v}\frac{\partial}{\partial\tilde{y}} + \frac{W}{h}\tilde{w}\frac{\partial}{\partial\tilde{z}}\right) \tag{15}$$

First we introduce the parameter $\beta = \frac{V}{U}$. Observations show that the cross-front velocity V is much smaller than the along-front velocity U such that $\beta \ll 1$. Furthermore, $l_y \ll l_x$ with typical values being 1000km along front l_x and 100km or less across front l_y. If we next consider the relative vorticity $\zeta = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$ than it is easily shown by considering the scales that $\frac{\partial v}{\partial x} \ll \frac{\partial u}{\partial y}$ and that the relative vorticity can be approximated by $\zeta \approx -\frac{\partial u}{\partial y}$.

Now we introduce the Rossby number as the ratio between the relative vorticity ζ and the planetary vorticity f (or equivalently: a measure of the importance of the change of momentum w.r.t. the Coriolis force) by:

$$\mathrm{Ro} = \frac{|\zeta|}{f} = \frac{U}{f l_y} \tag{16}$$

The first two equations in (14) and (15) can be rewritten in terms of Ro and β. Let us assume Ro being of order 1 for a strong front and check term by term if they play any role for such a front:

$$\tilde{D}_t \tilde{u} - \frac{1}{\mathrm{Ro}} \tilde{v} = -\frac{1}{\mathrm{Ro}} \frac{\Phi}{l_y} \frac{1}{fU} \left(\frac{U l_y}{V l_x} \right) \frac{\partial \tilde{\phi}}{\partial \tilde{x}} \tag{17a}$$

$$\mathrm{Ro}\beta^2 \tilde{D}_t \tilde{v} + \tilde{u} = -\frac{\Phi}{l_y} \frac{1}{fU} \frac{\partial \tilde{\phi}}{\partial \tilde{y}} \tag{17b}$$

$$\tilde{D}_t \equiv \frac{1}{\mathrm{Ro}} \left(\mathrm{Ro} \frac{\partial}{\partial \tilde{t}} + \underbrace{\frac{U}{l_x} \frac{l_y}{V}}_{I} \mathrm{Ro}\tilde{u} \frac{\partial}{\partial \tilde{x}} + \mathrm{Ro}\tilde{v} \frac{\partial}{\partial \tilde{y}} + \underbrace{\frac{W}{h} \frac{l_y}{V}}_{II} \mathrm{Ro}\tilde{w} \frac{\partial}{\partial \tilde{z}} \right) \tag{17c}$$

The quadratic β term in (17b) shows that accelerations across front are negligible and we assume GWB across front if $\frac{\Phi}{l_y}\frac{1}{fU}$ is of order 1 too (or else there is no balance at all). If the remaining terms in (14b) are combined with (14e) to filter $\tilde{\phi}$ then the result is an expression for thermal wind balance across front as we will show later.

In (17a), we see that the acceleration term is order 1. Assuming GWB along front would therefore be a bad approximation. The $\tilde{v}$ and $\frac{\partial \tilde{\phi}}{\partial \tilde{x}}$ terms are not equal, illustrated by the additional $\frac{U l_y}{V l_x}$ term on the r.h.s. The Coriolis force acts on the residual (i.e. the ageostrophic velocity) to accelerate the along-front flow.

Finally, (17c) illustrates in what direction advection of parameters is important. We will evaluate terms I and II. The change in the jet velocity along-front (figure 9) does not change much such that $\frac{U}{l_x} < \frac{V}{l_y} \approx \frac{W}{h}$, where the last equality comes from the continuity equation (14d). Hence, term I is negligible and term II is of order 1. However, we keep term I in our future derivations in order to conserve geostrophic non-divergence.

These approximations would eventually break down if: (i) the Rossby number $Ro \gg 1$ such that $Ro\beta^2$ becomes order 1 indicating a very sharp front where mixing processes would actually become relevant too or (ii)

$$
\begin{array}{c}
u = \underbrace{u_{jet}(y,z,t) + Ax}_{=u_g} \\
v = v_a(y,z,t) \underbrace{-Ay}_{=v_g} \\
w = w_a(y,z,t)
\end{array}
$$

Table 2 – Decomposition of velocity field

the curvature of the front becomes relevant (e.g. $l_x \sim l_y$) and a three-dimensional description would be needed.

3.2 Balanced equations

With this analysis in our mind, we continue with the general EOM for a 2D front that has now been reduced to:

$$
\frac{Du}{Dt} - fv = -\frac{\partial \phi'}{\partial x} \tag{18a}
$$

$$
\frac{Db}{Dt} + N^2 w = 0 \tag{18b}
$$

$$
\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{18c}
$$

$$
\left.\begin{array}{l} u = -\frac{1}{f}\frac{\partial \phi'}{\partial y} \\ b = \frac{\partial \phi'}{\partial z} \end{array}\right\} \frac{\partial b}{\partial y} = -f\frac{\partial u}{\partial z}
$$

$$
\frac{D}{Dt} = \frac{\partial}{\partial t} + v\frac{\partial}{\partial y} + w\frac{\partial}{\partial z}
$$

The scale analysis already gave a hint that the velocity field can be split up in balanced and unbalanced parts. The velocity field will now be split up in several components that are attributable to (i) the deformation field (confluence), (ii) the along-front geostrophic wind (jet stream) and (iii) the ageostrophic field (cross-frontal circulation). This is shown in table 2.

The jet stream, u_{jet}, is in TWB with the local meridional temperature gradient and is constant along-front. The jet does vary in the zonal direction in more advanced models (three-dimensional), which are suitable for describing local maxima in the jet stream, i.e. jet streaks. The cross-frontal circulation becomes zonally dependent too and obtains a more complex structure. At the entrance of the jet streak (where the air begins to accelerate) a thermally direct circulation is formed and at the exit region of the jet (deacceleration of air) an indirect circulation is formed.

The deformation field given by Ax and $-Ay$ represents only stretching deformation and favors frontogenesis in north-south direction. It is also considered as a balanced flow but then provided by the larger synoptic setting and constant in time and height. In the case of a confluence pattern, it represents the flow that balances the distribution of low and high pressure areas as shown in figure 5 for example. The Ax term does not play a dynamical role in the two-dimensional case as there are locally no gradients along-front such that advection has no effect here. Still, it is given here in order to have a non-divergent geostrophic deformation field.

Finally, the ageostrophic velocity field given by v_a, w_a forms the secondary cross-frontal circulation. Its role, as we will discuss later on, is to keep the front in TWB.

The resulting EOM become:

$$\frac{Du}{Dt} - fv_a = 0 \tag{20a}$$

$$\frac{Db}{Dt} + N^2 w_a = 0 \tag{20b}$$

$$\frac{\partial v_a}{\partial y} + \frac{\partial w_a}{\partial z} = 0 \tag{20c}$$

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + Ax\frac{\partial}{\partial x} + (v_a - Ay)\frac{\partial}{\partial y} + w_a\frac{\partial}{\partial z} \tag{20d}$$

In the continuity equation, the deformation field is non-divergent and disappears such that only the ageostrophic field remains. In the first equation (20a), v has been split up in two terms: one in geostrophic wind balance with $\frac{\partial \phi'}{\partial x}$ (such that both terms drop out) and an ageostrophic term, v_a.

The final simplification that we make here is the so-called geostrophic momentum approximation (GM), which in our case is particularly simple. In general, GM is an approximation based on the smallness of the Rossby number. GM means that the momentum can be approximated by the geostrophic component only (but can still be advected by ageostrophic velocity components!). The derivation is shown in two different manners in Hoskins [1975] in case of the three-dimensional primitive equations. For the two-dimensional form derived here, it becomes much easier as an along-front balanced flow was already assumed such that $u = u_g$. The zonal momentum equation (20a) can now be written as:

$$\frac{Du_g}{Dt} - fv_a = 0 \tag{21}$$

The consequences of the GM approximation are large. It means that only the geostrophic velocities change in time which are in balance at all times. The unbalanced ageostrophic components are now diagnostically determined, hence the GM approximation transforms (20) into the *balanced equations*. Any wave motions or instabilities (e.g. gravity waves) represented by periodic or exponential changes of ageostrophic velocities in time are filtered out. Moreover, this approximation allows us to use the invertibility principle of PV which is based on the assumption of a balanced atmosphere.

The balanced equations come in quasi-geostrophic and semi-geostrophic flavors. They shall both be examined in the next two sections.

4 QG theory

The Rossby number, defined as $Ro = \frac{U}{fL}$, is small on large (synoptic) scales. It implies that inertial forces are dominated by the Coriolis force which is approximately in balance with the pressure gradient force. We could do a *zero-order* approximation of $\vec{u}$ in terms of Ro. This would result in a velocity field constrained to both the geostrophic wind balance and the hydrostatic balance relations. Thus, a zero-order approximation would describe a steady-state flow. A zero-order approximation is rather useless in our study because we want to examine the evolution of fronts.

The QG theory results from a *first-order* approximation of $\vec{u}$ in terms of the Rossby number and allows for deviations from the balanced geostrophic flow $\vec{u_g}$, namely the ageostrophic velocity field $\vec{u_a}$ for which $|\vec{u_a}| \ll |\vec{u_g}|$. As a result, ageostrophic velocities restore balance but play a negligible role in the advection of other variables. Hence, $\frac{D}{Dt}$ becomes $\frac{D_g}{Dt}$ representing advection by the geostrophic velocity field only.

The resulting equations are:

$$\frac{D_g u_g}{Dt} - f v_a = 0 \tag{22a}$$

$$\frac{D_g b}{Dt} + N^2 w_a = 0 \tag{22b}$$

$$\frac{\partial v_a}{\partial y} + \frac{\partial w_a}{\partial z} = 0 \tag{22c}$$

$$\frac{\partial b}{\partial y} = -f \frac{\partial u_g}{\partial z} \tag{22d}$$

$$\frac{D_g}{Dt} = \frac{\partial}{\partial t} + u_g \frac{\partial}{\partial x} + v_g \frac{\partial}{\partial y} \tag{22e}$$

where $u_g = u_{jet} + Ax$ and $v_g = -Ay$ are the balanced flow fields.

It shows that the ageostrophic and geostrophic velocity fields are coupled in equations (22a) and (22b). Their physical connection can be understood by splitting up the geostrophic velocities from the ageostrophic velocities. In the next subsection, an equation for the ageostrophic component will be derived. After that, the ageostrophic velocities will be filtered out resulting in a (PV) advection equation.

4.1 Cross-frontal circulation

The GM approximation applied earlier implies that the ageostrophic velocities are not advected by the geostrophic field, nor do they change in time locally. This means that the ageostrophic field adjusts itself immediately (or on very short time scales) to the evolving geostrophic field (which is departing from TWB) to restore TWB. To find an expression for v_a, w_a, the Lagrangian time term needs to be removed. It will be assumed that the geostrophic field remains in TWB in the following derivation.

First, take $f\frac{\partial}{\partial z}$ of (22a) and $\frac{\partial}{\partial y}$ of (22b). Then rewrite the first two equations such that these derivatives are placed inside $\frac{D_g}{Dt}$. This results in:

$$\frac{D_g}{Dt}\left(f\frac{\partial u_g}{\partial z}\right) - f^2\frac{\partial v_a}{\partial z} = -f\frac{\partial u_g}{\partial z}\frac{\partial u_g}{\partial x} \tag{23a}$$

$$\frac{D_g}{Dt}\left(\frac{\partial b}{\partial y}\right) + N^2\frac{\partial w_a}{\partial y} = -\frac{\partial v_g}{\partial y}\frac{\partial b}{\partial y} \tag{23b}$$

where the extra terms on the rhs arise in a similar way as in (8) according to:

$$f\frac{\partial}{\partial z}\left(\frac{D_g u_g}{Dt}\right) = f\frac{\partial u_g}{\partial z}\frac{\partial u_g}{\partial x} + \underbrace{f\frac{\partial v_g}{\partial z}\frac{\partial u_g}{\partial y}}_{=0} + \frac{D_g}{Dt}\left(f\frac{\partial u_g}{\partial z}\right)$$

$$\frac{\partial}{\partial y}\left(\frac{D_g b}{Dt}\right) = \underbrace{\frac{\partial u_g}{\partial y}\frac{\partial b}{\partial x}}_{=0} + \frac{\partial v_g}{\partial y}\frac{\partial b}{\partial y} + \frac{D_g}{Dt}\left(\frac{\partial b}{\partial y}\right)$$

The terms on the rhs of (23) can be rewritten as follows (using TWB):

$$-f\frac{\partial u_g}{\partial z}\frac{\partial u_g}{\partial x} = -Af\frac{\partial u_{jet}}{\partial z} = A\frac{\partial b}{\partial y} = Q_2$$

$$-\frac{\partial v_g}{\partial y}\frac{\partial b}{\partial y} = A\frac{\partial b}{\partial y} = Q_2$$

so that (23) becomes:

$$\frac{D_g}{Dt}\left(f\frac{\partial u_g}{\partial z}\right) - f^2\frac{\partial v_a}{\partial z} = Q_2 \tag{26a}$$

$$\frac{D_g}{Dt}\left(\frac{\partial b}{\partial y}\right) + N^2\frac{\partial w_a}{\partial y} = Q_2 \tag{26b}$$

Our definition of frontogenetic forcing $Q_2 = A\frac{\partial b}{\partial y}$ illustrates that we neglect all other frontogenetic processes except for confluence given by the interplay between A and $\frac{\partial b}{\partial y}$. In our model, frontogenesis occurs when Q_2 is negative! (as the buoyancy gradient is negative too). Equation (26) shows how frontogenetic processes destroy TWB in two opposite ways:

Eq. (26a) the negative value of Q_2 on the rhs shows that frontogenesis *reduces* the vertical wind shear of the jet stream. Advection of momentum by stretching deformation causes the local maximum (minimum) of the jet stream velocity at the front to decrease (increase) in the upper (lower) parts of the atmosphere.

Eq. (26b) At the same time, frontogenesis results in a tendency to amplify the gradient $\frac{\partial b}{\partial y}$ in time, which means that the front becomes stronger.

The last point was already seen from a kinematic point of view. But it is interesting to see from a dynamical point of view how the vertical wind shear is influenced by frontogenesis in the opposite way.

The first terms in (26) can be removed by summing (26a) and (26b) using TWB, resulting in an equation for the ageostrophic circulation:

$$N^2\frac{\partial w_a}{\partial y} - f^2\frac{\partial v_a}{\partial z} = 2Q_2 \tag{27}$$

Using the non-divergence of the ageostrophic field, given by (22c), a streamfunction is introduced as $(-\frac{\partial \psi}{\partial z}, \frac{\partial \psi}{\partial y}) = (v_a, w_a)$ and the expression becomes:

$$N^2\frac{\partial^2 \psi}{\partial y^2} + f^2\frac{\partial^2 \psi}{\partial z^2} = 2Q_2 \tag{28}$$

The negative value of Q_2 results in positive values for ψ. This corresponds to a thermal direct CFC (clockwise circulation).

This is a classic Poisson equation characterized by a global solution for ψ forced by a local Q_2. This means that zero-flow boundary conditions have to be chosen far away from the geostrophic forcing area in the center of a domain. This shall be discussed in the model section in chapter 8.

In figure 2 a sketch was shown of the CFC as it would look like for the QG case. The CFC restores TWB in two ways:

- In the warm sector (W) rising air cools adiabatically and similarly in the cold sector (C) air cools adiabatically. This frontolytic effect of the CFC is particularly strong in mid troposphere (around 500 hPa) where vertical velocities are largest.

- The Coriolis force deflects the meridional components of the CFC to the right in the Northern hemisphere. Thus, in western direction in the lower troposphere and eastern direction in the upper troposphere. As a result, horizontal momentum of the CFC is transferred to the along-front winds. The geostrophic wind increases in opposing directions such that the vertical windshear increases throughout the troposphere. A new TWB is gained in accordance with a larger thermal gradient.

4.2 PV evolution

We start again from the basic QG equations given by (22). In a similar way as in the previous subsection, we can filter out the ageostrophic component (i.e. the CFC) this time. We will derive an equation for the evolution of the geostrophic wind and buoyancy. It was shown in the previous subsection by (28) how the ageostrophic field constantly adjusts itself to the evolving fields of the two geostrophic fields.

We will introduce the concept of quasi-geostrophic potential vorticity (hereafter QGPV) being the conserved field from which all other fields can be derived. A first glimpse of the advantage of this approach will be shown here by considering two approaches on how the QG set of equations can be (numerically) solved.

Approach 1:

Given an initial front with a constant deformation field, it is possible to derive the CFC. Once the CFC is known, its effect on the evolution of the geostrophic fields follows from (22a) and (22b). This means that we can directly calculate the new geostrophic fields b and u_g changing through: (i) advection (geostrophic), (ii) Coriolis effect on v_a and (iii) adiabatic warming/cooling effect on w_a. Once the new geostrophic fields are derived, one can calculate the new CFC again, etcetera.

It is clear from this approach what physical processes play a role. However, we can rewrite the equations in such a way that information on the CFC is no longer necessary! This is understandable because we know that the ageostrophic circulation does not play a (dynamical) role in time in QG theory. It is a diagnostic component which purpose is to restore atmospheric balance.

Approach 2:

This approach makes use of the non-divergent property of the CFC in order to rewrite the set of equations in terms of the conservative quantity q_{QG}.

First, take $-\frac{\partial}{\partial y}$ of (22a) and $\frac{f}{N^2}\frac{\partial}{\partial z}$ of (22b):

$$-\frac{\partial}{\partial y}\left(\frac{D_g u_g}{Dt}\right) + f\frac{\partial v_a}{\partial y} = 0 \tag{29a}$$

$$\frac{f}{N^2}\frac{\partial}{\partial z}\left(\frac{D_g b}{Dt}\right) + f\frac{\partial w_a}{\partial z} = 0 \tag{29b}$$

$$\frac{\partial v_a}{\partial y} + \frac{\partial w_a}{\partial z} = 0 \tag{29c}$$

Next, sum up both equations and put the derivatives inside the material time derivatives, using the following relations again:

$$-\frac{\partial}{\partial y}\left(\frac{D_g u_g}{Dt}\right) = \underbrace{-\frac{\partial u_g}{\partial y}\frac{\partial u_g}{\partial x} - \frac{\partial v_g}{\partial y}\frac{\partial u_g}{\partial y}}_{=0} + \frac{D_g}{Dt}\left(-\frac{\partial u_g}{\partial y}\right) \tag{30a}$$

$$\frac{f}{N^2}\frac{\partial}{\partial z}\left(\frac{D_g b}{Dt}\right) = \frac{f}{N^2}\underbrace{\left[\frac{\partial u_g}{\partial z}\frac{\partial b}{\partial x} + \frac{\partial v_g}{\partial z}\frac{\partial b}{\partial y}\right]}_{=0} + \frac{D_g}{Dt}\left(\frac{f}{N^2}\frac{\partial b}{\partial z}\right) \tag{30b}$$

where in both cases the terms on the rhs cancel out using geostrophic non-divergence ($\frac{\partial u_g}{\partial x} = -\frac{\partial v_g}{\partial y}$) and general TWB relations ($\frac{\partial b}{\partial x} = f\frac{\partial v_g}{\partial z}, \frac{\partial b}{\partial y} = -f\frac{\partial u_g}{\partial z}$).

This means that the sum of (30a) and (30b) takes a particular simple form:

$$\frac{D_g}{Dt}\left(\frac{f}{N^2}\frac{\partial b}{\partial z} - \frac{\partial u_g}{\partial y}\right) = 0 \tag{31}$$

This shows that the sum of buoyancy stratification and the geostrophic vorticity is conserved!

If we introduce the streamfunction (for an incompressible geostrophic flow) by $\Psi_g = \frac{\phi'}{f}$, then the zero order balances GWB and HB become: $u_g = -\frac{\partial \Psi_g}{\partial y}$ and $b = f\frac{\partial \Psi_g}{\partial z}$. Substitution into (31) results in:

$$\frac{D_g q_{QG}}{Dt} = 0 \tag{32}$$

where q_{QG} is given by:

$$q_{QG} = \frac{f^2}{N^2}\frac{\partial^2 \Psi_g}{\partial z^2} + \frac{\partial^2 \Psi_g}{\partial y^2} \tag{33}$$

This is the conservation of the quasi-geostrophic potential vorticity. Only geostrophic advection plays a dynamic role. This approach has the advantage over the first approach that no information is required about the CFC. The restoring effect of v_a and w_a, by changing b and u_g as described in (26), is now implicitly derived and decoupled from the evolution of geostrophic fields.

Figure 10 summarizes the set of equations that we have derived for QG theory. It is given in a flow diagram to visualize the right order for programming purposes. It forms a decoupled system. That means that all other variables can be calculated from q_{QG} once we have derived the new q_{QG} from a given two-dimensional deformation field A.

To derive Ψ_g from the PV-inversion, we also need proper boundary conditions which may change in time. The same holds for ψ in the other elliptical equation. We will specify the boundary conditions that we used for the two equations in chapter 8.

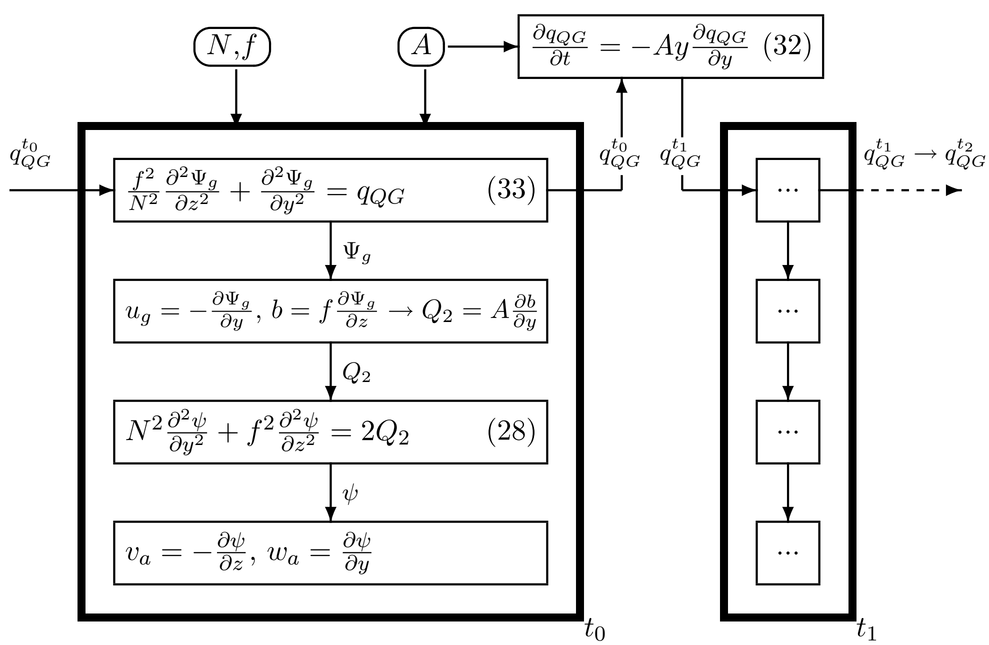


Figure 10 – Overview on the QG set of equations. The parameters within the ovals are constants. The equations inside the thick black box are used to derive all variables diagnostically from QGPV-inversion at one timestep. The evolution in time is in the right direction and is described by the QG advection equation.

5 SG theory

The QGPV model does a great job even when frontal gradients become large. However, it fails to represent both the tropopause folding and the formation of a realistic surface front. Still we can predict from QG theory these two important processes:

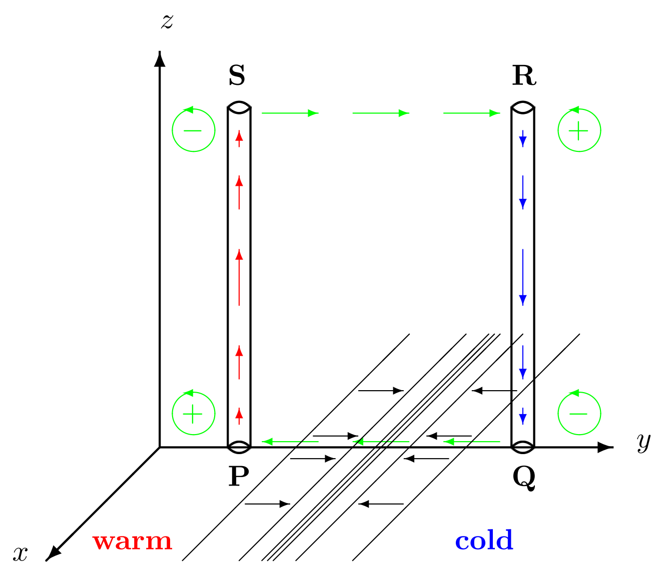


Figure 11 – Same as figure 2, but now highlighting the cross-frontal circulation. The red and blue arrows denote positive and negative domain of w_a. The green arrows denotes v_a. The green circles represent the vorticity changes due to vertical stretching and contraction effects near **P**, **Q**, **R** and **S**.

Firstly, the meridional ageostrophic flow is convergent $\frac{\partial v_a}{\partial y} < 0$ at P in fig. 11. This implies that the front would shift towards the warm sector. The same holds for R in the upper troposphere where the front would shift towards the cold sector. QG theory thus predicts a sloping front if it would allow the horizontal component of the CFC to play a dynamical role.

Secondly, when we examine the vertical component of the CFC, this frontal shift is also visible in terms of a vorticity balance across-front. In the vorticity equation, which we could derive from (20a) by taking the $-\frac{\partial}{\partial y}$ derivative, vertical stretching and contraction mechanisms are a source or sink of vorticity.

One can understand the role of w_a by considering the vorticity of an air column between two isentropic surfaces (e.g. surface θ_2 above surface

θ_1) near P in figure 11. The vorticity would increase when the distance between the two isentropic layers increases. If w_a is larger at θ_2 than at θ_1, the air column in between would be stretched resulting in additional vorticity near P. The same would hold for R. The air column would be compressed by w_a near Q and S. Vorticity destruction would occur such that the vorticity becomes less negative. QG theory predicts strong fronts with large vorticities near P, if it would allow the vertical component of the CFC to play a dynamical role.

When the nonlinear term $\zeta_g \frac{\partial w_a}{\partial z}$ becomes of the same order of magnitude as f, then the circulation is strongly developed and its dynamical role can no longer be neglected. In terms of the Rossby number Ro $= \frac{|\zeta|}{f}$, if Ro becomes order one, then we need to reconsider the QG approximation as we shall do now.

The SG equations are given by (18) and are shown here for reference:

$$\frac{Du_g}{Dt} - f v_a = 0 \tag{34a}$$

$$\frac{Db}{Dt} + N^2 w_a = 0 \tag{34b}$$

$$\frac{\partial v_a}{\partial y} + \frac{\partial w_a}{\partial z} = 0 \tag{34c}$$

$$\frac{\partial b}{\partial y} = -f \frac{\partial u_g}{\partial z} \tag{34d}$$

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + u_g \frac{\partial}{\partial x} + (v_g + v_a) \frac{\partial}{\partial y} + w_a \frac{\partial}{\partial z} \tag{34e}$$

The main difference are the two additional ageostrophic velocities in the total time derivative $\frac{D}{Dt}$.

In the next two subsections, the equations for the CFC and the q_{SG} evolution equation will be derived. The difficulty here is that the advection also contains ageostrophic terms.

5.1 Ageostrophic component

Using (34a) and (34b), one can apply the same steps to obtain the equation for the SG CFC. We start with manipulating both equations by taking the derivatives to y and z again (compare with (23)).

$$\frac{D}{Dt}\left(f \frac{\partial u_g}{\partial z}\right) - f^2 \frac{\partial v_a}{\partial z} = \underbrace{-f \frac{\partial u_g}{\partial z} \frac{\partial u_g}{\partial x}}_{Q_2} - f \frac{\partial v_a}{\partial z} \frac{\partial u_g}{\partial y} - f \frac{\partial w_a}{\partial z} \frac{\partial u_g}{\partial z} \tag{35a}$$

$$\frac{D}{Dt}\left(\frac{\partial b}{\partial y}\right) + N^2\frac{\partial w_a}{\partial y} = \underbrace{-\frac{\partial v_g}{\partial y}\frac{\partial b}{\partial y}}_{Q_2} - \frac{\partial v_a}{\partial y}\frac{\partial b}{\partial y} - \frac{\partial w_a}{\partial y}\frac{\partial b}{\partial z} \tag{35b}$$

Here, two extra terms are added on the rhs originating from the full advection operator. On the rhs of (35b), these represents confluence and tilting mechanisms by the *a*geostrophic field. One could see them as additional deformation parameters that together with Q_2 enhances frontogenesis. We retain our definition for Q_2 being the geostrophic forcing here and put the ageostrophic terms on the lhs forming the SG version of the CFC.

Now we can use TWB again to remove the time dependence by summing up both equations :

$$\left(N^2 + \frac{\partial b}{\partial z}\right)\frac{\partial w_a}{\partial y} - \left(f^2 - f\frac{\partial u_g}{\partial y}\right)\frac{\partial v_a}{\partial z} + \frac{\partial v_a}{\partial y}\frac{\partial b}{\partial y} + f\frac{\partial w_a}{\partial z}\frac{\partial u_g}{\partial z} = 2Q_2 \tag{36}$$

which can be further reduced to:

$$\left(N^2 + \frac{\partial b}{\partial z}\right)\frac{\partial w_a}{\partial y} - f\left(f - \frac{\partial u_g}{\partial y}\right)\frac{\partial v_a}{\partial z} + \left(\frac{\partial v_a}{\partial y} - \frac{\partial w_a}{\partial z}\right)\frac{\partial b}{\partial y} = 2Q_2 \tag{37}$$

In terms of the streamfunction ψ, defined similar to the QG case, the SG-CFC equation becomes:

$$N^2_{eff}\frac{\partial^2\psi}{\partial y^2} + 2S^2\frac{\partial^2\psi}{\partial y\partial z} + F^2\frac{\partial^2\psi}{\partial z^2} = 2Q_2 \tag{38}$$

where $N^2_{eff} \equiv N^2 + \frac{\partial b}{\partial z}$ is the total static stability, $S^2 \equiv -\frac{\partial b}{\partial y}$ the baroclinicity and $F^2 \equiv f(f - \frac{\partial u_g}{\partial y})$ the inertial stability. This equation is a simplified form (excluding diabatic heating) of the original one firstly derived by Sawyer and Eliassen. We shall refer to (38) as the Sawyer-Eliassen equation (hereafter SE).

This type of equation is a linear, second-order partial differential equation. Such equations can be classified as elliptical, parabolic or hyperbolic depending on the discriminant $D = (2S^2)^2 - 4N^2_{eff}F^2$ or the more well-known criterium:

$$N_{eff}F^2 - S^4 > 0 \tag{39}$$

This criterium corresponds to an atmosphere that is *symmetrically stable.*

The concept of symmetric stability can be understood by examining the three stability parameters N^2_{eff}, F^2 and S^2. Firstly, assume a statically stable atmosphere $N^2_{eff} > 0$ such that vertical displaced air parcels return to their original altitude. This corresponds to $\frac{\partial\theta}{\partial z} > 0$. Secondly, assume a inertial stable atmosphere $F^2 > 0$ such that latitudal displaced air columns return to their original latitude. This corresponds to $-\frac{\partial M_g}{\partial y} > 0$ where $M_g = u_g - fy$.

Figure 12 shows an illustration, very similar to the large scale baroclinic instability settings (some call it a second-order baroclinic instability). It shows that symmetric instability occurs when the θ surfaces are more steeply sloped than geostrophic absolute momentum M_g surfaces, even when the atmosphere is both statically and inertially stable!

We can now compare the inertial and static stabilities with the baroclinicity S^2 and distinguish three cases:

- If $N_{eff}F^2 - S^4 < 0$ then SE becomes a hyperbolic equation which we can interpret physically as symmetric instability. Hyperbolic equations are generally used for describing wave propagation. If $S^2 = -\frac{\partial b}{\partial y}$ becomes too large (i.e. the front becomes too strong) then symmetric instability sets in and exponentially growing waves will develop. The resulting exponential growth propagates mainly close along the isentropes resulting in so-called slantwise convection (Thorpe and Emanuel [1985]). This type of convection could occur in statically stable atmospheres and is responsible for non-convective precipitation.

- If $N_{eff}F^2 - S^4 = 0$ then SE becomes a parabolic equation. This case has been examined by Hoskins and Bretherton [1972] being the zero potential vorticity case (ZPV). Such conditions are rare in a dry atmosphere. However, in a moist atmosphere it becomes interesting as moist isentropes are steeper orientated than 'dry' isentropes, analogous to conditional static instability. One can compare the moist isentropes with the absolute momentum surfaces like in figure 12 to assess the presence of conditional symmetric instability (CSI). Thorpe and Emanuel [1985] show that near neutral symmetric conditions correspond to narrowing and intensification of the lifting part of the CFC.

- If $N_{eff}F^2 - S^4 > 0$ then SE becomes an elliptic equation which can be interpreted as a *balanced* solution for ψ! Furthermore, given the frontogenetic forcing Q_2, the whole domain will adjust itself in order to restore TWB. In other words: a local forcing in an elliptical equation has a typical 'action at distance' effect, physically established by the CFC. We can only solve ψ from the SE equation when this important condition is met.

It is shown in Hoskins [1982] and by others that the symmetric stability $N_{eff}^2 F^2 - S^4$ is a conserved quantity also known as potential vorticity. We shall refer to it as the semi-geostrophic potential vorticity (hereafter SGPV) given by (40).

$$q_{SG} = N_{eff}^2 F^2 - S^4 \tag{40}$$

Its conservation shall be shown in the next subsection.

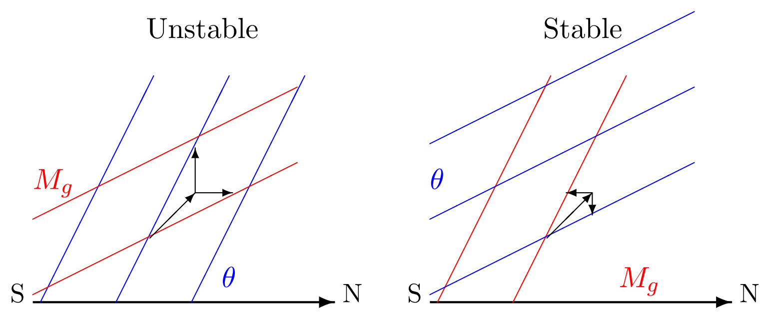


Figure 12 – Conceptual picture of symmetric instability in a statically and inertial stable atmosphere.

5.2 PV evolution

We start from the SG equations given by (34). The material time derivative in (34a) and (34b) now also contains ageostrophic velocities. Now calculate $-f\frac{\partial}{\partial y}$ of (34a) and $\frac{\partial}{\partial z}$ of (34b) and put the derivatives inside the material derivatives again:

$$-f\frac{D}{Dt}\left(\frac{\partial u_g}{\partial y}\right) + f^2\frac{\partial v_a}{\partial y} = f\frac{\partial v_a}{\partial y}\frac{\partial u_g}{\partial y} + f\frac{\partial w_a}{\partial y}\frac{\partial u_g}{\partial z} \tag{41a}$$

$$\frac{D}{Dt}\left(\frac{\partial b}{\partial z}\right) + N^2\frac{\partial w_a}{\partial z} = -\frac{\partial v_a}{\partial z}\frac{\partial b}{\partial y} - \frac{\partial w_a}{\partial z}\frac{\partial b}{\partial z} \tag{41b}$$

where the extra ageostrophic terms on the rhs were not present in the QGPV case. Using the definitions of N_{eff}^2 and F^2, (41) reduces to:

$$-\frac{D}{Dt}\left(f\frac{\partial u_g}{\partial y}\right) + F^2\frac{\partial v_a}{\partial y} = f\frac{\partial w_a}{\partial y}\frac{\partial u_g}{\partial z} \tag{42a}$$

$$\frac{D}{Dt}\left(\frac{\partial b}{\partial z}\right) + N_{eff}^2\frac{\partial w_a}{\partial z} = -\frac{\partial v_a}{\partial z}\frac{\partial b}{\partial y} \tag{42b}$$

Next, we multiply (42a) by N_{eff}^2 and (42b) by F^2 and sum up both equations. Use ageostrophic non-divergence to filter the ageostrophic terms on the lhs:

$$-N_{eff}^2 \frac{D}{Dt}\left(f\frac{\partial u_g}{\partial y}\right) + F^2 \frac{D}{Dt}\left(\frac{\partial b}{\partial z}\right) = N_{eff}^2 f \frac{\partial w_a}{\partial y}\frac{\partial u_g}{\partial z} - F^2 \frac{\partial v_a}{\partial z}\frac{\partial b}{\partial y} \tag{43}$$

Now use TWB to write this as:

$$-N_{eff}^2 \frac{D}{Dt}\left(f\frac{\partial u_g}{\partial y}\right) + F^2 \frac{D}{Dt}\left(\frac{\partial b}{\partial z}\right) = -\frac{\partial b}{\partial y}\left[N_{eff}^2 \frac{\partial w_a}{\partial y} + F^2 \frac{\partial v_a}{\partial z}\right] \tag{44}$$

The rhs still contains ageostrophic terms and we use (34a) and (34b) again to filter these terms out using the following relations (where derivatives to y and z are taken and placed inside $\frac{D}{Dt}$ again)

$$\frac{D}{Dt}\left(f\frac{\partial u_g}{\partial z}\right) = \underbrace{-f\frac{\partial u_g}{\partial z}\frac{\partial u_g}{\partial x} - f\frac{\partial v_g}{\partial z}\frac{\partial u_g}{\partial y}}_{Q_2} - f\frac{\partial w_a}{\partial z}\frac{\partial u_g}{\partial z} + \underbrace{f^2\frac{\partial v_a}{\partial z} - f\frac{\partial v_a}{\partial z}\frac{\partial u_g}{\partial y}}_{F^2\frac{\partial v_a}{\partial z}}$$

$$\frac{D}{Dt}\left(\frac{\partial b}{\partial y}\right) = \underbrace{-\frac{\partial u_g}{\partial y}\frac{\partial b}{\partial x} - \frac{\partial v_g}{\partial y}\frac{\partial b}{\partial y}}_{Q_2} - \frac{\partial v_a}{\partial y}\frac{\partial b}{\partial y} - \underbrace{\frac{\partial w_a}{\partial y}\frac{\partial b}{\partial z} - N^2\frac{\partial w_a}{\partial y}}_{N_{eff}^2\frac{\partial w_a}{\partial y}}$$

Next, we can write the rhs of (44) as:

$$\begin{aligned} -\frac{\partial b}{\partial y}\left[N_{eff}^2 \frac{\partial w_a}{\partial y} + F^2 \frac{\partial v_a}{\partial z}\right] &= -\frac{\partial b}{\partial y}\left[Q_2 - \frac{D}{Dt}\left(\frac{\partial b}{\partial y}\right) - \frac{\partial v_a}{\partial y}\frac{\partial b}{\partial y} + \right. \\ &\qquad \left. \frac{D}{Dt}\left(f\frac{\partial u_g}{\partial z}\right) - Q_2 + f\frac{\partial w_a}{\partial z}\frac{\partial u_g}{\partial z}\right] \\ &= 2\frac{\partial b}{\partial y}\frac{D}{Dt}\left(\frac{\partial b}{\partial y}\right) \end{aligned} \tag{46}$$

such that (44) becomes:

$$-N_{eff}^2 \frac{D}{Dt}\left(f\frac{\partial u_g}{\partial y}\right) + F^2 \frac{D}{Dt}\left(\frac{\partial b}{\partial z}\right) - 2\frac{\partial b}{\partial y}\frac{D}{Dt}\left(\frac{\partial b}{\partial y}\right) = 0 \tag{47}$$

Our last step is to collect all terms under one material derivative, starting by writing out all terms explicitly:

$$\begin{aligned} -N^2\frac{D}{Dt}\left(f\frac{\partial u_g}{\partial y}\right) - \frac{\partial b}{\partial z}\frac{D}{Dt}\left(f\frac{\partial u_g}{\partial y}\right) + f^2\frac{D}{Dt}\left(\frac{\partial b}{\partial z}\right) - \\ f\frac{\partial u_g}{\partial y}\frac{D}{Dt}\left(\frac{\partial b}{\partial z}\right) - 2\frac{\partial b}{\partial y}\frac{D}{Dt}\left(\frac{\partial b}{\partial y}\right) = 0 \end{aligned} \tag{48}$$

where the second and the fourth term can be combined together, resulting in:

$$-\frac{D}{Dt}\left(fN^2\frac{\partial u_g}{\partial y}\right)+\frac{D}{Dt}\left(f^2\frac{\partial b}{\partial z}\right)-\frac{D}{Dt}\left(f\frac{\partial u_g}{\partial y}\frac{\partial b}{\partial z}\right)-\frac{D}{Dt}\left(\frac{\partial b}{\partial y}\right)^2=0 \quad (49)$$

Next, we divide by a constant factor ($N^2 f$) and put all terms together:

$$\frac{Dq_{SG}}{Dt}=0 \quad (50)$$

where q_{SG} is given by:

$$q_{SG}=\frac{f}{N^2}\frac{\partial b}{\partial z}-\frac{\partial u_g}{\partial y}-\frac{1}{N^2}\frac{\partial u_g}{\partial y}\frac{\partial b}{\partial z}-\frac{1}{fN^2}\left(\frac{\partial b}{\partial y}\right)^2 \quad (51)$$

We introduce the streamfunction Ψ_g according to $\left(\frac{\partial \Psi_g}{\partial y},\frac{\partial \Psi_g}{\partial z}\right)=\left(-u_g,\frac{b}{f}\right)$ again and rewrite the expression for q_{SG} as:

$$q_{SG}=\underbrace{\frac{f^2}{N^2}\frac{\partial^2\Psi_g}{\partial z^2}+\frac{\partial^2\Psi_g}{\partial y^2}}_{q_{QG}}+\frac{f}{N^2}\frac{\partial^2\Psi_g}{\partial y^2}\frac{\partial^2\Psi_g}{\partial z^2}-\frac{f}{N^2}\left(\frac{\partial^2\Psi_g}{\partial y\partial z}\right)^2 \quad (52)$$

We notice the presence of two additional nonlinear terms on the rhs, illustrating that the q_{SG} is an extension of the q_{QG}, defined in (33). The inversion principle of PV allows us to derive Ψ_g from q_{SG} using (52). In other words: $\Psi_g = N_1^{-1}(q_{SG})$ where N_1 is a nonlinear operator in the SG case.

We derived a particular form of q_{SG}. The more general expression for q_{SG} was given in previous subsection and corresponds to a measure of symmetric stability: $q_{SG}=N_{eff}^2F^2-S^4$. The only difference is the inclusion of the f term in the vorticity term. Thus, the relative vorticity term $\frac{\partial^2\Psi_g}{\partial y^2}$ is replaced by the absolute vorticity $f+\frac{\partial^2\Psi_g}{\partial y^2}$. In the next subsection, a short derivation from (52) to the more general form is shown. We use PV defined in (52) because it is a clear extension of the quasi-geostrophic form and both SG and QG PV values are of the same order of magnitude. Typical values are in the order of 10^{-4} s^{-1} given in units of vorticity.

The conservation of q_{SG} in (50) shows that if the atmosphere is initially symmetrically stable on the whole domain, it will remain stable! Redistribution of q_{SG} by advection only can never result in negative q_{SG} values on that domain. This is of course based on the assumption that there are no sources and sinks for q_{SG} on the large scale, meaning that diffussion, diabatic heating effects, etc are negligible.

Figure 13 summarizes the set of equations that we have derived for SG theory. One can compare it with the flow diagram of the QG equations shown in figure 10 to see the following differences:

- It has now become a *coupled* system. Once all the variables are derived from PV, the ageostrophic velocity components are used for calculating new PV values by the advection equation.
- The PV-inversion equation has become nonlinear.
- The Sawyer-Eliassen equation is still linear, but can only be solved for symmetric stable conditions (when $N_{eff}^2 F^2 - S^4 > 0$). Symmetric stability corresponds to $q_{SG} > 0$.
- From the conservation of PV, we conclude that if the atmosphere is initially symmetrically stable, it remains stable for all times.

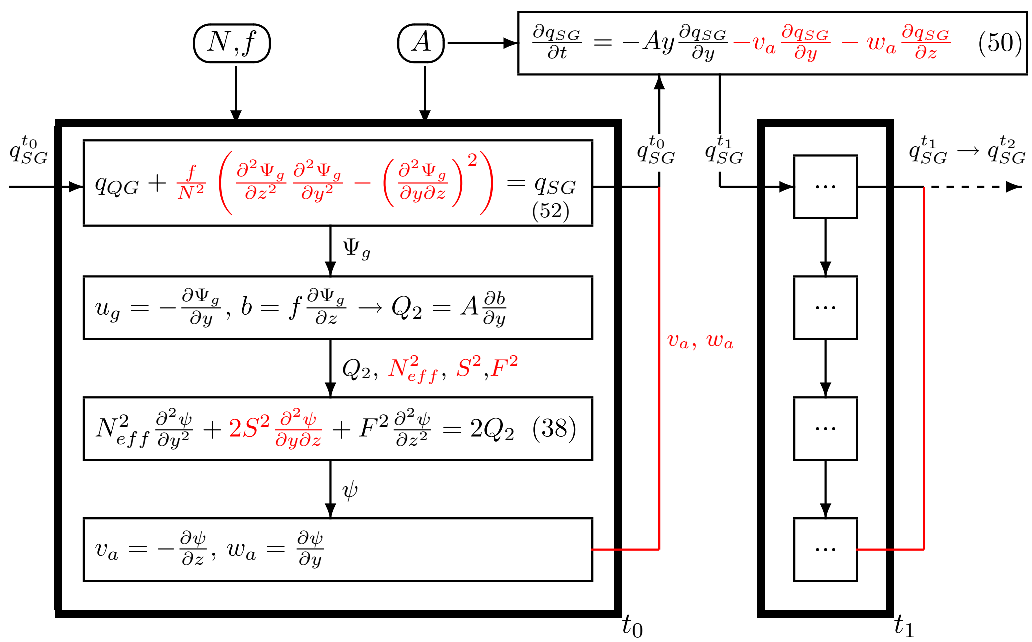


Figure 13 – Overview on the SG set of equations. The parameters within the ovals are constants. The equations inside the thick black box are used to derive all variables diagnostically from SGPV-inversion at one timestep. The evolution in time is in the right direction and is described by the SG advection equation. All changes with respect to figure 10 are colored in red.

5.2.1 Other forms of PV

Other PV forms are given here in comparison with q_{SG}. Firstly, the q_{SG} form $N_{eff}^2F^2 - S^4$ is more common and used in Hoskins [1982] for example. If the relative vorticity term $-\frac{\partial u_g}{\partial y}$ is replaced by the absolute vorticity $f - \frac{\partial u_g}{\partial y}$, the expression for q_{SG} in (51) changes to:

$$q_{SG} = \frac{f}{N^2}\frac{\partial b}{\partial z} + \left(f - \frac{\partial u_g}{\partial y}\right) - \frac{1}{N^2}\frac{\partial u_g}{\partial y}\frac{\partial b}{\partial z} - \frac{1}{fN^2}\left(\frac{\partial b}{\partial y}\right)^2 \tag{53}$$

Which can be rewritten as:

$$q_{SG} = \frac{1}{N^2 f}\left[f^2\frac{\partial b}{\partial z} + N^2f^2 - N^2f\frac{\partial u_g}{\partial y} - f\frac{\partial u_g}{\partial y}\frac{\partial b}{\partial z} - \left(\frac{\partial b}{\partial y}\right)^2\right] \tag{54}$$

With absolute vorticity f included, the terms can be arranged in terms of N_{eff}^2, F^2 and S^2:

$$q_{SG} = \frac{1}{N^2 f}\left[\left(N^2 + \frac{\partial b}{\partial z}\right) f\left(f - \frac{\partial u_g}{\partial y}\right) - \left(\frac{\partial b}{\partial y}\right)^2\right] \tag{55}$$

And finally:

$$q_{SG} = \frac{1}{N^2 f}\left[N_{eff}^2F^2 - S^4\right] \tag{56}$$

Apart from the f-plane approximation and a constant scaling factor N^2f, our definition of q_{SG} given in (52) conforms to the definition of q_{SG} given above. Typical values of q_{SG} are in the order of $10^{-12}\mathrm{s}^{-4}$.

We can also make a comparison with the isentropic form of PV (IPV) (introduced by Reed and Sanders [1953]) given by:

$$IPV = -g\left(f + \zeta_\theta\right)\frac{\partial \theta}{\partial p} \tag{57}$$

The IPV is a product of absolute vorticity and static stability. Typical units of IPV are expressed in PVU where 1 PVU = 10^{-6} $\mathrm{Kkg^{-1}m^2s^{-1}}$. Motions are constrained to isentropic surfaces in absence of friction and cross-surface transport by diabatic processes. In that case, IPV is conservative like the other PV forms. This means that there is a potential for creating vorticity by changing latitude f or by changing the distance between two isentropic layers $\frac{\partial \theta}{\partial p}$.

A nice application of its use is tropopause folding from an isentropic PV view. Davies and Rossa [1998] discusses how the high-PV stratospheric air, constrained to move along isentropes, meets the lower low-PV tropospheric air on an isentropic surface. High-PV air is characterized by strong stratification $\frac{\partial \theta}{\partial p}$. In their view, the intrusion of high-PV air downward is linked to a large horizontal thermal gradient such that upper-level fronts are identified as zones of strong IPV gradients on isentropic surfaces. The intensification of such IPV gradients by deformation fields, defined on isentropic surfaces, is known as PV-frontogenesis. In the end, this means that the same approach can be taken (i.e. studying deformation fields on isentropic surfaces) to understand the evolution of a tropopause fold.

In the Appendix D, it is shown how the hydrostatic form of IPV can be scaled to the variable $\hat{q}$, from which the SGPV form in (52) can be derived again.

6 Geostrophic coordinates

When applying the semi-geostrophic approximation, we saw that the Sawyer-Eliassen equation includes a cross partial derivative term and that the PV-inversion equation even becomes nonlinear. Solving both equations requires more advanced numerical methods as we shall discuss later on. There is a way to avoid this by using a different coordinate system, namely the geostrophic coordinates (X, Y, Z, T). The SG equations in geostrophic coordinates has been first derived by Eliassen [1962], studied in more detail in Hoskins and Bretherton [1972] and Hoskins [1975], and has become standard in many studies of surface and upper air frontogenesis (Ostdiek and Blumen [1995], Thorpe [1997]) and squall lines (Schubert et al. [1989]). A nice short overview of SG in terms of PV and the geostrophic coordinates is given in the tables of Schubert [1985]. Because of its common use, we shall shortly discuss the theory of geostrophic coordinates here.

The basic idea is that you can rewrite (using $v_a = v - v_g$) the zonal momentum equation (34a) as:

$$v_g = v - f^{-1}\frac{Du_g}{Dt} = \frac{Dy}{Dt} - \frac{Df^{-1}u_g}{Dt} = \frac{D}{Dt}\left(y - f^{-1}u_g\right) \tag{58}$$

Now introduce the meridional geostrophic coordinate as $\frac{DY}{Dt} = v_g$, i.e.:

$$Y \equiv y - f^{-1}u_g \tag{59}$$

Physically, this means that the geostrophic position of a particle Y is like a ghost position where the particle would be if it was only advected by geostrophic wind. Advection by the horizontal ageostrophic wind is thus *implicit in this coordinate system* making the EOM less nonlinear.

The geostrophic coordinate Y and the absolute momentum M_g are related through $M_g = -fY$ such that we can interpret Y as surfaces being parallel to M_g surfaces in real space. It was shown before that the absolute vorticity is given by $-\frac{\partial M_g}{\partial y}$. This means that the distance between two surfaces of the absolute momentum $|\nabla M_g|$, or equivalently $|\nabla Y|$, is a measure of the relative vorticity. This is shown in figure 14c where at the surface in the warm sector the relative vorticity is largest.

The advection by ageostrophic winds is not included in the geostrophic coordinates which means that the frontal slope in normal coordinates is replaced by a vertical front again in geostrophic coordinates, as shown in figure 14b. If we transform back to Cartesian coordinates, then the CFC is wrapped around the isolines of Y or absolute momentum and the CFC is more intense near the surface where relative vorticity is largest, i.e. a narrower and stronger updraft and more intense surface winds.

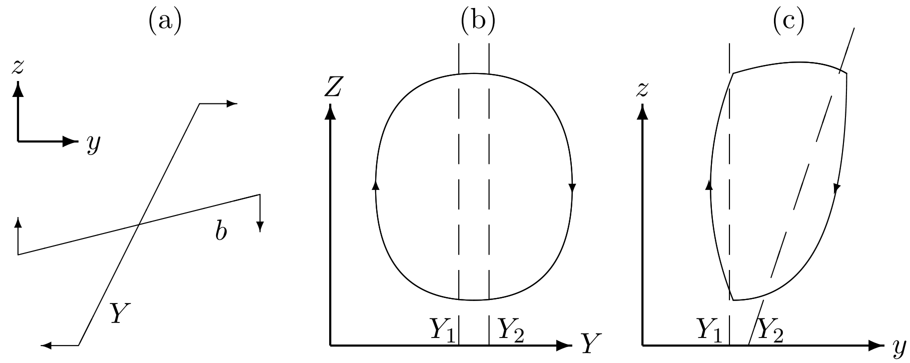


Figure 14 – (a) Ageostrophic motions (indicated by arrows) forming the thermal direct circulation which tends to restore TWB. The w_a gradient along b decreases $\frac{\partial b}{\partial y}$ and the u_a gradient along Y increases $\frac{\partial Y}{\partial z} \propto -\frac{\partial u_g}{\partial z}$. (b) The circulation around a closed contour in the YZ plane in a region where Q_2 is negative. (c) The same circulation in the xz plane. The dashed lines are the lines of constant Y which are close together near the surface indicating a region of large vorticity. Figure is based on [Hoskins, 1982, Fig.3]

Figure 14 shows how the CFC gets a simpler shape similar to the one in the QG case. We can derive the transformed SE equation by transforming (38) to geostrophic coordinates $(Y, Z, T) = (y - f^{-1}u_g, z, t)$. The partial derivatives can be transformed in the following way:

$$\frac{\partial}{\partial y} = \left.\frac{\partial Y}{\partial y}\right|_z \frac{\partial}{\partial Y} + \left.\frac{\partial Z}{\partial y}\right|_y \frac{\partial}{\partial Z} \tag{60a}$$

$$\frac{\partial}{\partial z} = \left.\frac{\partial Y}{\partial z}\right|_z \frac{\partial}{\partial Y} + \left.\frac{\partial Z}{\partial z}\right|_y \frac{\partial}{\partial Z} \tag{60b}$$

The four coefficients of the partial derivatives form the Jacobian matrix:

$$\begin{bmatrix} \frac{\partial Y}{\partial y} & \frac{\partial Y}{\partial z} \\ \frac{\partial Z}{\partial y} & \frac{\partial Z}{\partial z} \end{bmatrix} = \begin{bmatrix} 1 - f^{-1}\frac{\partial u_g}{\partial y} & -f^{-1}\frac{\partial u_g}{\partial z} \\ 0 & 1 \end{bmatrix} \tag{61}$$

The determinant of the Jacobian matrix is the Jacobian J given by:

$$J \equiv \frac{\partial Y}{\partial y}\frac{\partial Z}{\partial z} - \frac{\partial Y}{\partial z}\frac{\partial Z}{\partial y} = \frac{\partial Y}{\partial y} = 1 - f^{-1}\frac{\partial u_g}{\partial y} \tag{62}$$

such that (60) can be written as:

$$\frac{\partial}{\partial y} = J\frac{\partial}{\partial Y} \tag{63a}$$

$$\frac{\partial}{\partial z} = -f^{-1}\frac{\partial u_g}{\partial z}\frac{\partial}{\partial Y} + \frac{\partial}{\partial Z} \tag{63b}$$

Now we can substitute these derivatives in the Sawyer-Eliassen equation, resulting in:

$$\begin{aligned} N_{eff}^2 J\frac{\partial}{\partial Y}\left(J\frac{\partial\psi}{\partial Y}\right) + 2S^2 J\frac{\partial}{\partial Y}\left(-f^{-1}\frac{\partial u_g}{\partial z}\frac{\partial\psi}{\partial Y} + \frac{\partial\psi}{\partial Z}\right) + \\ F^2\left(-f^{-1}\frac{\partial u_g}{\partial z}\frac{\partial}{\partial Y} + \frac{\partial}{\partial Z}\right)\left(-f^{-1}\frac{\partial u_g}{\partial z}\frac{\partial\psi}{\partial Y} + \frac{\partial\psi}{\partial Z}\right) = 2Q_2 \end{aligned} \tag{64}$$

Now divide by J and write out the terms:

$$\begin{aligned} N_{eff}^2\frac{\partial}{\partial Y}\left(J\frac{\partial\psi}{\partial Y}\right) - 2S^2\frac{\partial}{\partial Y}\left(f^{-1}\frac{\partial u_g}{\partial z}\frac{\partial\psi}{\partial Y}\right) + 2S^2\frac{\partial^2\psi}{\partial Y\partial Z} + \\ \frac{F^2}{J}f^{-1}\frac{\partial u_g}{\partial z}\frac{\partial}{\partial Y}\left(f^{-1}\frac{\partial u_g}{\partial z}\frac{\partial\psi}{\partial Y}\right) - \frac{F^2}{J}f^{-1}\frac{\partial u_g}{\partial z}\frac{\partial^2\psi}{\partial Y\partial Z} - \\ \frac{F^2}{J}\frac{\partial}{\partial Z}\left(f^{-1}\frac{\partial u_g}{\partial z}\frac{\partial\psi}{\partial Y}\right) + \frac{F^2}{J}\frac{\partial^2\psi}{\partial Z^2} = \frac{2Q_2}{J} \end{aligned} \tag{65}$$

Use the relations $F^2/J = f^2$ and $S^2 = f\frac{\partial u_g}{\partial z}$ to rewrite:

$$\begin{aligned} N_{eff}^2\frac{\partial}{\partial Y}\left(F^2 f^{-2}\frac{\partial\psi}{\partial Y}\right) - (2S^2 - S^2)\frac{\partial}{\partial Y}\left(f^{-2}S^2\frac{\partial\psi}{\partial Y}\right) + \\ \frac{\partial^2\psi}{\partial Y\partial Z}\left(2S^2 - S^2\right) - f^2\frac{\partial}{\partial Z}\left(f^{-2}S^2\frac{\partial\psi}{\partial Y}\right) + f^2\frac{\partial^2\psi}{\partial Z^2} = \frac{2Q_2}{J} \end{aligned} \tag{66}$$

Next, we place the frequency parameters inside the derivatives:

$$\begin{aligned} \frac{\partial}{\partial Y}\left(N_{eff}^2 F^2 f^{-2}\frac{\partial\psi}{\partial Y}\right) - \frac{\partial N_{eff}^2}{\partial Y}F^2 f^{-2}\frac{\partial\psi}{\partial Y} - \frac{\partial}{\partial Y}\left(f^{-2}S^4\frac{\partial\psi}{\partial Y}\right) + \\ \frac{\partial S^2}{\partial Y}f^{-2}S^2\frac{\partial\psi}{\partial Y} + S^2\frac{\partial^2\psi}{\partial Y\partial Z} - S^2\frac{\partial^2\psi}{\partial Y\partial Z} - \frac{\partial S^2}{\partial Z}\frac{\partial\psi}{\partial Y} + f^2\frac{\partial^2\psi}{\partial Z^2} = \frac{2Q_2}{J} \end{aligned} \tag{67}$$

Here we see that the cross-derivative terms drop out. Moreover, the first and third term can be combined to:

$$\begin{aligned} f^{-2}\frac{\partial}{\partial Y}\left[\left(N_{eff}^2 F^2 - S^4\right)\frac{\partial\psi}{\partial Y}\right] + f^2\frac{\partial^2\psi}{\partial Z^2} \\ -\frac{\partial N_{eff}^2}{\partial Y}F^2 f^{-2}\frac{\partial\psi}{\partial Y} + \frac{\partial S^2}{\partial Y}f^{-2}S^2\frac{\partial\psi}{\partial Y} - \frac{\partial S^2}{\partial Z}\frac{\partial\psi}{\partial Y} = \frac{2Q_2}{J} \end{aligned} \tag{68}$$

$$
\begin{aligned}
f^{-2}\frac{\partial}{\partial Y}\left[\left(N_{eff}^2F^2 - S^4\right)\frac{\partial\psi}{\partial Y}\right] + f^2\frac{\partial^2\psi}{\partial Z^2}& \\
-\frac{\partial\psi}{\partial Y}J\frac{\partial N_{eff}^2}{\partial Y} - \frac{\partial\psi}{\partial Y}\left[-f^{-2}\left(f\frac{\partial u_g}{\partial z}\right)\frac{\partial}{\partial Y} + \frac{\partial}{\partial Z}\right]S^2 &= \frac{2Q_2}{J}
\end{aligned}
\tag{69}
$$

Use (63) to write the derivatives as:

$$
\begin{aligned}
f^{-2}\frac{\partial}{\partial Y}\left[\left(N_{eff}^2F^2 - S^4\right)\frac{\partial\psi}{\partial Y}\right] + f^2\frac{\partial^2\psi}{\partial Z^2}& \\
-\frac{\partial\psi}{\partial Y}\left(\frac{\partial N_{eff}^2}{\partial y} + \frac{\partial S^2}{\partial z}\right) &= \frac{2Q_2}{J}
\end{aligned}
\tag{70}
$$

Now use $N_{eff}^2 = N^2 + \frac{\partial b}{\partial z}$ and $S^2 = -\frac{\partial b}{\partial y}$:

$$
\begin{aligned}
f^{-2}\frac{\partial}{\partial Y}\left[\left(N_{eff}^2F^2 - S^4\right)\frac{\partial\psi}{\partial Y}\right] + f^2\frac{\partial^2\psi}{\partial Z^2}& \\
-\frac{\partial\psi}{\partial Y}\left(\frac{\partial^2 b}{\partial y\partial z} - \frac{\partial^2 b}{\partial z\partial y}\right) &= \frac{2Q_2}{J}
\end{aligned}
\tag{71}
$$

Using the definition of q_{SG} in (40), the final equation becomes:

$$
f^{-2}\frac{\partial}{\partial Y}\left[q_{SG}\frac{\partial\psi}{\partial Y}\right] + f^2\frac{\partial^2\psi}{\partial Z^2} = \frac{2Q_2}{J}
\tag{72}
$$

In geostrophic coordinates, the SE-equation above has the same form as the QG equation and is thus easier to solve in absence of the cross-derivative term.

q_{SG} itself is still given in Cartesian coordinates. We will now transform it also to geostrophic coordinates. We start by regrouping the terms in (50) as derivatives of b:

$$
\begin{aligned}
q_{SG} =& \frac{f}{N^2}\left[\frac{\partial b}{\partial z} - f^{-1}\frac{\partial u_g}{\partial y}\frac{\partial b}{\partial z} - f^{-2}\left(\frac{\partial b}{\partial y}\right)^2 - f^{-1}N^2\frac{\partial u_g}{\partial y}\right] \\
q_{SG} =& \frac{f}{N^2}\left[f^{-2}F^2\frac{\partial b}{\partial z} - f^{-2}\left(\frac{\partial b}{\partial y}\right)^2 - f^{-1}N^2\frac{\partial u_g}{\partial y}\right] \\
q_{SG} =& \frac{f}{N^2}\left[J\frac{\partial b}{\partial z} + f^{-1}\frac{\partial u_g}{\partial z}\frac{\partial b}{\partial y}\right] - \frac{\partial u_g}{\partial y}
\end{aligned}
\tag{73}
$$

where in the last step the replacement $f^{-2}F^2 = J$ is made. Following the line of Hoskins [1975], we can use the inverse transformation to simplify the first term. The inverse transformations are given by the inverse Jacobian matrix that can be derived from (61):

$$\begin{bmatrix} J & -f^{-1}\frac{\partial u_g}{\partial z} \\ 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} J^{-1} & 0 \\ f^{-1}\frac{\partial u_g}{\partial z}J^{-1} & 1 \end{bmatrix} \tag{74}$$

such that:

$$J\frac{\partial}{\partial Y} = \frac{\partial}{\partial y} \tag{75a}$$

$$J\frac{\partial}{\partial Z} = f^{-1}\frac{\partial u_g}{\partial z}\frac{\partial}{\partial y} + J\frac{\partial}{\partial z} \tag{75b}$$

Now compare (75b) with (73) and use this to simplify the first term of (73). Transform the second term too using (63a).

$$q_{SG} = J\left[\frac{f}{N^2}\frac{\partial b}{\partial Z} - \frac{\partial u_g}{\partial Y}\right] \tag{76}$$

The q_{SG} equation in GC is thus identical to the QG form in Cartesian coordinates apart from a Jacobian factor. Moreover, the expression has become linear again.

In a similar way one can check that q_{SG} is conserved too in geostrophic space, i.e. $\frac{Dq_{SG}}{DT} = 0$ where:

$$\frac{D}{DT} = \frac{\partial}{\partial T} + v_g\frac{\partial}{\partial Y} + w\frac{\partial}{\partial Z} \tag{77}$$

Horizontal advection by v_a is implicit as expected. w_a may be derived from the SE equation (72).

Now all ingredients are available for deriving the evolution of a front using semi-geostrophic theory but with the numerical ease of the quasi-geostrophic approach. Once all the fields are derived in geostrophic space, one has to return to physical space using inverse Jacobian transformations. This is not a trivial calculation. The whole (numerical) procedure is found in the appendix of Lu et al. [1997].

Part II
Model

7 Basic model description

In the second part of the thesis, we will explore the possibilities of both QG and SG theory in representing the effects of cross-frontal circulation in the context of a two-dimensional model. The reader can expect a detailed overview of several aspects of the model under which: (i) the input and the output, (ii) the algorithm in between, (iii) the numerical techniques that we used to solve linear and nonlinear equations and (iv) its constraints and possibilities. This part of the thesis will serve two goals:

Our primary goal is to answer the two research questions, which are:

1. What does the semi-geostrophic approach add to the quasi-geostrophic approach in the representation of a cross-frontal circulation and a tropopause fold?

2. Can we simulate a realistic tropopause folding event with a two-dimensional semi-geostrophic model based on stretching deformation?

For that, we will use the results of three model runs starting from simple to more complex configurations. We also analyzed an interesting case of a very deep tropopause fold over Europe. The data was taken from the ECMWF operational data archive, modified and put into the model in order to study this particular case.

Our secondary goal is to provide the reader the possibility to repeat the model runs and change parameters, e.g. in order to do a sensitivity analysis. A technical discussion on how the model works is therefore included. Moreover, the model is structured in a user-friendly way as much as possible. In the Appendix E, a small tutorial is given. The source code is also included in the Appendix F and some references to the code are found in the following chapters. The process of making the program user-friendly is a continuous process but currently the user may expect to be able to:

(i) change initial settings in the script

(ii) choose from command line: a configuration (initial jet/front system, initial PV anomaly, imported PV field), a balanced theory (QG or SG) and choose to include stratosphere component (y or n)

(iii) have insight in the algorithm during a run with errors given for the most common problems (for example: when negative PV values occur)

(iv) examine the output (from two data files) and visualize the data with several GNUplot scripts.

The GNUplot scripts are not given in the Appendix, but can be obtained by contacting the author.

7.1 Model properties

One can use the three conditions of Hoskins given on the first page of the introduction to get an idea what we need to get the program running. In short, this means that the model needs a formulation for (i) the reference atmosphere, (ii) a set of equations describing a balanced atmosphere and (iii) the boundary conditions. Only then can the PV-inversion be applied and other variables be solved globally in the model.

Firstly, we choose a 'constant-N' reference atmosphere to describe a linear increase of potential temperature θ with height in the troposphere. We choose a value of $N = 1.2 \times 10^{-2}$ s^{-1}. More information on our chosen reference atmosphere is found in the beginning of section 3.

In reality, the static stability increases beyond the tropopause up to values of around $4N$, corresponding to an isothermal reference atmosphere in the stratosphere. One now has to make an important decision on a realistic stratosphere model:

In older two-dimensional models (e.g. Hoskins and Bretherton [1972]), the tropopause was modeled as a discontinuous boundary between the low PV troposphere and the high PV stratosphere. This requires an additional dynamic boundary condition for the internal boundary.

It would make physically sense to use a height-dependent value $N(z)$, representing a reference atmosphere *at rest* throughout the troposphere and stratosphere, upon which (PV) perturbations are superpositioned related to the atmospheric dynamics. However, the numerical techniques are more complicated compared to a 'constant-N' formulation.

Here, we will do it in a different way. We interpret the additional stratification in the stratosphere as a PV perturbation upon the 'constant-N' reference atmosphere, defined as $q'(z)_{ts}$ where ts stands for the troposphere-stratosphere transition. Figure 15 shows the structure of q'_{ts} at t_0. This is done for numerical ease and it is a simpler but good way to represent the effect of the stratosphere on atmospheric dynamics. We model q'_{ts} using the following function:

$$q'_{ts}(z) = N^2 1.5 \left[1 + \mathrm{erf}\left(\frac{z - z_0}{\alpha} \right) \right] \tag{78}$$

where the erf function has a range from -1 to 1 centered at z_0 and $\alpha = 2$ km is the associated transition height scale.

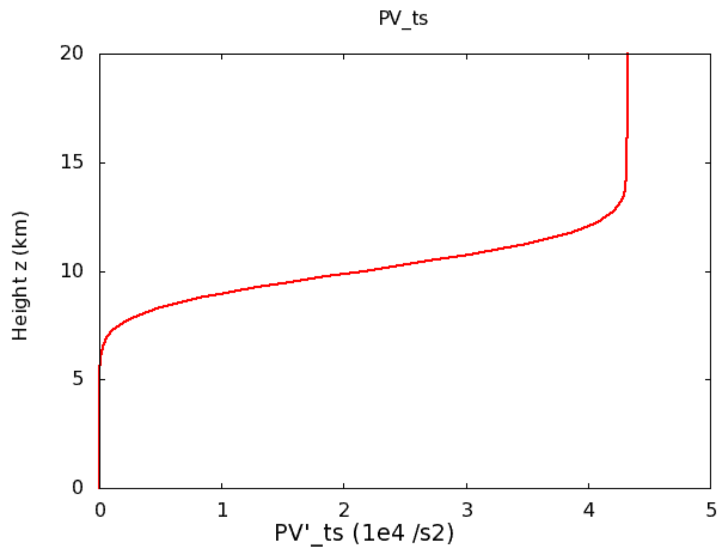


Figure 15 – q'_{ts} (y axis) as a function of height (z axis). q'_{ts} is a stratospheric anomaly representing additional static stability.

Secondly, upon the reference atmosphere the dynamic PV perturbation, or q'_{jet}, is placed. The balanced set of equations that we use are the quasi-geostrophic and semi-geostrophic ones. The initial configurations of q'_{jet} shall be discussed in the next subsection.

And finally the PV-inversion can be applied when proper boundary conditions are used. We choose a large domain with boundaries far away from the folding process in the center. Our two-dimensional domain is defined as:

$$y \in [-2500, 2500] \text{ km}$$

$$z \in [0, 50] \text{ km}$$

In most of the model illustrations, we will zoom in on the domain $y \in [-2000, 2000]$ km and $z \in [0, 20]$ km. The surface boundary is the only boundary that will have a large effect on the front evolution and therefore a *time-varying* boundary condition is required.

The domain contains 201×201 grid points including all boundaries. This corresponds to a horizontal resolution of 25 km and a vertical resolution of 0.25 km. Such a high resolution is necessary for a good representation of a tropopause fold with a typical horizontal scale of 100 km. The disadvantage is that especially for the SG PV-inversion, it can take several hours to complete an 1-day integration. A coarser grid of 101×101 grid points still gives qualitatively good results and is a good alternative, where 1-day integrations take no longer than half an hour.

Besides these three conditions, required for PV-inversion, we also include a deformation field and its associated frontogenetic forcing Q. We choose a constant stretching deformation $A = -\frac{\partial v_g}{\partial y} = 10^{-5}$ s^{-1} with the axis of dilatation positioned along $y = 0$. This value corresponds to an increase in the cross-frontal wind of 1 ms^{-1} each 100 km. Near the lateral boundaries, wind speeds are maximum at 25 ms^{-1}.

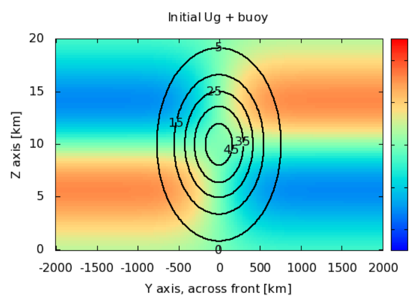
All values used in our three model runs are listed in table 3 for reference. These can be changed in the code lines 110 - 128.

Parameter	Name	Value
N	static stability	1.2×10^{-2} s^{-1}
f	Coriolis parameter	10^{-4} s^{-1}
A	stretching deformation	10^{-5} s^{-1}
θ_0	surface reference temperature	300 K
z_0	tropopause position	10 km
α	t.s. transition height scale	2 km
ygrid	# horizontal grid points	201
zgrid	# vertical grid points	201
domain_y	Length domain	5000 km
domain_z	Height domain	50 km
dt	time step	300 s
t_1	end time	86400 s (1 day)

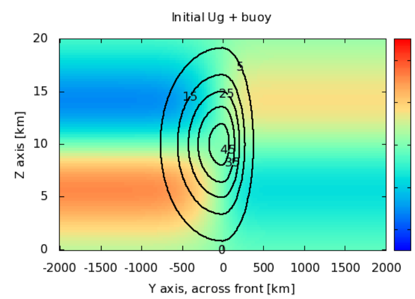
Table 3 – Value of parameters used in program.

7.2 Initial jet configurations

(a) Symmetric jet



(b) Complex jet



Parameter	**Value 'sj'**	**Value 'cj'**
$U_{jet,0}$ (ms^{-1})	50	50
y_0 (km)	0	0
z_0 (km)	10	10
$Y_{scale,1}$ (km)	500	250
$Y_{scale,2}$ (km)	500	500
Z_{scale} (km)	6	6

Table 4 – Illustrations and parameters for a symmetric jet ('sj') and a more complex jet ('cj'). Illustration: Jet velocities are shown as black contours (each 10 ms^{-1}). The buoyancy field is visualized using reddish colors for positive buoyancy and blueish colors for negative buoyancy values. Table: $U_{jet,0}$ is the maximum speed at position (y_0, z_0). $Y_{scale,1}$ is the horizontal length scale for $y > 0$. Similarly, $Y_{scale,2}$ is the horizontal length scale for $y < 0$. Z_{scale} is the vertical length scale.

We will use two initial jet configurations given in table 4. The equation for $U_g(y, z)$ is found in discretized form in the code lines 624 - 637 and is given by:

$$u_g(y,z) = \begin{cases} U_{jet,0}\, e^{-\left(\frac{y-y_0}{Y_{scale,1}}\right)^2} e^{-\left(\frac{z-z_0}{Z_{scale}}\right)^2} & \text{if } y > y_0 \\ U_{jet,0}\, e^{-\left(\frac{y-y_0}{Y_{scale,2}}\right)^2} e^{-\left(\frac{z-z_0}{Z_{scale}}\right)^2} & \text{if } y < y_0 \end{cases} \tag{79}$$

Such a wind field should be in thermal wind balance with the buoyancy field.

The initial buoyancy field is calculated analytically from TWB, $\frac{\partial b}{\partial y} = -f\frac{\partial u_g}{\partial z}$:

$$b(y,z) = \begin{cases} \frac{\sqrt{\pi} f U_{jet,0} Y_{scale,1}}{Z_{scale}^2}(z-z_0)\mathrm{erf}\left(\frac{y-y_0}{Y_{scale,1}}\right) e^{-\left(\frac{z-z_0}{Z_{scale}}\right)^2} \text{ if } y > y_0 \\ \\ \frac{\sqrt{\pi} f U_{jet,0} Y_{scale,2}}{Z_{scale}^2}(z-z_0)\mathrm{erf}\left(\frac{y-y_0}{Y_{scale,2}}\right) e^{-\left(\frac{z-z_0}{Z_{scale}}\right)^2} \text{ if } y < y_0 \end{cases} \quad (80)$$

where the erf function appears due to integration with respect to y. It is found in discretized form in the code lines 639 - 655. The buoyancy difference between the lateral boundaries is approximately 0.5 ms^{-2} which corresponds to a (potential) temperature difference of 15 K.

These two jet configurations presented here are chosen for the following reason:

'sj' With this symmetric configuration we wish to do a 1-day integration with both (i) QG theory and (ii) SG theory in order to study the differences in the evolution of the tropopause.

'cj' In this case, only a 1-day model run with SG theory shall be shown and be compared with the previous jet structure. This structure is characterized by negative vorticity on the left side of the jet ($-\frac{\partial u_g}{\partial y} < 0$), but now has a higher positive vorticity on the right side of the jet ($-\frac{\partial u_g}{\partial y} > 0$).

This brings us to a total of three model runs of which the results shall be discussed in chapter 9. Their names are given in table 5.

`QGsymjet`	`SGsymjet`	`SGcomjet`

Table 5 – Names of the three model runs

Our choices for $Y_{scale,1}$, $Y_{scale,2}$ and Z_{scale} determine the initial downward extension of the dynamic tropopause. When making choices for these initial parameters, one should keep in mind:

Z_{scale}: the vertical extent of the jet is proportional to the vertical extent of the fold

Y_{scale}: the horizontal extent of the jet is inversely proportional to the strength of the fold

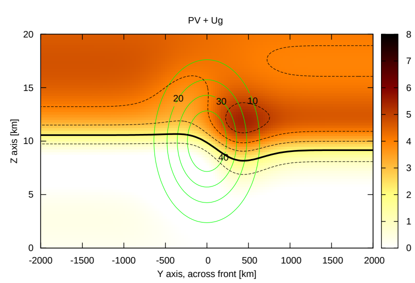
The program requires the input of an initial PV field. The next step is thus to calculate the PV anomaly associated with the jet structures, q'_{jet}, and to include the stratospheric PV anomaly, q'_{ts}. The term q'_{jet} is calculated from the definitions of q_{QG} or q_{SG} depending on the model run, and are given by the equations (81a) and (81b). It is also shown in the code lines 712 - 761.

$$q_{QG} = \frac{f}{N^2}\frac{\partial b}{\partial z} - \frac{\partial u_g}{\partial y} \tag{81a}$$

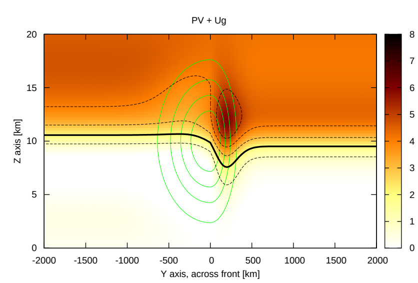
$$q_{SG} = \frac{f}{N^2}\frac{\partial b}{\partial z} - \frac{\partial u_g}{\partial y} - \frac{1}{N^2}\frac{\partial u_g}{\partial y}\frac{\partial b}{\partial z} - \frac{1}{fN^2}\left(\frac{\partial b}{\partial y}\right)^2 \tag{81b}$$

The total PV field (i.e. the sum of both PV anomalies) for the SG case is shown in figure 16. It does not differ much from the q_{QG} structure, i.e. the nonlinear contribution is small (illustrated by the two upper panels of figure 24(a)).

Both panels show that the dynamic tropopause is located above 10 km on the left side and beneath 10 km on the right side. It is related to a front in the troposphere and a reversed front in the stratosphere as shown by the figures in table 4. Around 10 km, the vertical buoyancy gradient (first term in (81b)) is negative on the left side and positive on the right side resulting in negative and positive contributions. Moreover, the second term in (81b) is vorticity which is positive just right of the jet such that the tropopause is even lower there, forming the beginning of a developing tropopause fold.



(a) Symmetric jet



(b) Complex jet

Figure 16 – The same jet configurations as in table 4 with the associated q_{SG} fields visualized by color-shading with a color scale given on the right. The dynamic tropopause is indicated by the thick black line having a PV value of 2×10^{-4} s^{-1}.

Now that the initial PV fields are prepared, a numerical model is used to calculate its future development. An overview on the procedure and the applied numerical techniques shall be given in the next section.

8 Numerical techniques

8.1 Overview of algorithm

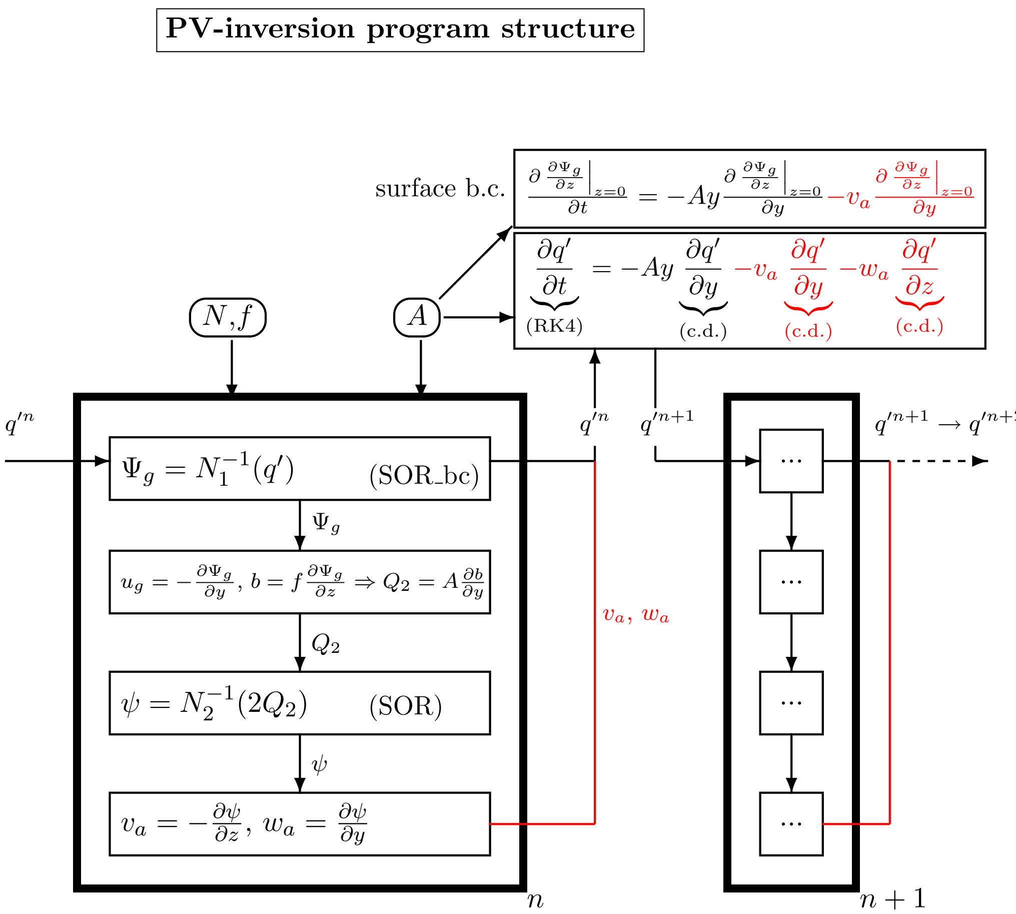


Figure 17 – Schematic overview of the program algorithm. Illustration is similar to figure 13. It is now given for a general PV perturbation field q' at timestep n. q' can be the quasi-geostrophic or semi-geostrophic PV field. N_1 is the PV-inversion operator and N_2 the Sawyer-Eliassen operator. Their forms are shown in figure 10 resp. figure 13 for the QG resp. SG case. The red terms in the advection equations are zero in the QG case and nonzero in the SG case. Finally, the numerical procedures SOR(_bc), RK4 and c.d. are abbreviations for Successive Over-Relaxation (including or excluding relaxation on boundaries), Runge-Kutta 4th order and centered differences.

The five numerical steps in the program are schematically displayed in figure 17). Given an initial PV perturbation field, q'^n, the first four steps (within the thick black box) are from top to bottom: (i) solving the PV-inversion, (ii) calculating the frontogenetic forcing, (iii) solving the Sawyer-Eliassen equation and (iv) calculating the ageostrophic velocity components. Once all variables are determined diagnostically, the dynamic system evolves by (v) calculating the advection of q'. After that, we calculate all other variables diagnostically again from q'^{n+1}. We will now go through the five steps in more detail.

(i) The PV-inversion relations were derived for QG and SG theory and are given by (33) and (52). The SGPV-inversion contains two additional *nonlinear* terms such that the operator N_1 is nonlinear too. The numerical approach therefore involves an additional iteration loop to evaluate the nonlinear terms as we shall see in the next subsection.

For now, let us focus on the QG case. To solve the linear elliptical equation we use Successive Over-Relaxation (hereafter SOR). We use a modified version (named SOR_bc) to incorporate boundaries conditions in the iteration process.

The reason for that is because a good representation of $\frac{\partial \Psi_g}{\partial z} = b/f$ (i.e. buoyancy) on the surface boundary is needed for realistic thermal gradients *above* the surface boundary. The same holds for the lateral boundary conditions. Without the condition, $\frac{\partial \Psi_g}{\partial y} = -u_g$, the geostrophic velocity would be required to go to zero there.

(ii) Once the geostrophic streamfunction Ψ_g has been derived, the next step is to derive u_g and b which are in TWB by definition. For our two jet configurations, we checked that the resulting u_g field is the same as defined initially in (79). From the buoyancy field and the given geostrophic deformation field, the frontogenetic forcing Q_2 is calculated.

(iii) & (iv) The second operator N_2 corresponds to the Sawyer-Eliassen equation. It is linear for both QG and SG theory as given by the equations (28) and (38). In the SG however, the stability parameters N_{eff}^2, F^2 and S^2 are neither zero nor constant and have to be determined a priori. We use SOR again to derive the ageostrophic circulation here. Unlike SOR_bc, we set $\psi = 0$ on all boundaries and relax on the whole domain *excluding* the boundaries.

One important aspect of the operator N_2 in the SG case is that it is only elliptical and solvable when the condition given in (39) is met. Thus, one should be aware not to choose initial conditions that are 'too extreme' (e.g. a strong and very local jet, $U_{jet,0} = 100$ ms^{-1} and $Y_{scale,2} = 50$ km) which correspond to symmetric instability or, equivalently, negative PV values.

(v) Once the initial PV field is positive on the whole domain, it *remains* positive and symmetric instability will not occur. It is the result of PV conservation, given for QG and SG case by equations (32) and (50).

In the QG case, only the large scale deformation field advects PV and the system of equations becomes decoupled. If there would be an error in the Sawyer-Eliassen equation for example, it would not affect the evolution of the PV field. In the SG case however, v_a and w_a are included in the advection (see red lines in figure 17).

For the local time derivative, we use Runge-Kutta 4th order (hereafter RK4) method and for the spatial derivatives, we use centered differences. RK4 is an accurate and still a reasonably quick method for calculating new PV fields on the whole domain. The RK4 method is also applied to the surface boundary to evaluate the new surface buoyancy, or $\frac{\partial \Psi_g}{\partial z}$, at every timestep. This is important. If the surface boundary condition was held constant in time than that would lead to large differences between the thermal gradients on and above the surface boundary.

Once the new potential vorticity is evaluated, the algorithm repeats itself. This goes on until the final time t_1 has been reached which is one day in our model runs. With the timestep being set on 300 s, this corresponds to a total of 288 loops.

In the next subsections, we discuss both SOR and RK4 techniques in more detail.

8.2 Successive over-relaxation

In the program we apply the SOR subroutine in various ways. We use two versions of SOR: (i) One SOR subroutine with constant boundary conditions on which the solution is set to zero (called SOR4), given in the code lines 1127 - 1173 and (ii) a more complex one where the boundaries are part of the routine (called SOR5), given in the code lines 1177 - 1244. SOR4 shall be used for solving the linear Sawyer-Eliassen equation in QG and SG form. SOR5 for solving the linear QGPV-inversion relation and the nonlinear SGPV-inversion. Thus, we use the SOR method for four different situations in total. We start by explaining the most basic form and add the other elements step by step.

The simplest elliptical equation is the QG form of the Sawyer-Eliassen equation:

$$N^2\psi_{yy} + f^2\psi_{zz} = N_2(\psi) = 2Q_2 \tag{82}$$

where N_2 is an elliptic operator on ψ.

We can interpret this equation as a final equilibrium state of the following partial differential equation:

$$\frac{\partial \psi}{\partial t} = N_2(\psi) - 2Q_2 \tag{83}$$

In other words, an initial well-chosen solution *relaxes* towards the equilibrium state ($\frac{\partial \psi}{\partial t} = 0$) given in (82) for $t \to \infty$.

SOR is a good example of a numerical relaxation routine. It uses an overcorrection ω, implanted in such a way that we anticipate on future corrections towards equilibrium. This results in faster convergence. The theory behind SOR is based on chapter 19.5 of Press et al. [2005] and explained in the Appendix C. This overcorrection is shown in discretized form as follows:

$$\psi_{j,l}^{(r)} = \psi_{j,l}^{(r-1)} - \omega f(\psi_{j,l}^{(r-1)}) \tag{84}$$

where $f(\psi^{(r-1)})$ consists of both the operator $N_2(\psi)$ and the forcing $2Q_2$.

It is shown in Appendix C, that we can rewrite (84) in a practical form for numerical purposes, namely:

$$\psi_{j,l}^{(r)} = \psi_{j,l}^{(r-1)} - \omega \frac{\xi_{j,l}}{e_{j,l}} \tag{85}$$

where $\xi_{j,l}$ is the so-called residual term. It is similar to the discretized form of the rhs of (83):

$$\xi_{j,l} = a_{j,l}\psi_{j+1,l} + b_{j,l}\psi_{j-1,l} + c_{j,l}\psi_{j,l+1} + d_{j,l}\psi_{j,l-1} + e_{j,l}\psi_{j,l} - f_{j,l} \tag{86}$$

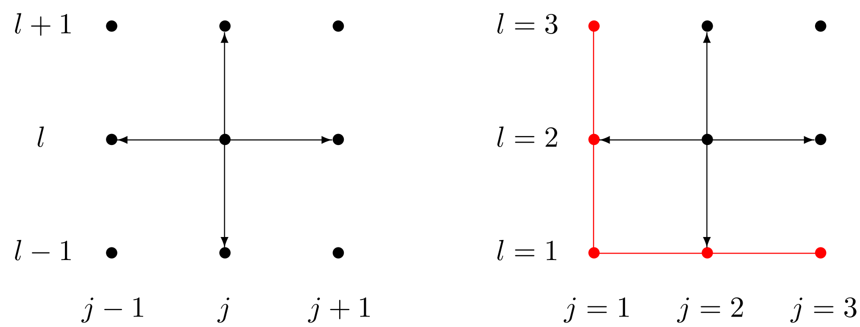
and we see here that the denominator in (85), $e_{j,l}$, is the coefficient of $\psi_{j,l}$.

The coefficients a, b, c, d, e, f have to be derived in order to calculate the residual vector and subsequently the value of $\psi_{j,l}^{(r)}$. We will derive these now for each of the four elliptical equations.

8.2.1 SOR routines

QG Sawyer-Eliassen

$$N^2\frac{\partial^2\psi}{\partial y^2}+f^2\frac{\partial^2\psi}{\partial z^2}=2Q_2$$



(a) Central domain (b) Near corner

Figure 18 – Simplest version of SOR: (a) evaluation of $\psi_{j,l}$ using information from *five* nearby points. (b) no evaluation of $\psi_{1,1}$, $\psi_{1,2}, \ldots$ on boundaries (red line) where $\psi=0$

It is illustrated in figure 18 that we only applied the SOR algorithm on the central domain and not on the boundaries. We applied Dirichlet boundary conditions, i.e. $\psi=0$ on all boundaries.

Now let us take a grid point (j,l) and discretize the QG Sawyer-Eliassen equation in two steps:

$$\frac{N^2}{\Delta y}\frac{\partial}{\partial y}\left(\psi_{j+0.5,l}-\psi_{j-0.5,l}\right)+\frac{f^2}{\Delta z}\frac{\partial}{\partial z}\left(\psi_{j,l+0.5}-\psi_{j,l-0.5}\right)=2Q_{2(j,l)} \tag{87}$$

$$\frac{N^2}{\Delta y^2}(\psi_{j+1,l}-2\psi_{j,l}+\psi_{j-1,l})+\frac{f^2}{\Delta z^2}(\psi_{j,l+1}-2\psi_{j,l}+\psi_{j,l-1})=2Q_{2(j,l)} \tag{88}$$

We can reorder them in the following way as a linear combination and multiply by Δy^2:

$$N^2\psi_{j+1,l} + N^2\psi_{j-1,l} + f^2\beta^2\psi_{j,l+1} + f^2\beta^2\psi_{j,l-1} - 2(N^2 + f^2\beta^2)\psi_{j,l} \\ = 2\Delta y^2 Q_{2(j,l)} \tag{89}$$

where $\beta = \frac{\Delta y}{\Delta z}$.

It is now possible to determine the coefficients a, b, c, d, e, f given that:

$$a_{j,l}\psi_{j+1,l} + b_{j,l}\psi_{j-1,l} + c_{j,l}\psi_{j,l+1} + d_{j,l}\psi_{j,l-1} + e_{j,l}\psi_{j,l} = f_{j,l} \tag{90}$$

such that the coefficients $a_{j,l}, ..., f_{j,l}$ are given by:

$$a_{j,l} = N^2 \tag{91a}$$

$$b_{j,l} = N^2 \tag{91b}$$

$$c_{j,l} = f^2\beta^2 \tag{91c}$$

$$d_{j,l} = f^2\beta^2 \tag{91d}$$

$$e_{j,l} = -2(N^2 + f^2\beta^2) \tag{91e}$$

$$f_{j,l} = 2\Delta y^2 Q_{2(j,l)} \tag{91f}$$

In the quasi-geostrophic case, these are constant in time except for the forcing $Q_{2(j,l)}$.

The residual vector $\xi_{j,l}$ in (86) can now be calculated. In the procedure, $\xi_{j,l}$ will eventually go to zero. The coefficients a, b, c, d, e and the corresponding solution for ψ will converge to the given forcing $f_{j,l}$. The iteration process shall be terminated when the norm of ξ, e.g. $||\xi||$, becomes smaller than $10^{-3}||f||$. Or in other words, when the *relative error* (ξ includes $-f$ term) becomes smaller than 10^{-3}:

$$\frac{||\xi||}{||f||} < 10^{-3} \tag{92}$$

We now summarize the SOR procedure here for the QG Sawyer-Eliassen equation:

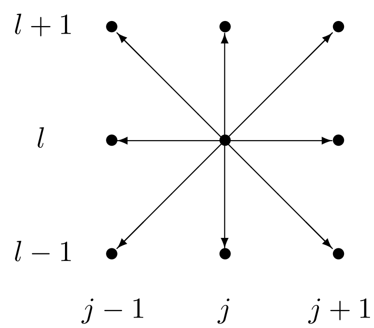
1. Derive the coefficients a, b, c, d, e, f of the elliptical equation (see code lines 529 - 550). Also calculate these on the boundaries where $\psi = 0$.

2. Insert a first guess for the ψ field, take ψ values from previous time step or use initial zero values.

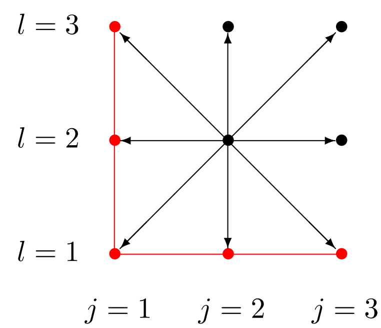
3. Given these input fields the relaxation process can start. The SOR4 subroutine is called in code lines 1127 - 1173.
4. In the SOR4 subroutine, the norm of the forcing term f is derived first.
5. Start the iteration, derive new ψ fields from equation (85) every iteration step until $||\xi||$ becomes smaller than $10^{-3}||f||$

SG Sawyer-Eliassen

$$N_{eff}^2 \frac{\partial^2 \psi}{\partial y^2} - 2S^2 \frac{\partial^2 \psi}{\partial y \partial z} + F^2 \frac{\partial^2 \psi}{\partial z^2} = 2Q_2$$



(a) Central domain



(b) Near corner

Figure 19 – SOR version including discretization of cross-derivative terms: (a) evaluation of $\psi_{j,l}$ using information from *nine* nearby points. (b) no evaluation of $\psi_{1,1}$, $\psi_{1,2}, \ldots$ on boundaries (red line) where $\psi = 0$

We follow the same procedure as in the QG case by deriving the coefficients a, b, c, d, e, f for SOR4. The Sawyer-Eliassen equation is discretized in the following two steps:

$$\frac{N_{eff(j,l)}^2}{\Delta y} \frac{\partial}{\partial y} \left(\psi_{j+0.5,l} - \psi_{j-0.5,l}\right) - 2\frac{S_{j,l}^2}{\Delta z} \frac{\partial}{\partial y} \left(\psi_{j,l+0.5} - \psi_{j,l-0.5}\right) + \frac{F_{j,l}^2}{\Delta z} \frac{\partial}{\partial z} \left(\psi_{j,l+0.5} - \psi_{j,l-0.5}\right) = 2Q_{2(j,l)} \tag{93}$$

$$
\begin{aligned}
\frac{N^2_{eff(j,l)}}{\Delta y^2}\left(\psi_{j+1,l} - 2\psi_{j,l} + \psi_{j-1,l}\right) - \\
2\frac{S^2_{j,l}}{\Delta y \Delta z}\left(\psi_{j+0.5,l+0.5} - \psi_{j-0.5,l+0.5} - \psi_{j+0.5,l-0.5} + \psi_{j-0.5,l-0.5}\right) + \\
\frac{F^2_{j,l}}{\Delta z^2}\left(\psi_{j,l+1} - 2\psi_{j,l} + \psi_{j,l-1}\right) = 2Q_{2(j,l)}
\end{aligned} \tag{94}
$$

Compared to the QG case, we see two new elements here, namely: (i) the coefficients (containing N_{eff}, F^2, S^2) are no longer constant, but vary in space and (ii) the discretized form of the cross-derivative term requires information from 4 additional grid points ($\times$-shaped) as shown in figure 19.

However, no information is available halfway between the grid points. Therefore, we extend the $\times$-shape and use information from $\psi_{j+1,l+1}$, $\psi_{j-1,l+1}$, $\psi_{j-1,l-1}$ and $\psi_{j+1,l-1}$ instead such that (94) becomes:

$$
\begin{aligned}
\frac{N^2_{eff(j,l)}}{\Delta y^2}\left(\psi_{j+1,l} - 2\psi_{j,l} + \psi_{j-1,l}\right) - \\
2\frac{S^2_{j,l}}{4\Delta y \Delta z}\left(\psi_{j+1,l+1} - \psi_{j-1,l+1} - \psi_{j+1,l-1} + \psi_{j-1,l-1}\right) + \\
\frac{F^2_{j,l}}{\Delta z^2}\left(\psi_{j,l+1} - 2\psi_{j,l} + \psi_{j,l-1}\right) = 2Q_{2(j,l)}
\end{aligned} \tag{95}
$$

and multiply by Δy^2:

$$
\begin{aligned}
N^2_{eff(j,l)}\psi_{j+1,l} + N^2_{eff(j,l)}\psi_{j-1,l} - \\
\frac{1}{2}S^2_{j,l}\beta\left(\psi_{j+1,l+1} - \psi_{j-1,l+1} - \psi_{j+1,l-1} + \psi_{j-1,l-1}\right) + \\
F^2_{j,l}\beta^2\psi_{j,l+1} + F^2_{j,l}\beta^2\psi_{j,l-1} - 2(N^2_{eff(j,l)} + F^2_{j,l}\beta^2)\psi_{j,l} = 2\Delta y^2 Q_{2(j,l)}
\end{aligned} \tag{96}
$$

And finally, the coefficients for the SOR4 subroutine become:

$$a_{j,l} = N^2_{eff(j,l)} \tag{97a}$$

$$b_{j,l} = N^2_{eff(j,l)} \tag{97b}$$

$$c_{j,l} = F^2_{j,l}\beta^2 \tag{97c}$$

$$d_{j,l} = F^2_{j,l}\beta^2 \tag{97d}$$

$$e_{j,l} = -2(N^2_{eff(j,l)} + F^2_{j,l}\beta^2) \tag{97e}$$

$$f_{j,l} = 2\Delta y^2 Q_{2(j,l)} \tag{97f}$$

$$g_{j,l} = -\frac{1}{2} S_{j,l}^2 \beta \tag{97g}$$

where we introduce a new coefficient $g_{j,l}$ of the variables $\psi_{j+1,l+1}$, $-\psi_{j-1,l+1}$, $-\psi_{j+1,l-1}$ and $\psi_{j-1,l-1}$, forming the discretized form of the cross-derivative term.

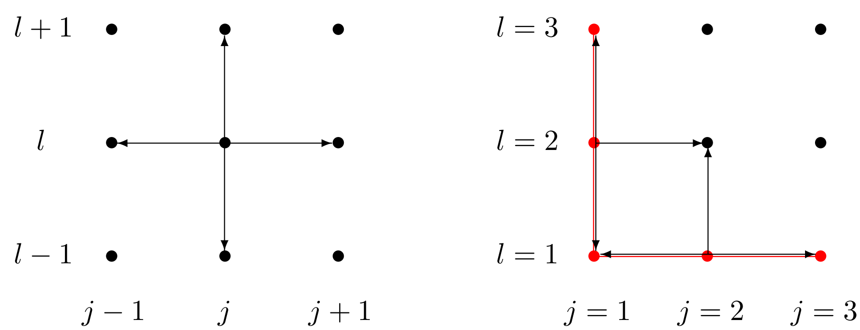
The same steps in the iteration procedure summarized in QG case can be taken where we should note that the residual vector is now given by:

$$\begin{aligned} \xi_{j,l} = a_{j,l}\psi_{j+1,l} + b_{j,l}\psi_{j-1,l} + c_{j,l}\psi_{j,l+1} + d_{j,l}\psi_{j,l-1} + e_{j,l}\psi_{j,l} - f_{j,l} \\ +g_{j,l}(\psi_{j+1,l+1} - \psi_{j+1,l-1} - \psi_{j-1,l+1} + \psi_{j-1,l-1}) \end{aligned} \tag{98}$$

If we set $g_{j,l}$ to zero and take constant values N and f for N_{eff} and F^2 we end up with the QG case again. Therefore, both QG and SG forms of the Sawyer-Eliassen equation use the same SOR4 subroutine.

QGPV-inversion

$$\frac{f^2}{N^2}\frac{\partial^2 \Psi_g}{\partial z^2} + \frac{\partial^2 \Psi_g}{\partial y^2} = q_{QG}$$



(a) Central domain

(b) Near corner

Figure 20 – SOR version including evaluation on boundaries: (a) evaluation of $\psi_{j,l}$ using information from *five* nearby points. (b) grid points $\psi_{1,1}$, $\psi_{1,2}$, ... on boundaries (red line) are included in SOR iteration process.

The elliptical equation that has to be solved for QGPV-inversion shows large similarities with the QG Sawyer-Eliassen equation. The discretization of the QGPV-inversion is performed in the same way (with almost similar coefficients) and will not be shown here. The only additional numerical problem is the evaluation at the boundaries which is crucial for performing the PV-inversion. Therefore, we shall discretize the QGPV-inversion equation for two grid points on the boundary and solve for $\psi_{1,2}$ and $\psi_{2,1}$ illustrated in figure 20.

We start with the discretization in two steps for $\psi_{1,2}$:

$$\frac{f^2}{N^2\Delta z}\frac{\partial}{\partial z}\left(\psi_{j,l+0.5}-\psi_{j,l-0.5}\right)+\frac{1}{0.5\Delta y}\frac{\partial}{\partial y}\left(\psi_{j+0.5,l}-\psi_{j,l}\right)=q_{QG(j,l)} \tag{99}$$

$$\begin{aligned}\frac{f^2}{N^2\Delta z^2}\left(\psi_{j,l+1}\right.&\left.-2\psi_{j,l}+\psi_{j,l-1}\right)+\\ &\frac{1}{0.5\Delta y^2}\left(\psi_{j+1,l}-\psi_{j,l}-\Delta y\left.\frac{\partial\psi_{j,l}}{\partial y}\right|_{j=1}\right)=q_{QG(j,l)}\end{aligned} \tag{100}$$

We see that additional information is needed on the lateral boundary, namely $\frac{\partial\psi_{j,l}}{\partial y}|_{j=1}=-u_g$. This may be set to zero for some jet configurations. But when the boundary's influence is large (e.g. smaller domain), one should use an appropriate value for u_g at the lateral boundary at all times.

The importance of the surface boundary, on the other hand, is large at all times. We can derive the discretized form at $\psi_{2,1}$ in a similar way:

$$\begin{aligned}\frac{f^2}{0.5N^2\Delta z^2}\left(\psi_{j,l+1}-\psi_{j,l}\right.&\left.-\Delta z\left.\frac{\partial\psi_{j,l}}{\partial z}\right|_{l=1}\right)+\\ &\frac{1}{\Delta y^2}\left(\psi_{j+1,l}-2\psi_{j,l}+\psi_{j-1,l}\right)=q_{QG(j,l)}\end{aligned} \tag{101}$$

The input of $\frac{\partial\psi_{j,l}}{\partial z}|_{l=1}=b/f$ is required this time, i.e. the input of buoyancy values on the surface. From observations, one could import the 2 meter temperature for example.

The boundary constraints are now set and we can calculate the coefficients a,b,c,d,e,f again on the surface boundary. First, we multiple (101) by $N^2\Delta y$ and reorder afterward:

$$\begin{aligned}2f^2\beta^2\left(\psi_{j,l+1}-\psi_{j,l}-\Delta z\left.\frac{\partial\psi_{j,l}}{\partial z}\right|_{l=1}\right)+&\\ N^2\left(\psi_{j+1,l}-2\psi_{j,l}+\psi_{j-1,l}\right)&=\Delta y^2N^2q_{QG(j,l)}\end{aligned} \tag{102}$$

$$N^2\psi_{j+1,l} + N^2\psi_{j-1,l} + 2f^2\beta^2\psi_{j,l+1} - 2(N^2 + f^2\beta^2) = \Delta y^2 N^2 q_{QG(j,l)} + 2f^2\beta^2\Delta z \left.\frac{\partial\psi_{j,l}}{\partial z}\right|_{l=1} \tag{103}$$

The coefficients on the surface boundary are given by:

$$a_{j,1} = N^2 \tag{104a}$$

$$b_{j,1} = N^2 \tag{104b}$$

$$c_{j,1} = 2f^2\beta^2 \tag{104c}$$

$$d_{j,1} = 0 \tag{104d}$$

$$e_{j,1} = -2(N^2 + f^2\beta^2) \tag{104e}$$

$$f_{j,1} = \Delta y^2 N^2 q_{QG(j,1)} + 2f^2\beta^2\Delta z \left.\frac{\partial\psi_{j,l}}{\partial z}\right|_{l=1} \tag{104f}$$

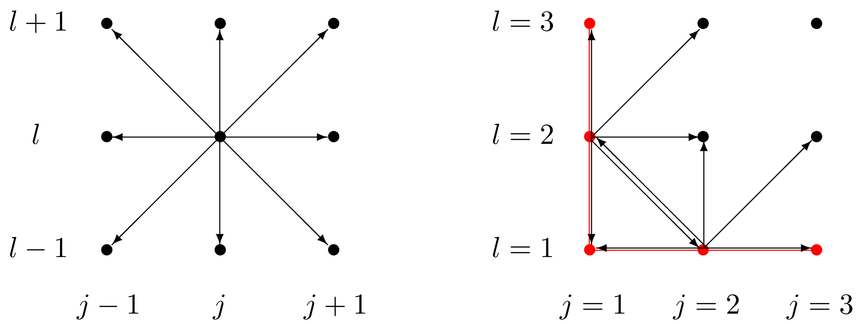
$$g_{j,1} = 0 \tag{104g}$$

Coefficient $d_{j,1}$ corresponds to a grid point below the surface boundary and is zero as one would expect. Also, a factor 2 has been included in $c_{j,1}$. Finally, the surface buoyancy can be interpreted as additional forcing along with PV.

These coefficients can also be calculated for other boundaries (and in the corners) and these are given in the code lines 374 - 445. After that, the SOR5 subroutine is called which is given in code lines 1177 - 1244. SOR5 includes the boundaries in the iteration process.

SGPV-inversion

$$\frac{f^2}{N^2}\frac{\partial^2\Psi_g}{\partial z^2} + \frac{\partial^2\Psi_g}{\partial y^2} + \frac{f}{N^2}\frac{\partial^2\Psi_g}{\partial y^2}\frac{\partial^2\Psi_g}{\partial z^2} - \frac{f}{N^2}\left(\frac{\partial^2\Psi_g}{\partial y\partial z}\right)^2 = q_{SG}$$



(a) Central domain (b) Near corner

Figure 21 – SOR version including evaluation on boundaries and cross-derivative terms: (a) evaluation of $\psi_{j,l}$ using information from *nine* nearby points. (b) grid points $\psi_{1,1}$, $\psi_{1,2}$, ... on boundaries (red line) are included in SOR iteration process.

The SGPV-inversion equation is the most complex elliptical equation to solve with SOR because it contains two additional nonlinear terms. One of them contains a cross-derivative and we know from our discussion above that we need information on ϕ from the center + 8 surrounding grid points. Moreover, this means that at the boundaries information from 6 grid points is needed as shown in figure 21.

The SGPV-inversion equation contains two nonlinear terms. Therefore, *an additional iteration process is required to evaluate the nonlinear terms.* This is illustrated by the thick black box in figure 22.

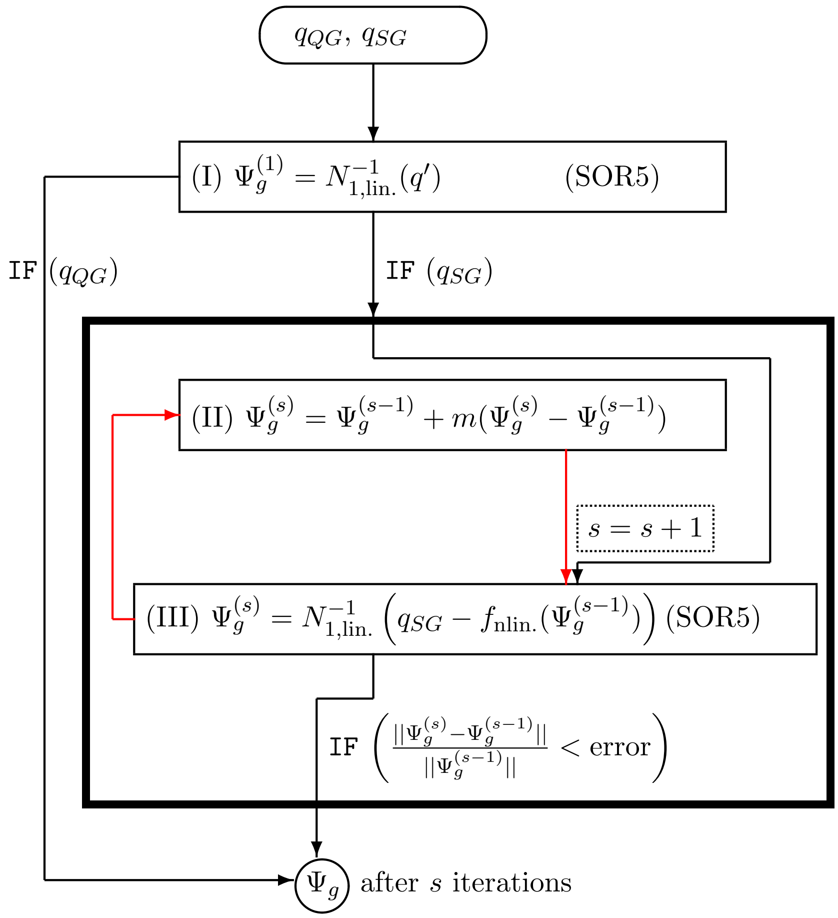


Figure 22 – Subroutine SOR5 is only called once for QGPV-inversion to estimate the final Ψ_g. For the nonlinear SGPV-inversion, the SOR procedure is called s times in an additional iteration loop in order to derive the nonlinear terms. This is illustrated inside the thick black box where the red arrows visualize the additional iteration loop. $f_{\text{nlin.}}(\Psi_g^{(s-1)})$ is a function of the nonlinear terms which represents additional 'forcing'. The procedure inside the thick box is based on the method discussed in Verkley [2001]

The procedure shall now be discussed in more detail. For SGPV-inversion, the following steps are taken:

1. In box (I) of figure 22, we insert the q_{SG} field and initially set the nonlinear terms to zero. The operator N_1 is therefore linear and is similar to the QGPV-inversion. The coefficients a, b, c, d, e, f, g can be derived and inserted into the SOR5 subroutine to calculate the initial $\Psi_g^{(1)}$ field.

2. We now start the additional iteration loop inside the thick black box corresponding to the subroutine described in the code lines 308 - 372. The initial $\Psi_g^{(1)}$ field provides a first guess field for $\Psi_g^{(s-1)}$ in the nonlinear terms in (III). The two nonlinear terms are now on the rhs and may be interpreted as additional forcing terms. Now we can calculate the new $\Psi_g^{(2)}$ field again by using the SOR5 subroutine.

3. The next step is crucial. If the nonlinear terms are too large compared with q_{SG}, $\Psi_g^{(2)}$ will differ significantly from the initial $\Psi_g^{(1)}$ field. Therefore, we use a correction parameter m in order to reduce the influence of the $\Psi_g^{(2)} - \Psi_g^{(1)}$ difference. If $m = 1$, no correction on $\Psi_g^{(2)}$ is done. If $m = 0$, Ψ_g will not change at all by 'nonlinear forcing'. By testing, we have chosen $m = 0.8$ as the optimal value where Ψ_g converges most quickly to the true Ψ_g field.

4. After the m correction, we go to the next iteration step $s = s + 1$ with the corrected 'nonlinear forcing'. The iteration loop continues until the relative error in $\Psi_g^{(s)}$ becomes smaller than the relative error specified by the user. We use $err = 10^{-4}$.

The whole nonlinear process can be summarized to $\Psi_g = N_{1,SG}^{-1}(q'_{SG})$ as given in figure 17, where $N_{1,SG}$ is the nonlinear operator in the SG case.

It is not completely understood under what conditions convergence occurs or not. In general, negative values for $q_{SG} + f$ results in exponential growth instead of a solution for Ψ_g. We saw in paragraph 5.2.1 that $q_{SG} + f$ corresponds to symmetric instability apart from a constant term. Thus, the SGPV-inversion does not work in general when areas of symmetric instability are located in the domain. However, under some circumstances it did converge (local symmetrically unstable areas seems to be tolerated) and more strangely, under some *stable* circumstances it did not. More investigation is required on the procedure with respect to the nonlinear component.

8.3 Runge-Kutta 4th order

PV-conservation

$$\frac{\partial q'}{\partial t} = -v_g \frac{\partial q'}{\partial y} - v_a \frac{\partial q'}{\partial y} - w_a \frac{\partial q'}{\partial z} = f(q'(t))$$

The PV conservation equation above illustrates that the q' changes locally in time by advection. In the QG case, only geostrophic advection by v_g is considered. Additional advection by the ageostrophic components is included in the SG case. We use *centered differences* in order to represent the spatial derivatives, i.e.:

$$\frac{\partial q'}{\partial y} \approx \frac{q'_{j+1,l} - q'_{j-1,l}}{2\Delta y}, \qquad \frac{\partial q'}{\partial z} \approx \frac{q'_{j,l+1} - q'_{j,l-1}}{2\Delta z} \tag{105}$$

The simplest way to discretize the time-derivative is to use forward Euler. In combination with (105), this is called the Forward-Time-Centered-Space or FTCS method given by:

$$\begin{aligned} q'^{n+1}_{j,l} =& q'^{n}_{j,l} - \Delta t \Bigg[v_{g(j,l)} \frac{q'_{j+1,l} - q'_{j-1,l}}{2\Delta y} \\ & + v_{a(j,l)} \frac{q'_{j+1,l} - q'_{j-1,l}}{2\Delta y} + w_{a(j,l)} \frac{q'_{j,l+1} - q'_{j,l-1}}{2\Delta z} \Bigg] \end{aligned} \tag{106}$$

This method is numerically unstable and seldomly used in practice.

The Runge-Kutta 4th order (hereafter RK4) method is based on a better representation of the local time-derivative. We use figure 23 to illustrate the basic idea behind RK4, namely that it is an extension of the simple forward Euler method (first-order) and the Midpoint method (second-order).

The black line in the figure represents a 'true' solution of a PV anomaly q' in time at a certain grid point (for example, just under a downward extending tropopause fold). It is an arbitrary curve drawn inside the domain. The red, blue and black dots at t_1 represent solutions from the Euler, Midpoint and RK4 method of which the latter is closest to the true solution. The green and yellow dots at t_1 are the third and fourth steps in order to derive the RK4 solution. All these solutions are determined graphically by drawing tangent lines to the black curve at several positions. It is a rough but quick way to illustrate the basic idea of the RK4 method.

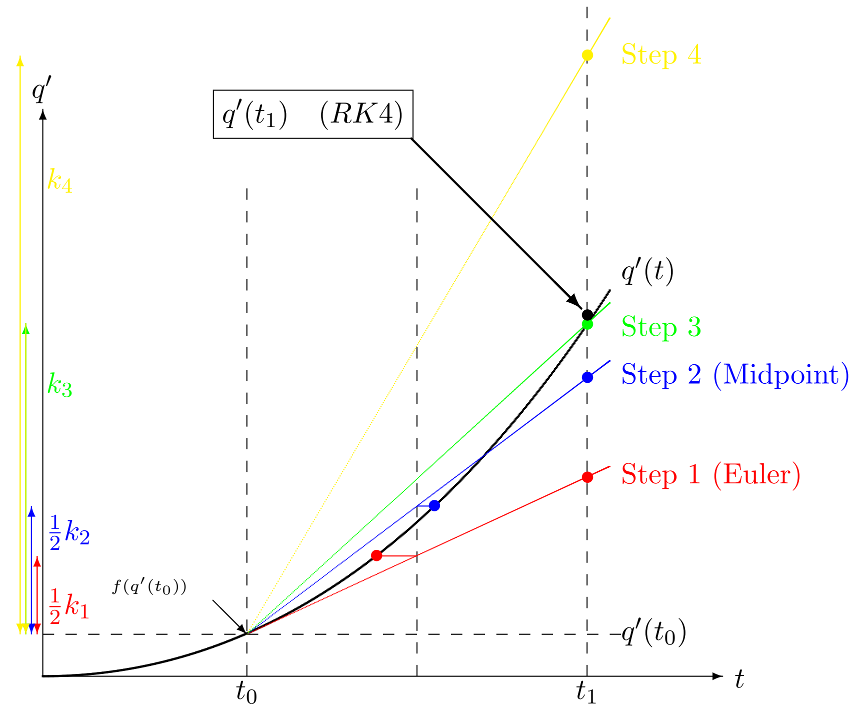


Figure 23 – Graphical representation of the Runge-Kutta 4th order method. The thick black line represents the 'true' value of q' as function of time t. The graphically estimated solution for $q'(t_1)$ of the RK4 method (black dot) is the result of four steps represented in the colors red, blue, green and yellow. Every next step uses information from the previous step. Graph is based on the graph in McMillan [2011].

The RK4 procedure is given by:

$$k_1 = \Delta t f(q'(t_0)) \tag{107a}$$

$$k_2 = \Delta t f(q'(t_0) + \frac{1}{2}k_1) \tag{107b}$$

$$k_3 = \Delta t f(q'(t_0) + \frac{1}{2}k_2) \tag{107c}$$

$$k_4 = \Delta t f(q'(t_0) + k_3) \tag{107d}$$

which are combined in the following way to derive the new $q'(t_1)$:

$$q'(t_1) = q'(t_0) + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{108}$$

which is also found in the code lines 1251 - 1288

The function $f(q'(t))$ corresponds to the rhs of the PV conservation equation representing (a)geostrophic advection. We can write the PV conservation equation shortly as:

$$\frac{\partial q'}{\partial t} = f(q'(t)) \tag{109}$$

Thus, the tangent to the function (black line) in figure 23 equals the value of advection term $f(q'(t))$ and is known at t_0.

Let us now go through these four steps, starting with k_1. If we would only use k_1 the result would be the simplest time integration scheme, namely Euler's method.

Euler

$$\frac{\partial q'}{\partial t} = f(q'(t_0))$$

Only k_1 is used in the forward Euler method, meaning that we simply use the tangent of the function at t_0 to calculate the new q' field at t_1. At t_1, the method will use the new tangent at $q'(t_1)$ to calculate $q'(t_2)$ and so on. But the discrepancy has already become large at t_1. Shorter time steps Δt would result in better results. However, Euler's method is only first-order and requires very short time scales. Moreover, it is vulnerable to numerical instabilities. Hence, forward Euler in combination with centered spatial differences (FTCS method) is seldomly used in practice.

Midpoint

$$\frac{\partial q'}{\partial t} = f(q'(t_0) + 1/2k_1)$$

The Midpoint method is based on the idea that, analogous to spatial coordinates, one can apply a centered time difference estimate instead of forward time difference which makes it a *second-order* method.

$$\left.\frac{\partial q'}{\partial t}\right|_{t_0+1/2\Delta t} = \frac{q'(t_1) - q'(t_0)}{\Delta t} \tag{110}$$

The discretized PV conservation equation now becomes:

$$q'(t_1) = q'(t_0) + \Delta t f(q'(t_0 + 1/2\Delta t)) \tag{111}$$

However, we do not have information on $f(q')$ at a half timestep beyond t_0. Our best guess to estimate $f(q')$ would be to use Euler's method. This means that we can guess $q'(t_0 + \Delta t)$ by:

$$q'(t_0 + \Delta t) \underbrace{\approx}_{Euler} q'(t_0) + 1/2k_1 \tag{112}$$

were k_1 is the change in q' according to Euler's method.

As a result, equation (111) becomes:

$$q'(t_1) = q'(t_0) + \underbrace{\Delta t f(q'(t_0) + 1/2k_1)}_{k_2} \tag{113}$$

in which we recognize the k_2 term.

This is illustrated in figure 23 by the red line crossing the dashed line at $t_0 + \Delta t$. The corresponding increase in q', or $1/2k_1$ is indicated by the nearby red dot. The tangent of q' at the red dot shall be used as the new slope for the Midpoint method which is indicated by the blue line. By using Euler's method up to halfway, the Midpoint method has become a better way to estimate $q'(t_1)$! Moreover, it is numerically stable in contrast with Euler.

Runge-Kutta 4th order

$$\frac{\partial q'}{\partial t} = \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

The idea of using simpler lower-order methods to guess q' at times beyond t_0 is interesting and can be applied several times. The Runge-Kutta methods are based on that idea and in our RK4 routine it is applied four times.

We can use the same expression as in (111) but now use the Midpoint method instead of Euler's method to guess the value of q' at $t_0 + \Delta t$. This results in an expression for k_3. In the last step we use information from the third step at t_1 instead in order to derive k_4.

These contributions from k_1 to k_4 are properly weighted in the final expression given in (108). In figure 23 the result is indicated by the black

Method	z-coordinate at t_1
True q'(t)	**32.9**
Euler	19.5
Midpoint	28.2
RK4	33.7

Table 6 – Comparison of three numerical methods. The values are the z-coordinates of the colored dots at t_1 in figure 23. These are estimated graphically by drawing tangent lines to the 'true' black curve at every step in the RK4 method.

dot at t_1. It is positioned closest to the true solution. The corresponding z-coordinates from the figure are given in table 6 and give an indication of the accuracy of the three methods.

We have discussed how the RK4 routine works for deriving the evolution of a PV anomaly as given by the code lines 1251 - 1288. The same discussion holds for the RK4 routine applied only on the surface boundary. The code lines 1291 - 1328 describe the routine in a similar way. The function $f(\frac{\partial \Psi_g}{\partial z})$ now represents the advection of buoyancy by v_g (QG case) plus v_a (SG case) on the surface.

8.3.1 Remarks

We use RK4 for the advection equation because it is an accurate method, requires relatively little computation time and does not include numerical diffusion. However, two remarks must be given for our use of RK4 *in the SG case.*

Firstly, one should be aware of the fact that $q'_{SG}(t_1)$ might be slightly underestimated with respect to the 'true' solution. We experienced for areas in our domain where $q'_{SG} \ll 1$ that this could result in negative q'_{SG} values after a few hours. Negative q'_{SG} values results in problems with the PV-inversion. It causes the SOR routine to diverge and the program may become unstable. After applying RK4, we set q'_{SG} to zero wherever q'_{SG} has become negative to fix this problem.

Secondly, we only apply RK4 routine on the jet PV anomaly q'_{jet} (see code lines 1251 - 1288). We assume the stratospheric anomaly to be constant in time, such that it remains a function of height only, i.e. $q'_{ts}(z)$. We then apply SOR_bc to solve the SGPV-inversion on both PV anomalies *separately* after calculating the advection of q'_{jet}. This results in a total buoyancy and geostrophic velocity field, including the time-independent $b_{ts}(z)$ implicitly.

In other words, we only consider the evolution of q'_{jet} which evolves by its own CFC! But $b_{ts}(z)$ does play an indirect role on the dynamics. The additional stratification in the stratosphere reduces the strength of the stratospheric CFC. The CFC in the troposphere is stronger than the stratospheric CFC and is responsible for tropopause folding.

It would be better to include the advection of the stratospheric PV anomaly q'_{ts}. In that case, $q'_{ts}(z)$ would evolve in time too and would become part of our dynamical system. This would pose us with an interesting *nonlinear* attribution problem (note: only in the SG case). We shall discuss here shortly the nonlinear interaction between q'_{ts} and q'_{jet} in time, which is not included in our program.

Assume that the RK4 routine also includes advection of $q'_{ts}(z)$. To illustrate the nonlinear interaction between both PV anomalies, imagine that:

(i) The ageostrophic circulation attributable to q'_{jet} would initially deform the troposphere-stratosphere transition, $q'_{ts}(z)$, too.

(ii) This creates a wave-like disturbance in the stratospheric PV component, hence $q'_{ts}(\mathbf{y}, z)$ becomes dependent on y.

(iii) The horizontal deformation field compresses the wave-like pattern of q'_{ts} in the cross-front direction.

(iv) A part of the frontogenetic forcing is now *attributable* to the deformation of q'_{ts}. As a result, 'another' ageostrophic velocity field has to compensate for this.

(v) In other words: we now have an ageostrophic field which is partly *attributable* to q'_{ts} and partly *attributable* to q'_{jet}. Returning to (i), both ageostrophic circulations will advect both PV anomalies simultaneously.

In time, which part of the ageostrophic circulation is attributable to q'_{ts} and which part is attributable to q'_{jet}? The attribution problem has been studied extensively by Thorpe [1997] diagnostically.

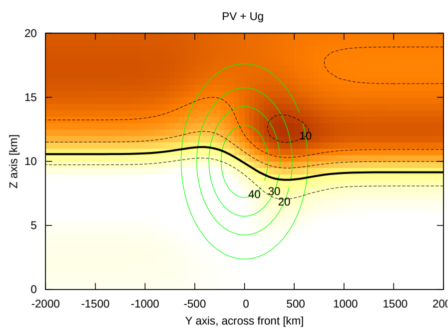
Is it possible to advect the *whole* PV field in time and perform the SGPV-inversion on the *whole* newly derived PV field? Why not avoiding the attribution problem? It is possible, but recall that the SGPV-inversion is a nonlinear elliptical equation. For that reason, SGPV-inversion has to be applied on both PV anomalies *separately* in order to derive the proper balanced u_g and b fields. Hence, the advection of both PV anomalies (by the nonlinear interacting CFC components) should ideally be derived separately with the RK4 method. We simplify here by considering the advection of q'_{jet} only.

Part III
Results

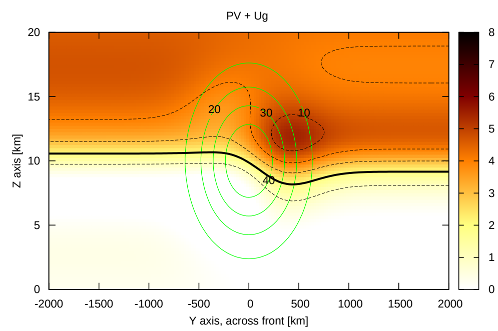
9 Idealized model runs

The results of the three idealized models runs (`QGsymjet`, `SGsymjet`, `SGcomjet`) shall be shown here. We first compare the QG results with the SG results for a symmetric jet. After that we focus on the differences between a symmetric and a more complex asymmetric jet based on SG theory.

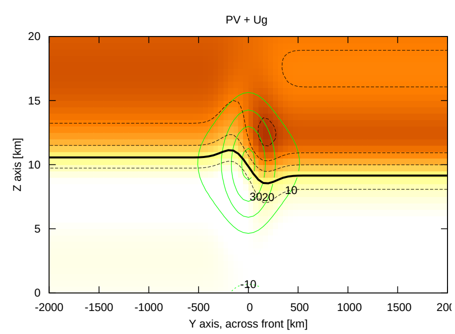
9.1 QG versus SG



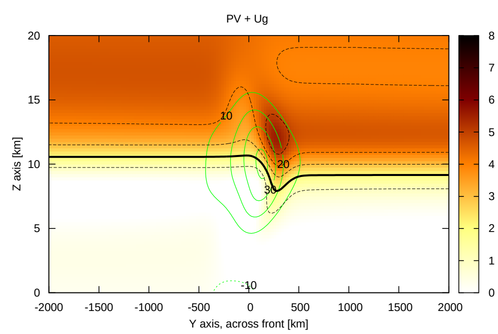
(a) Initial (QGsymjet)



(b) Initial (SGsymjet)



(c) +24 hours (QGsymjet)



(d) +24 hours (SGsymjet)

Figure 24 – Comparison between QGsymjet (left) and SGsymjet (right) model runs. The colored background is the PV field (in 10^{-4} s^{-2}) where the dynamic tropopause is indicated by the thick black line with a value of 2×10^{-4} s^{-2}. The jet stream u_g is indicated by solid green lines for positive values and dashed green contours for negative contours (each 10 ms^{-1})

We start by comparing the evolution of the PV field and the associated jet stream in figure 24. At initial time, the PV fields already differ. The equations for q_{QG} and q_{SG} are different as shown in (81). The tropopause has a wave-like shape in the QG case. The two additional nonlinear terms in q_{SG} add up to a tropopause having a downward extension only. But after PV-inversion, it results in the same jet stream configuration (green contours in figure 24).

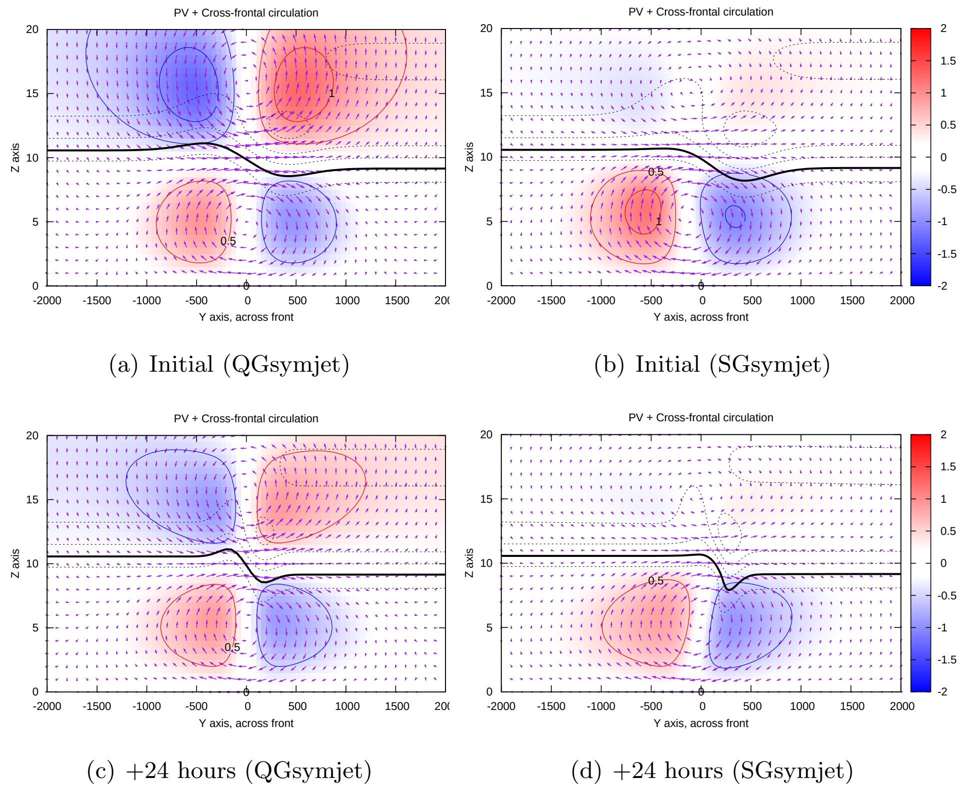
After 1 day in the `QGsymjet` run, both the PV field and the jet stream remain symmetric. Both structures are only compressed by the geostrophic deformation field. The jet stream is weakened by the deformation field on both sides. This is partly compensated by the existence of the CFC which has an accelerating effect on u_g, due to the Coriolis force on v_a. Near the surface, the same CFC adjustment results in negative velocities for u_g. The overall vertical wind shear has increased during time-integration but has become more local. Therefore, the thermal gradient has also increased locally (TWB).

The SG results after 1 day show additional deformation of the jet, because advection by the CFC is now included. The jet core moves northward, while on the surface the jet minimum moves southward. The frontal zone develops a more realistic slope. We see an increase of vorticity north of the jet core, as predicted in the introduction of chapter 5, and this contributes to the strength of the PV anomaly. The evolution of the symmetric jet actually motivated us to do a model run starting from an asymmetric jet configuration.

Figure 25 shows the corresponding cross-frontal circulations. The largest difference in w_a is found in the stratosphere. The results of QG theory show unrealistic strong vertical velocities in a strongly stratified area. We can explain this by our PV anomaly interpretation of the additional stratification in the stratosphere. It is not included in the QG form of the Sawyer-Eliassen equation (28), where a constant N^2 term is used instead. But N^2 is replaced by N_{eff}^2 in the SG form of the Sawyer-Eliassen equation (38), thus *including* the additional stratification. Therefore, the vertical velocity field reduces to more realistic values.

The vertical velocity field is initially stronger in the SG case than in the QG case. That is due to the additional convergence of the v_a in the warm sector, which can be interpreted as an additional frontogenetic forcing. The convergence also explains why the updraft is stronger than the downdraft. According to the continuity equation, horizontal convergence is related to an increase of w_a with height.

In time, we see that the CFC remains steady in the troposphere of the QG model run. It becomes weaker for the SG model run. We expected a stronger CFC instead of a weakening. A sensitivity analysis on several model



(a) Initial (QGsymjet)

(b) Initial (SGsymjet)

(c) +24 hours (QGsymjet)

(d) +24 hours (SGsymjet)

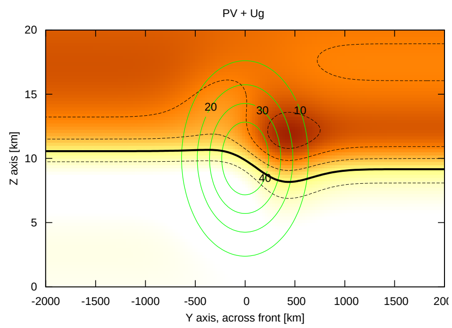
Figure 25 – Comparison between QGsymjet (left) and SGsymjet (right) model runs. The dynamic tropopause is similarly defined as in fig. 24. The purple arrows illustrate the cross-frontal circulation. Red contours are used for positive vertical velocities and blue contours for negative vertical velocities (each 0.5 cm s^{-1}).

parameters was performed in order to investigate the CFC weakening. We observed that the CFC became *stronger* in time when:

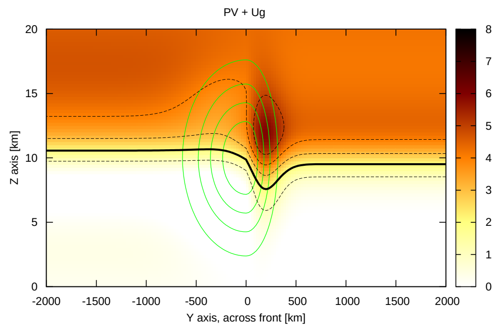
- The horizontal scale length of the jet (Y_{scale}) was set to larger values such that the horizontal scale of the frontogenetic forcing becomes larger too. A broader jet structure is not realistic though.
- A more realistic stretching deformation field was introduced, where A is larger in the center of the domain and remains 10^{-5} on the outskirts. This directly resulted in a larger frontogenetic forcing in the center and thus a stronger CFC.

The analysis showed that the sensitive interplay between the *scale* and the *strength* of frontogenetic forcing determines whether the CFC will weaken or strengthen in time. The initial model parameters determine which of the two effects dominates.

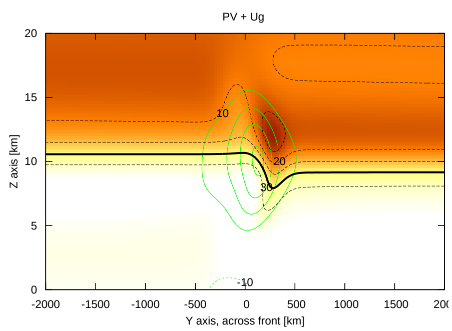
9.2 Symmetric jet versus complex jet



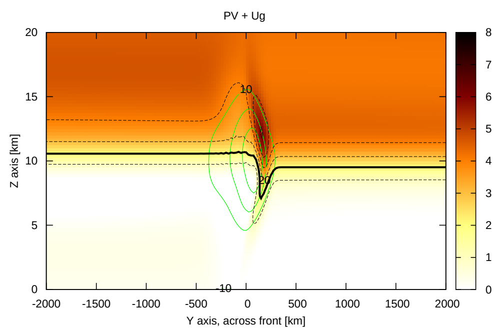
(a) Initial (SGsymjet)



(b) Initial (SGcomjet)



(c) +24 hours (SGsymjet)



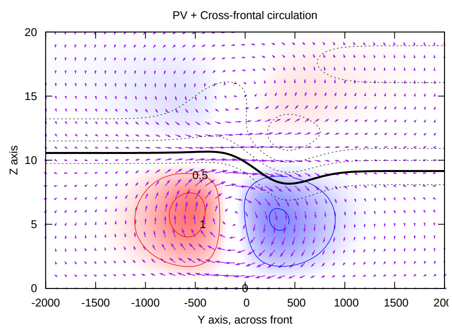
(d) +24 hours (SGcomjet)

Figure 26 – Comparison between SGsymjet (left) and SGcomjet (right) model runs. PV colors and u_g contours are similar to fig. 24

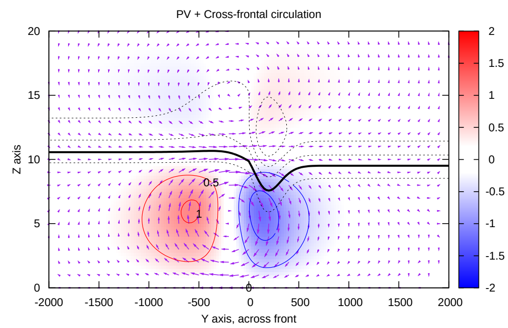
In the complex jet configuration, we start with a jet shape similar to the symmetric jet after 24h of integration. Figure 26 shows that this corresponds to a higher initial vorticity on the north side of the jet (indicated by the higher PV values or by the separation between isotachs).

Our simple formulation (code lines 624 - 637) has the disadvantage that it introduces a discontinuity in the center at y_0. The effect on the tropopause evolution is negligible, but for smaller values of $Y_{scale,1}$ the discontinuity could result in numerical instability. An upgrade on the formulation with the use of the continuous erf function (which we also just for troposphere-stratosphere transition) would solve this problem.

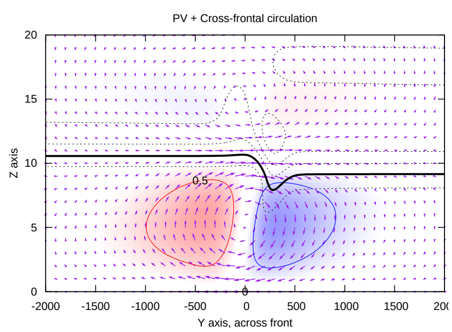
The model run `SGcomjet` shows a deeper tropopause fold after one day. Its shape is sharper and the dynamic tropopause even starts to fold under tropospheric air. Isolines of lower PV values around $(1 - 1.5) \times 10^{-4}$ s^{-2} show an even deeper and more characteristic folding structure (yellowish area under dynamic tropopause in fig. 26).



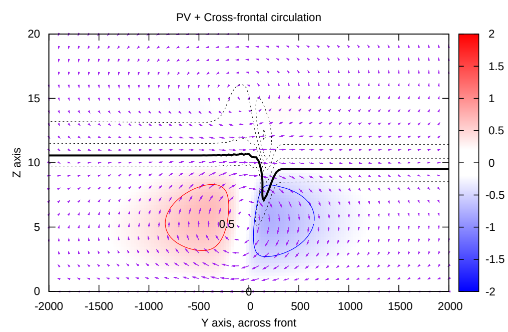
(a) Initial (SGsymjet)



(b) Initial (SGcomjet)



(c) +24 hours (SGsymjet)



(d) +24 hours (SGcomjet)

Figure 27 – Comparison between SGsymjet (left) and SGcomjet (right) model runs. Cross-frontal circulation represented by arrows and contours similar to fig. 25

The initial vertical velocity field, shown in figure 27, also becomes more characteristic for a tropopause fold. The downward velocities are stronger than in the SGsymjet run and the upward velocities are slightly weaker.

Unfortunately, the same weakening of the CFC is observed over time (at an even faster rate). We believe that a stronger local deformation field may counteract the CFC weakening.

With the use of three idealized model runs, we examined the evolution of an initial PV anomaly in this section. We saw that QG theory is not able to transport stratospheric air downward. SG theory applied to a simple jet can do that, but does not immediately result in realistic folding structures. A more realistic tropopause fold is related to an asymmetric jet deformed by a well-developed cross-frontal circulation. Such a circulation requires a reconsideration of our simple constant deformation field. These foundings will be useful in the interpretation of the case-study results in the next section.

10 Case-study

During our research the focus shifted from surface frontogenesis to frontogenesis near the tropopause and the initiation of tropopause folds. Therefore, we have chosen to study a case of a deep tropopause fold over Europe and Northern Africa in detail.

The goal of this case-study is to elaborate on the model run results in the previous chapter. We repeat our research questions here, but we apply them on our chosen case:

1. What are the differences between QG and SG PV-inversion applied on an observed PV field?

2. What is the contribution from stretching deformation in the evolution of a tropopause fold?

We will investigate this in two steps. First, we apply PV-inversion (both QG and SG) to calculate the vertical velocity field with our model and compare these with observations. Second, we simulate the evolution of a tropopause fold with our two-dimensional SGPV-inversion model.

10.1 The Algerian flood

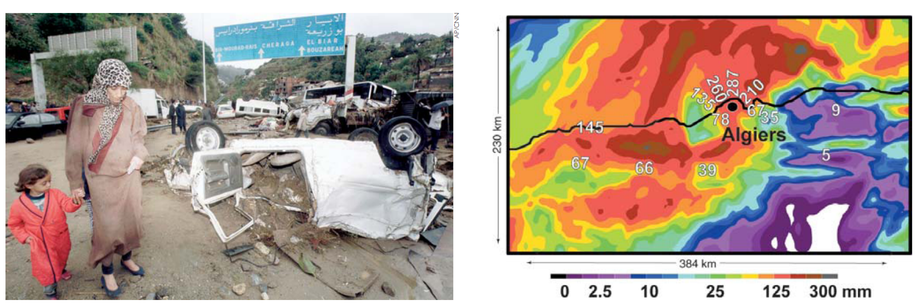


Figure 28 – Consequences of a rapidly formed meso-scale cyclone (≈ 100km) on the 9th and 10th of November 2001. Left: Devastation in Algeria caused by flooding rains and land slides. Right: Measured (white numbers) and simulated (color scale) total accumulating storm rainfall (in mm) along the North African coast. Simulation carried out by Tripoli et al. [2005].

The case-study is about the rapid formation of a meso-scale cyclone near the coast of Algeria on the 9th and 10th of November 2001. It caused severe weather in the Mediterranean Sea, especially in Algeria near the city of

Algiers. Excessive convective rainfall was measured and up to 300 mm rain fell within 48 hours, as shown in figure 28(b). These were the highest flood records since the beginning of measurements in 1908. It led to land slides on several locations and the consequences were devastating.

A detailed study on this case was carried out by Tripoli et al. [2005]. It took several years to collect and understand all the ingredients that caused this local severe weather. In their article, the ingredients are considered on both synoptic scale and meso-scale on the 9th and 10th of November. We focus here on the synoptic settings which will be described in terms of PV.

The rapid formation of the meso-cyclone above Algeria was related to an amplifying upper-troposphere PV anomaly coming down from Europe which *coupled* with the surface PV anomaly attributable to the strong surface heating above the dry African soil. This is visualized in figure 29(b) by the small updraft trajectory. It is related to deep moist convection developing right under the downward folding PV anomaly on the 10th of November.

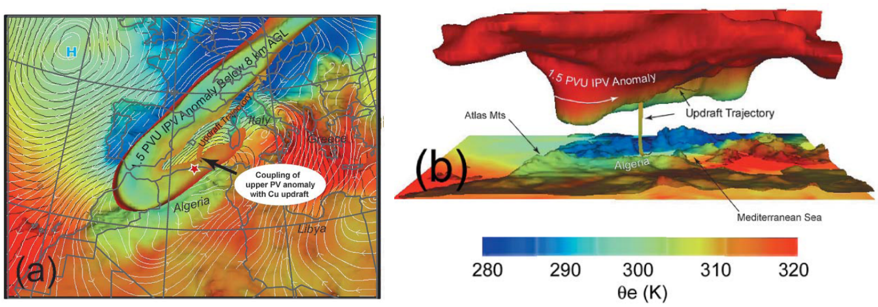


Figure 29 – Left: 5 PVU PV anomaly below 8 km crossing the whole of Europe and coupling with a Cu updraft above Algeria indicated by the red star. Surface pressure is shown by the white isolines. Right: Coupling of upper level 1.5 PVU PV anomaly with the updraft zone ahead, shown in 3D. Simulation from Tripoli et al. [2005].

We can understand this coupling event by examining the presence of the three most important ingredients for triggering severe convective storms: (i) moist, (ii) potential instability and (iii) a source of lift (van Delden [2001]).

Moist was brought into Algeria from the Mediterranean Sea by a low-level northeasterly jet. In combination with strong surface heating, it led to high equivalent potential temperatures in the boundary-layer, a first sign of potential instability.

The potential instability increased when cold air was being advected into the upper-troposphere over Algeria at the same time. An amplifying upper-level PV anomaly was responsible for that. Figure 29(a) shows how unusually strong the PV anomaly was on the 10th of November.

Only a source of lift was needed to release the enormous amount of built-up potential energy. The combination of the Atlas mountains near the coast and the uplifting branch of the CFC in front of the approaching PV anomaly was enough to trigger deep moist convection (Tripoli et al. [2005]).

The formation of these initial storms in the updraft zones was the final step towards a coupling of both PV anomalies. After the coupling, their simulation showed that the surface vorticity increased very rapidly resulting in the surface meso-cyclone that led to excessive convective rainfall continuing on the 10th of November.

It is clear that in such situations the role of upper-level PV anomalies can be crucial for the severness and duration of surface weather. The conservative nature of PV anomalies makes it a promising predictor of upper-level atmospheric conditions. Here, we will focus on the development of the tropopause fold over Europe and study its relation with frontogenesis.

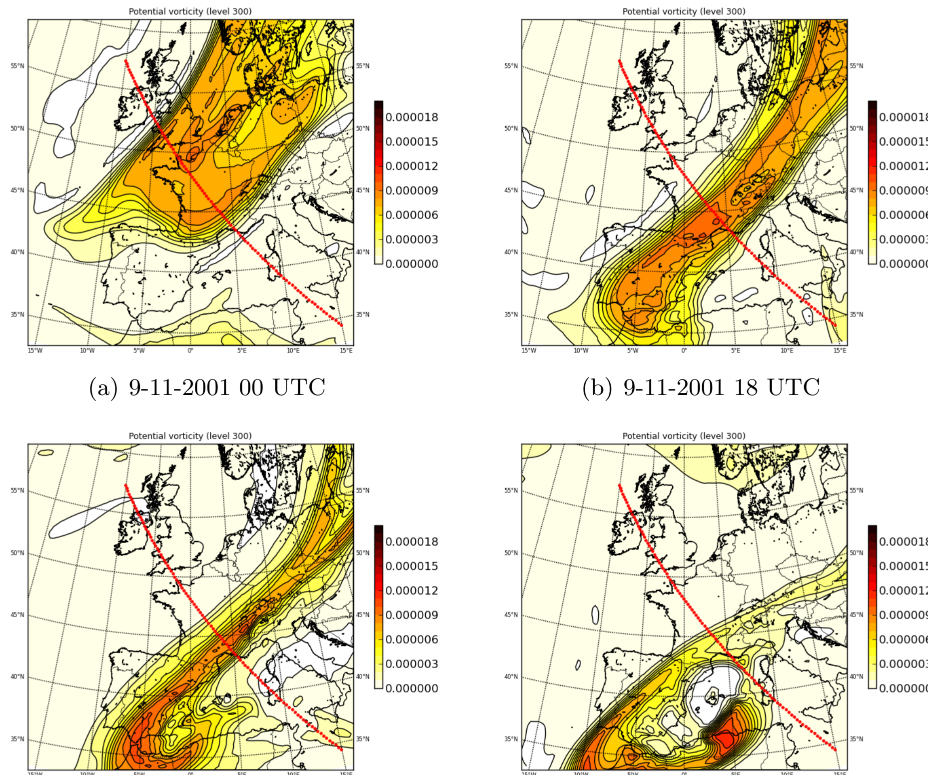
10.2 Observations

The figures in this subsection are directly obtained or calculated from data of the ECMWF operational data archive. From the year 2000 onwards, it also includes potential vorticity fields on pressure levels. Four PV fields are shown on a 300 hPa pressure level in figure 30, spanning a period of three days from the 9th until the 11th of November. The horizontal resolution is 0.25 degrees in both longitudal and meridional direction for all figures.

On the 9th of November 00 UTC, we see a large area over western Europe where the dynamic tropopause has descended beyond 300 hPa. These are called Coherent Tropopause Disturbance or CTDs by some authors (Donnadille et al. [2001]). The jet stream attributable to the PV anomaly is meandering from UK to Spain and back to Italy along the borders of the PV anomaly. The jet stream separates the southward moving cold air from the warm plumes moving northward over Ireland and Italy.

The anomaly has been expanding in southward direction for 18 hours. It retains an approximately two-dimensional structure. Variations in the southwest-northeast direction remain small compared to variations in the southeast-northwest direction. In some areas, the colors are darker hinting that PV is also extending towards the surface. This is an ideal configuration to examine with a two-dimensional model.

An important development is shown on the next day, the 10th of November, when the PV anomaly starts to curl over the Mediterranean Sea. It is an interesting area where the curvature effect results in additional convergence



(a) 9-11-2001 00 UTC

(b) 9-11-2001 18 UTC

(c) 10-11-2001 06 UTC

(d) 11-11-2001 00 UTC

Figure 30 – PV anomaly over Europe with Ertel's PV contours (each 1 PVU) on a 300 hPa pressure surface at the four given time steps. The red line over France corresponds to the position of the cross-section shown in figure 31. The data is from the ECMWF operational data archive.

resulting in larger vertical velocities. It is related to a developing strong jet streak over Spain and a favorable place for a very deep tropopause fold. But it is also a more complex three-dimensional structure not suitable for our model. We will examine the evolution of the PV anomaly over France where the structure remains two-dimensional.

In the last figure, corresponding to the 11th of November, we see that the anomaly is loosely connected to the main structure in the north at 300 hPa. We note that there is no real separation yet, which we can explain by the conservation principle of PV on isentropic surfaces. In the vertical direction, the anomaly is now coupled to the surface anomaly resulting in rapid cyclogenesis near the surface.

10.2.1 Cross-sections

We applied our model over France where the PV anomaly has an approximately two-dimensional structure. Figure 30 shows that the anomaly is stretched in the northeast-southwest direction and compressed in the perpendicular direction. We consider a cross-section along the axis of contraction to examine the vertical development.

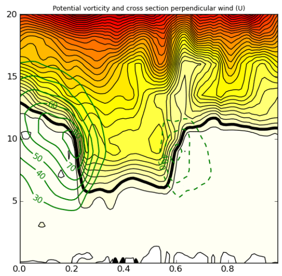
The cross-section is divided into 201 horizontal segments to simplify data input in our model. Horizontal interpolation is applied to estimate the new value for every cross-section point using the values at four nearby ECMWF grid points. In the vertical direction, we use information from 21 pressure levels of which the following in the troposphere: 200, 250, 300, 400, 500, 700, 850, 925, 1000 hPa. These levels are used for vertical interpolation to z-coordinates (height above mean sea level). The vertical scale is divided into 177 segments, up to the limit 40 km with a resolution of 250 m like in our model. The result is shown in figure 31 for $z \in [0, 20]$ km and y is made dimensionless from 0 to 1. The total horizontal distance is approximately 3300 km.

The cross-sections show the initial PV anomaly on the 9th of November again. Over time, it gradually becomes thinner by confluence. Another interesting aspect are the two downward extensions near the edges of the PV anomaly. In that sense, it shows many similarities with the very deep tropopause fold case in Donnadille et al. [2001]. In panels (d) and (e) of figure 31, the structure is, what some would call, a double fold structure.

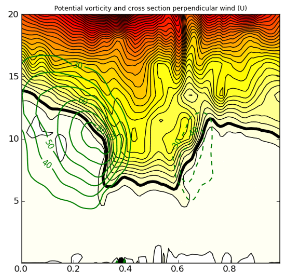
In their and our case-studies, we see that the true tropopause fold with the deepest stratospheric intrusion develops during the superpositioning of both 'folds' around 06 UTC on the 10th of November. The shape of the anomaly deforms in a more asymmetric way. It extends in western direction and the western jet streak and the related CFC become dynamically more important. A tropopause fold is developing over the next 12 hours, characterized by the following properties:

(i) The extension of the PV anomaly develops slightly under the western jet (visible in (g), (h) and (i) of figure 31). This corresponds to Shapiro's definition of a tropopause fold.

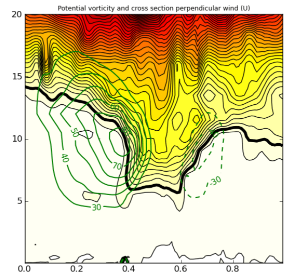
(ii) The transition from a two jet-core structure to an one jet-core structure as illustrated in panels (d) to (g). The eastern jet weakens and its core descends to lower altitudes. The western jet remains in position at 10 km.



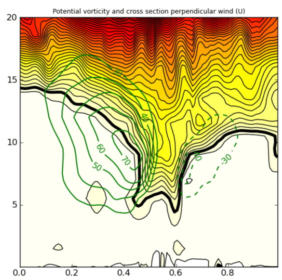
(a) 9-11-2001 00 UTC



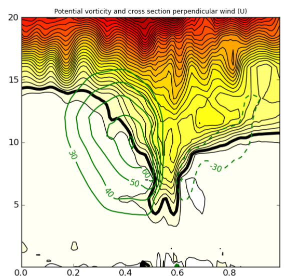
(b) 9-11-2001 06 UTC



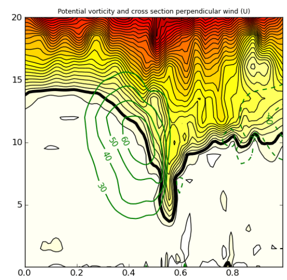
(c) 9-11-2001 12 UTC



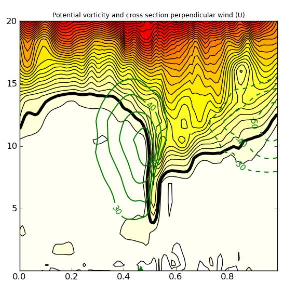
(d) 9-11-2001 18 UTC



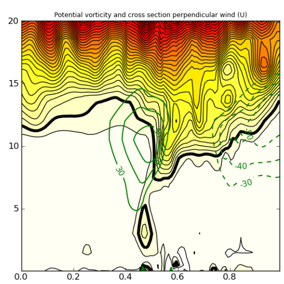
(e) 10-11-2001 00 UTC



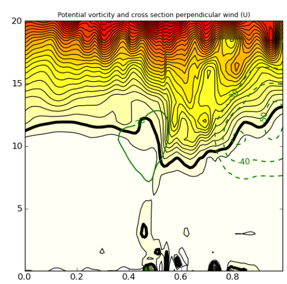
(f) 10-11-2001 06 UTC



(g) 10-11-2001 12 UTC



(h) 10-11-2001 18 UTC



(i) 11-11-2001 00 UTC

Figure 31 – Evolution of Ertel's PV (contours each 1 PVU) on a local (y, z) cross-section over France (red line in figure 30) where the thick line denotes the dynamic tropopause at 2 PVU. The along-front velocity (for $|u_g| > 30$ ms^{-1}) is visualized by green contours which are dashed for negative values (southwesterly flow) and solid for positive values (each 10 ms^{-1}). Data is over a period from 9-11-2001 00 UTC until 11-11-2001 00 UTC each 6 hours in the panels (a) until (i).

The folding process is relatively short and weak over France. The more we go in southern direction along the PV anomaly, the more clearer and deeper the tropopause structure develops. However, the along-front variations also become important, making it a three-dimensional process.

10.2.2 10-11-2001 06 UTC

Our hypothesis is that the tropopause fold is the result of deformation frontogenesis and the effect of the resulting cross-frontal circulation. Figure 32 shows observations of two frontal ingredients: a temperature field and a deformation wind field. We focus here on 06 UTC on the 10th of November, just before the tropopause fold starts to develop.

The potential temperature field on 500 hPa shows a double front structure along the cross-section. Along-front variations are small over France. The double front structure is visible throughout the troposphere, as shown in the cross-section. The thermal gradient is larger on the northwest side and is in TWB with the stronger developed western jet structure.

The stretching deformation A is shown in panels (c) and (d) of figure 32 and coincides nicely with the position of the tropopause fold. It is partly responsible for compressing the PV anomaly along with convergence and other processes. It is calculated from ECMWF data as follows:

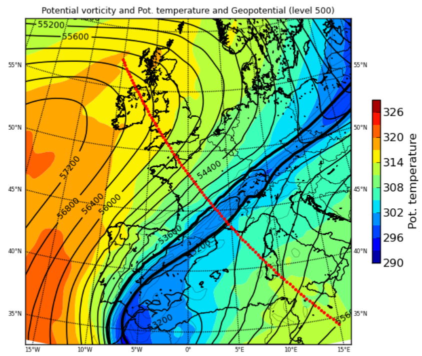
- The geostrophic velocity components u_g and v_g are derived from the geopotential field from ECMWF data using the relations $u_g = -f^{-1}\frac{\partial \phi}{\partial y}$, $v_g = f^{-1}\frac{\partial \phi}{\partial x}$ where x and y are zonal and meridional local coordinates.

- These geostrophic velocity components are redefined for the local rotated (say x' and y') coordinates of the cross-section.

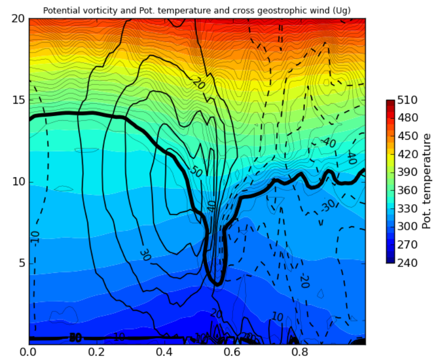
- The stretching deformation field is derived from the local velocity components and is given by: $A = \frac{\partial u_g}{\partial x'} - \frac{\partial v_g}{\partial y'}$.

Panel (c) shows that the largest deformation is found over the south of Spain. In general, this is the area within an amplifying upper level wave where the forcing by (i) horizontal confluence and (ii) the indirect vertical circulation result in the largest tropopause folds, and also where the western jet stream is being accelerated and becomes a jet streak (Mudrick [1974]). Although the folding process is weaker over France, we see that the deformation field and the CFC become large enough to initiate a fold.

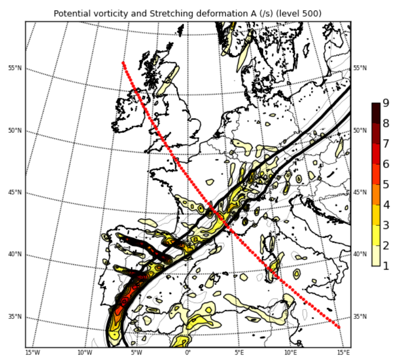
The observed vertical velocities are negative within the fold as one would expect (panels (e) & (f) of figure 32). The upward velocities seem not to be directly related to the CFC over France but more to orographic lifting over the island of Corsica. The separated red core near the surface might be part of the CFC. The velocities are in the order of 10 cm s^{-1} which corresponds



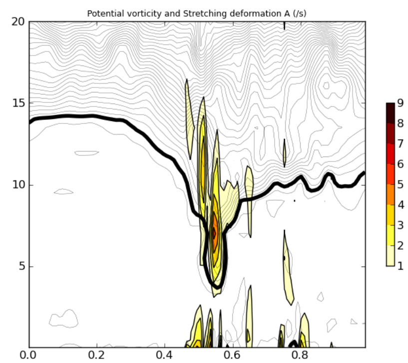
(a) map slice 10-11-2001 06 UTC



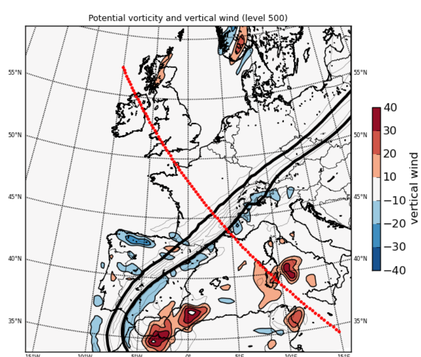
(b) cross-section 10-11-2001 06 UTC



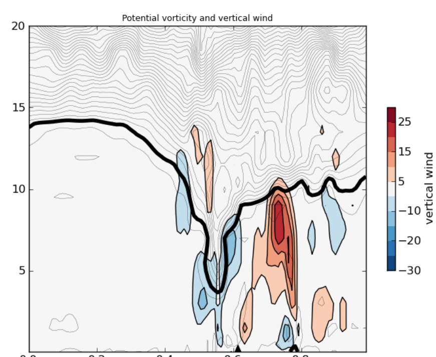
(c) map slice 10-11-2001 06 UTC



(d) cross-section 10-11-2001 06 UTC



(e) map slice 10-11-2001 06 UTC



(f) cross-section 10-11-2001 06 UTC

Figure 32 – Frontogenesis ingredients and vertical velocities shown on map slices at 500 hPa (left) and cross-sections (right), including the dynamic tropopause (thick black line at 2 PVU) in all figures. (a) potential temperature (color-filled contours each 3 K) and geopotential (each 400 m^{2}s^{-2}). (b) potential temperature (color-filled contours each 10 K) and cross-front geostrophic velocity (positive: solid, negative: dashed). (c)&(d): Stretching deformation A (color-filled contours each 10^{-4} s^{-1}). (e) & (f): vertical velocities (color-filled contours, (e) each 10 cm s^{-1}, (f) each 5 cm s^{-1}). Upward motions are in red, downward motions in blue.

to a descent of 2 km over 6 hours which seems to be a bit on the large side. We derived $w_{cm/s}$ by: $w_{cm/s} = -100 \frac{RT}{pg} w_{Pa/s}$ (where $w_{Pa/s}$ is retrieved from ECMWF data) and interpolated $w_{cm/s}$ vertically to z coordinates.

The largest upward velocities are found where the indirect vertical circulation is largest, between Spain and Morocco and moving towards Algeria. Additional processes play a role here, namely: (i) additional convergence due to the curvature of the PV anomaly and (ii) cold-air shear advection (or "Shapiro effect"). These effects were found to be important in another case-study of a fold described by Donnadille et al. [2001].

Let us return to our cross-section over France. The vertical velocities here are the result of various frontogenetic processes besides stretching deformation (also divergence, shearing deformation). Our goal is to estimate the importance of stretching deformation using our model. Therefore, we need input in the form of (i) a reference atmosphere, (ii) PV anomalies and (iii) a valid representation of a stretching deformation field. The conversion from the previously shown observations to the required input fields shall be discussed now.

10.3 Modifications

The ECMWF operational data archive contains data on the total three-dimensional wind field, temperature field, etc. We will use these data to study the relation between geostrophic forcing and ageostrophic adjustment. The challenge is (i) to split the velocity fields into balanced and unbalanced components and (ii) to derive reference profiles and perturbations for potential temperature.

Geostrophic balanced flows are given in terms of the geopotential field ϕ. It is therefore possible to derive from these fields the initial PV fields corresponding to QG and SG balance. However, we experienced that the use of ϕ observations for the derivation of initial PV fields resulted in noisy fields, not suitable for applying PV-inversion.

Fortunately, Ertel's PV fields are also available from the ECMWF archive and these provide us with a better alternative. Ertel's PV in pressure coordinates and hydrostatic form is given by (Hoskins et al. [1985]):

$$q(x,y,p) = -g \underbrace{\left(\zeta + f - \frac{\partial v}{\partial \theta}\frac{\partial \theta}{\partial x} + \frac{\partial u}{\partial \theta}\frac{\partial \theta}{\partial y}\right)}_{I} \underbrace{\frac{\partial \theta}{\partial p}}_{II} \tag{114}$$

It consists of two factors:

I Absolute vorticity + two horizontal vorticity terms. Contributes to Ertel's PV in the form of PV anomalies in areas with large relative vorticities. It is mainly determined by the dynamical configuration of the atmosphere.

II Stratification. This factor is dominated by the vertical stratification of the atmosphere, especially in the strongly stratified stratosphere.

We can use the definition of potential temperature to expand factor II in Ertel's PV as follows:

$$\begin{aligned} q &= -g(\zeta + f - \frac{\partial v}{\partial \theta}\frac{\partial \theta}{\partial x} + \frac{\partial u}{\partial \theta}\frac{\partial \theta}{\partial y})\left[\frac{\partial T}{\partial p}\left(\frac{p_r}{p}\right)^{\kappa} - \frac{T\kappa}{p_r}\left(\frac{p_r}{p}\right)^{\kappa+1}\right] \\ &= -g(\zeta + f - \frac{\partial v}{\partial \theta}\frac{\partial \theta}{\partial x} + \frac{\partial u}{\partial \theta}\frac{\partial \theta}{\partial y})\left[\frac{\partial T}{\partial p}\left(\frac{p_r}{p}\right)^{\kappa} - \frac{T\kappa}{p_r}\left(\frac{\theta}{T}\right)^{\frac{\kappa+1}{\kappa}}\right] \end{aligned} \tag{115}$$

where in the last step we used the potential temperature definition again and where $\kappa = R/c_p \approx 2/7$. In the stratosphere, the atmosphere is rather close to isothermal such that the first term inside the brackets is relatively small. The second term shows the dominant relation between PV and Θ in the stratosphere given by: $q \propto \theta^{9/2}$. The background stratification mainly determines the vertical structure of Ertel's PV.

The dominance of the background stratification makes it difficult to examine horizontal variations in vertical cross-sections. For example, the jet stream is related to horizontal variations of the PV anomaly. But in the inversion process, it would be difficult to extract the jet because of the dominance of variations in the stratosphere (see figure 31).

We will now rescale Ertel's PV by a factor $\theta^{-9/2}$ in order to filter out this height dependence. We use the same scaling approach as described in Lait [1994] and more extensively in Juckes [1999].

The definition of the resulting scaled PV, $\hat{q}$, is given by:

$$\hat{q} = \frac{q}{S_{ref}(\theta)} \tag{116}$$

where:

$$S_{ref}(\theta) = -g\frac{\partial \theta_{ref}}{\partial p} \tag{117}$$

corresponds to a static reference atmosphere. It is important to choose an appropriate reference atmosphere, one that reduces the dominant increase of θ throughout the stratosphere. Various formulations for $\frac{\partial \theta_{ref}}{\partial p}$ are discussed in Juckes [1999]. We will use an isothermal reference atmosphere.

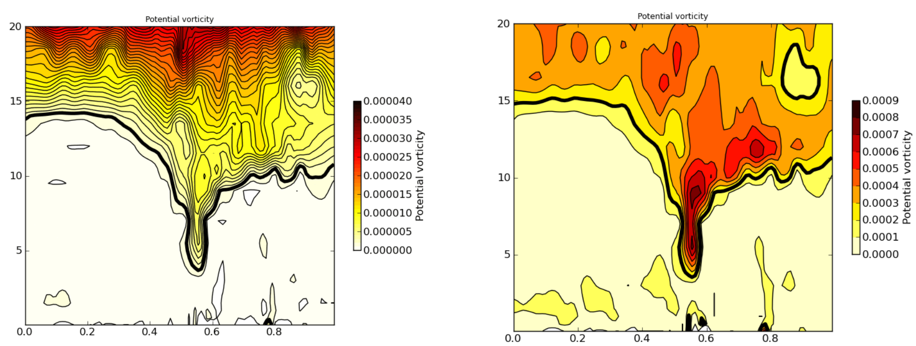
The potential temperature only depends on pressure p for an isothermal reference atmosphere ($T_{ref} = T_0$). This simplifies the derivation:

$$\begin{aligned} -g\frac{\partial \theta_{ref}}{\partial p} &= -gT_0\kappa\left(\frac{p_r}{p}\right)^{\kappa-1}\frac{-p_r}{p^2} \\ &= \frac{gT_0\kappa}{p_r}\left(\frac{p_r}{p}\right)^{\kappa+1} \\ &= \frac{gT_0\kappa}{p_r}\left(\frac{\theta}{T_0}\right)^{\frac{\kappa+1}{\kappa}} \end{aligned} \tag{118}$$

where in the last step, the definition of potential temperature is substituted. The scaling factor S_{ref} has now been expressed in terms of θ only. With $\kappa = R/c_p \approx 2/7$, we see indeed that $\frac{\partial\theta}{\partial p} \propto \left(\frac{\theta}{T_0}\right)^{9/2}$.

The equation for $\hat{q}$ becomes:

$$\hat{q} = \frac{-g(\zeta + f - \frac{\partial v}{\partial \theta}\frac{\partial \theta}{\partial x} + \frac{\partial u}{\partial \theta}\frac{\partial \theta}{\partial y})\frac{\partial \theta}{\partial p}}{\frac{gT_0\kappa}{p_r}\left(\frac{\theta}{T_0}\right)^{\frac{\kappa+1}{\kappa}}} \tag{119}$$



(a) Ertel's PV (b) Scaled PV

Figure 33 – Cross-sections showing (a) Ertel's PV and (b) scaled PV, $\hat{q}$. The dynamic tropopause is given by the 2 PVU thick black isoline in (a) and by the 2×10^{-4} s^{-1} thick black isoline in (b). Other PV isolines are given the thin black lines each 1 PVU in (a) and each 10^{-4} s^{-1} in (b).

It is shown in appendix D that $\hat{q}$ has a similar form as the semi-geostrophic PV or the quasi-geostrophic PV for $Ro \ll 1$. Because the scaling factor only depends on θ, $\hat{q}$ is also materially conserved! Using this alternative form of PV, we can derive all other fields from PV-inversion.

The new $\hat{q}$ field is shown in figure 33 and one should recognize the stratospheric anomaly and the dynamic PV anomaly in it. For the temperature T_0 of the reference atmosphere, we have taken $T_0 = 300$ K. The main $\theta^{9/2}$ height dependence has been removed and some minor horizontal variations are visible which are small compared to the notable PV anomaly.

The model results that are shown in the next subsection are based on the assumption of a *constant* deformation field $A = 10^{-5}$ s^{-1} as we used in our theoretical models. Thus, we underestimate the deformation rate near the fold, but overestimate the deformation rate on the outskirts of the domain. The average rate of deformation over the domain is approximately the same and it allows us to compute the evolution of the fold in time with a comparable rate of deformation.

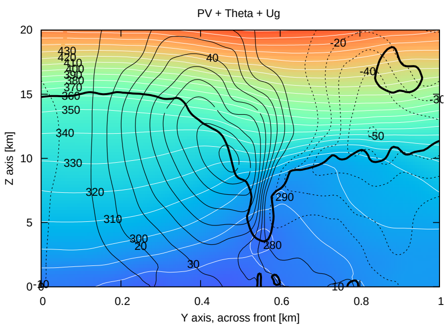
10.4 Model results

The initial PV field is prepared for a QG or a SG model run. The reader is referred to figure 17 for an overview on the model algorithm used to produce the results here. This overview shows that we also need to supply lateral and surface boundary conditions. For the lateral boundaries, we use u_g calculated from available ϕ data. We use the buoyancy, with respect to the reference temperature $T_0 = 300$ K, on the surface boundary.

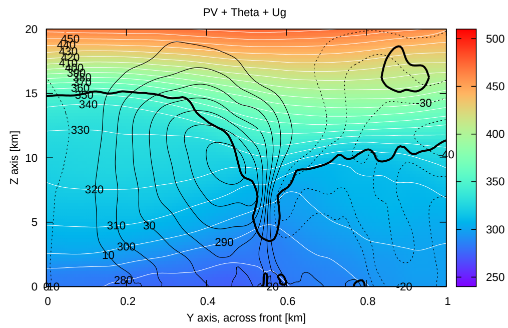
We will first show diagnostic results where we compare the QG model with the SG model applied on the scaled PV field at 10 November 2001, 06 UTC. After that, we will discuss the contribution of stretching deformation in the observed tropopause folding process using a SG model run of 18 hours.

10.4.1 QG versus SG

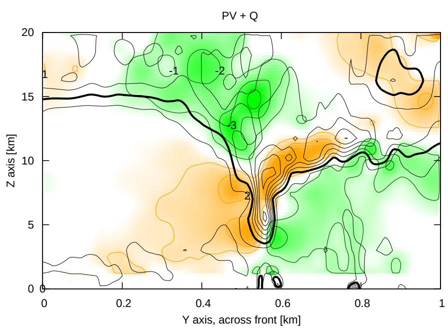
We apply PV-inversion for both QG and SG cases. It should be noted that the initial field, $\hat{q}$, is the *same* for both cases. This is different from the jet configurations we analyzed earlier, where q_{SG} contained two additional non-linear terms. After PV-inversion, the geostrophic velocity, buoyancy, frontogenetic forcing and ageostrophic circulation fields are derived and shown in figure 34. Table 7 provides a quick overview of the differences. The results can be compared with the fields from the ECMWF archive shown in figures 32.



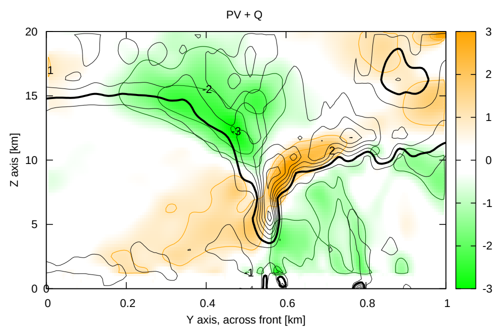
(a) QG solution



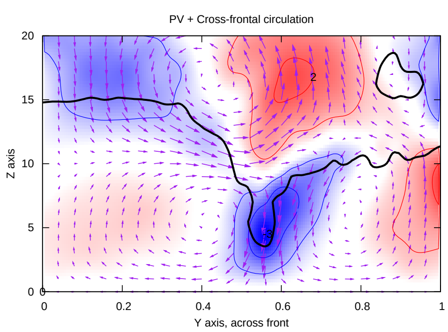
(b) SG solution



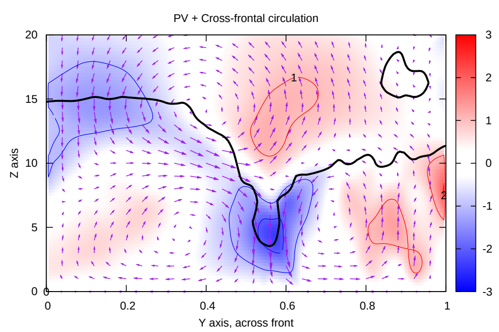
(c) QG solution



(d) SG solution



(e) QG solution



(f) SG solution

Figure 34 – Comparison between QG (left) and SG (right) PV inversion. (a) & (b): dynamic tropopause (thick black line), positive (solid) and negative (dashed) geostrophic jet velocities (black lines each 10 ms^{-1}) and potential temperature field in color-filled white contours (each 10 K). (c) & (d): including frontogenetic forcing as color-filled area. Green for negative Q and orange for positive Q. (e) & (f): including the cross-frontal circulation. w_a is illustrated by color-filled contours (each 1 cm s^{-1}) where blue is used for negative velocities and red for positive velocities. The purple arrows represent the cross-frontal circulation. Results are derived for 06 UTC on the 10th of November 2001.

Parameter	Analysis	QG	SG
$\max(u_g)$ (ms^{-1})	55	90	75
$\min(u_g)$ (ms^{-1})	-30	-20	-25
$\max(Q)$ (10^{-10} s^{-3})	n.a.	3	2.5
$\min(Q)$ (10^{-10} s^{-3})	n.a.	-2.5	-1.5
$\max(v_a)$ (ms^{-1})	n.a.	3	2.5
$\min(v_a)$ (ms^{-1})	n.a.	-2.5	-2
$\max(w_a)$ (cm s^{-1})	10	1	1
$\min(w_a)$ (cm s^{-1})	-10	-3	-2

Table 7 – Comparison of QG and SG results with ECMWF operational data archive. Values are estimated from figures and are directly related to the PV anomaly, e.g. higher values near boundaries are not considered.

Both QG and SG results in figure 34 show the basic features of a tropopause fold in development:

- A jet core west of the anomaly at 10 km height where the horizontal PV gradient is largest.
- A double front structure beneath the anomaly.
- Larger separation between isotherms on both sides of the anomaly, indicating reduced static stability.
- Frontogenetic forcing is mostly positioned on western side of the anomaly, left of the descending PV structure, where the product of the thermal gradient and deformation field is largest.

The largest difference between the QG and SG results is the strength of the jet. In both cases, the strength is overestimated. But the SG PV-inversion does a better job on both the strength of the jet as on representing the lower static stabilities on both sides.

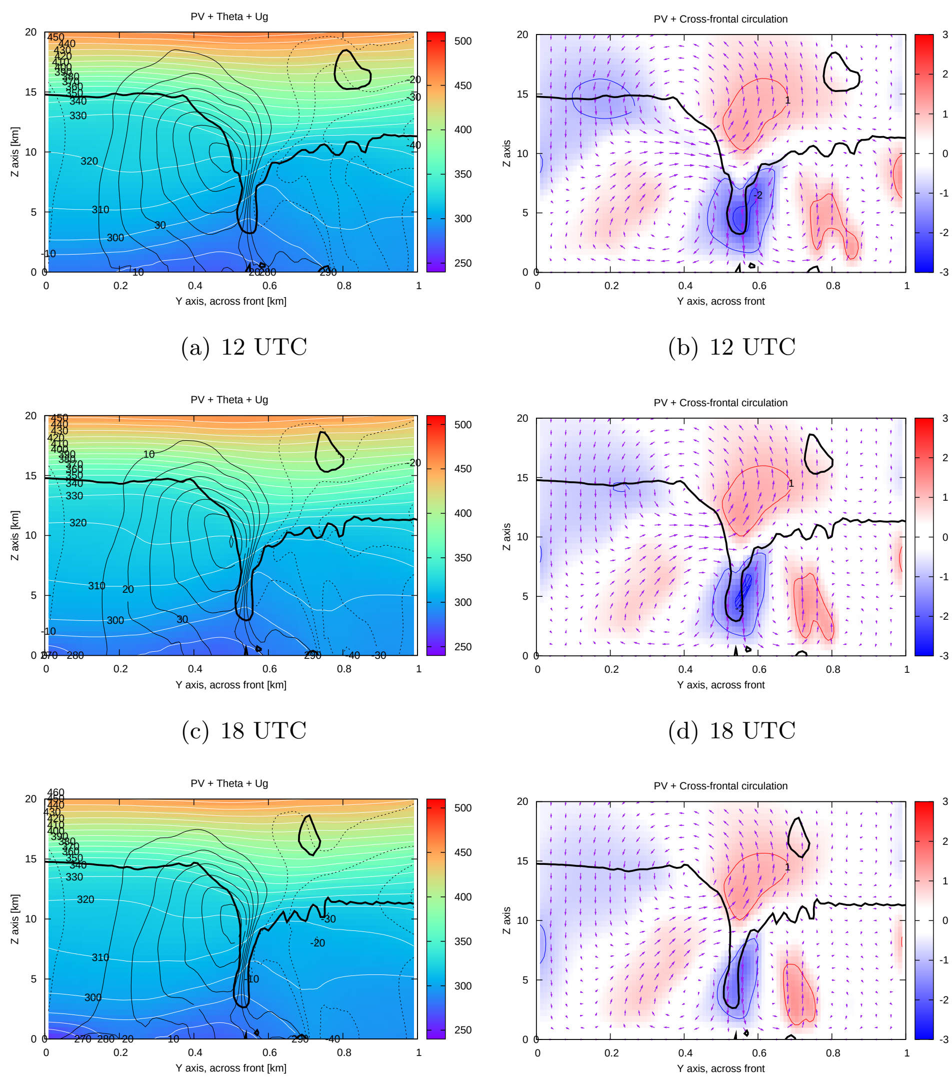
Panels (e) and (f) of figure 34 show that SG theory gives better results for the CFC, i.e. it is more localized. In the stratosphere, w_a is overestimated in the QG case, as expected. However, w_a inside the fold is also larger for the QG case compared to the SG case. The QG values seem to be more realistic (i.e. closer to the ECMWF archive data), but their values are the result of the unrealistic geostrophic velocities in panels (a) and (b). The main reason for these low values is that we use a constant deformation field $A = 10^{-5}$ s^{-1} for both cases, whereas panels (c) and (d) of figure 32 suggest that A must have higher values locally.

10.4.2 Evolution of a fold

The true value of SG theory becomes apparent when studying a tropopause folding process. We saw in the three idealized model runs that QG theory was not able to produce a tropopause fold. Therefore, we will only show SG results for an 18 hour time integration starting at 06 UTC, 10th of November.

The results are shown in figure 35. The most interesting aspect is the increasing role of the cross-frontal circulation in the folding process. That is nicely illustrated by the ageostrophic vortices moving towards the fold. Their strength has slightly decreased, but we see that the downward velocities remain positioned at the tip of the fold. The horizontal ageostrophic wind blows steadily at ≈ 10 km height and advects the base of the fold in eastern direction. This causes the PV anomaly to become tilted. The jet stream also weakens during the 18 hours of integration. The eastward movement of the jet core is associated with positive vorticity advection (PVA) in the direction of the fold.

The model is not always robust to (numerical) instabilities. The bottom figure in 35 illustrates this in the form of growing sawtooth structures on the dynamic tropopause. The input of a smooth PV field where artificial PV anomalies are filtered out, is a must in order to let the PV-inversion routine work for more than one day.



(a) 12 UTC (b) 12 UTC

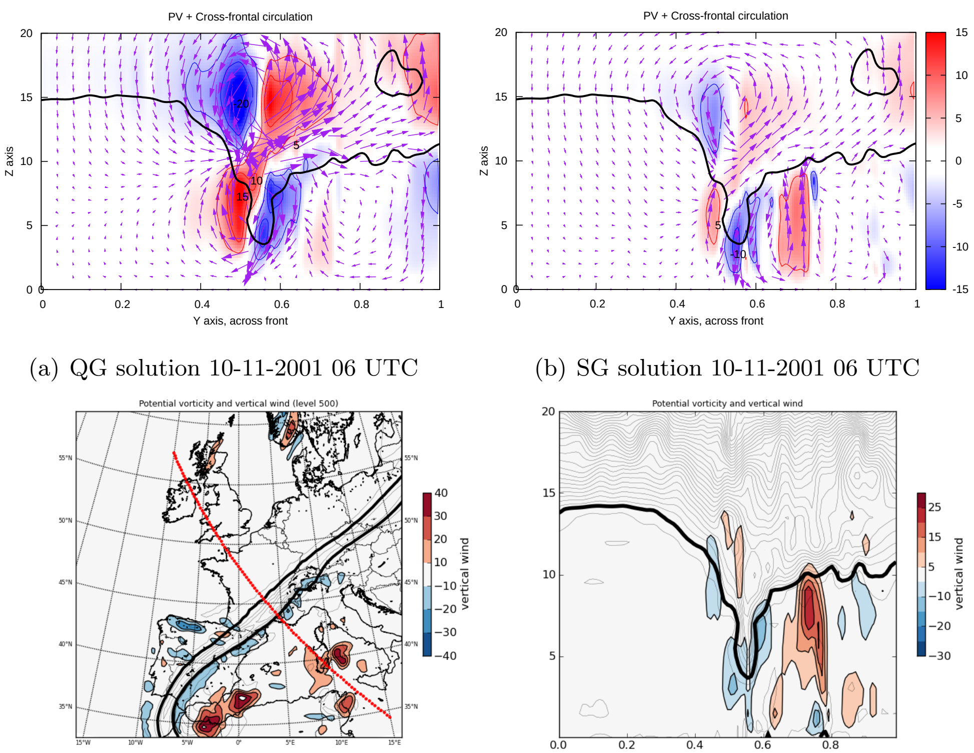
(c) 18 UTC (d) 18 UTC

(e) 00 UTC (f) 00 UTC

Figure 35 – Same as figure 34, but now only SG model run results, showing the evolution of a tropopause fold at time steps 12 UTC (top), 18 UTC (center) and 00 UTC on the 11th of November (bottom). Simulation started on the 10th of November 06 UTC.

10.4.3 Deformation field from archived data[1]

Although the stretching deformation field derived from ECMWF data is very noisy, some first results shall be shown here. The vertical velocities derived from the 'observed' A fields are shown at the top of figure 36 for the QG and SG case.



(a) QG solution 10-11-2001 06 UTC

(b) SG solution 10-11-2001 06 UTC

(c) map slice 10-11-2001 06 UTC

(d) cross-section 10-11-2001 06 UTC

Figure 36 – (a) & (b): Same as panels (e) and (f) of figure 34, illustrating the cross-frontal circulation by purple arrows and w_a by color-filled contours. The results are based on a stretching deformation field derived from ECMWF archived data. (c) & (d) Same as panels (e) & (f) in figure 32 showing w_a observations for reference.

We tried several ways of smoothing and averaging. For this particular timestep, we took a vertical average of A from the surface to 10 km height where the stretching deformation field is mostly present. As a result, the

[1]This work is still in progress. No completely satisfying method has yet been found to modify the noisy A data into a reliable smooth stretching deformation field for our model.

stretching deformation A derived from the ECMWF archive only varies in the horizontal direction. Finally, by the smoothing effect of the operator N_2 of the Sawyer-Eliassen equation, the cross-frontal circulation field acquires a recognizable structure. The results have improved with respect to the constant deformation field $A = 10^{-5}$ as the differences between QG and SG are clearer.

In the QG case, the vertical velocities are overestimated in the stratosphere because the additional stratification of the stratosphere anomaly does not play a role. The red area in the troposphere is not realistic either.

The SG results show smaller and more realistic velocities in the stratosphere. These are smaller because of the additional stratification in the stratosphere included in the Sawyer-Eliassen equation. Moreover, the downward velocities near the tip of the fold are very close to the observational values. The updraft zone on the east side of the fold is more local compared to panels (e) and (f) of figure 34.

The term A also plays a role in the advection equation, but unfortunately we could not derive the evolution of a tropopause fold based on the noisy A fields. This is partly related to the artificial velocity fields near the eastern boundary.

These results show that a careful modification and application of the observed A fields rewards itself with a better representation of the tropopause fold and the related cross-frontal circulation.

11 Summary & conclusion

Our goal in this thesis is to understand the dynamical role of the cross-frontal circulation (CFC) in the initiation and development of a tropopause fold. We addressed the following two research questions:

1. What does the semi-geostrophic approach add to the quasi-geostrophic approach in the representation of a cross-frontal circulation and a tropopause fold?
2. Can we simulate a realistic tropopause folding event with a two-dimensional semi-geostrophic model based on stretching deformation?

We focused on several aspects in detail to understand the interplay between the CFC and a tropopause fold, namely: (i) a theoretic study of the dynamics using the modern framework of potential vorticity, (ii) a detailed numerical description of our two-dimensional PV-inversion model and (iii) a tropopause fold case-study.

11.1 Theory

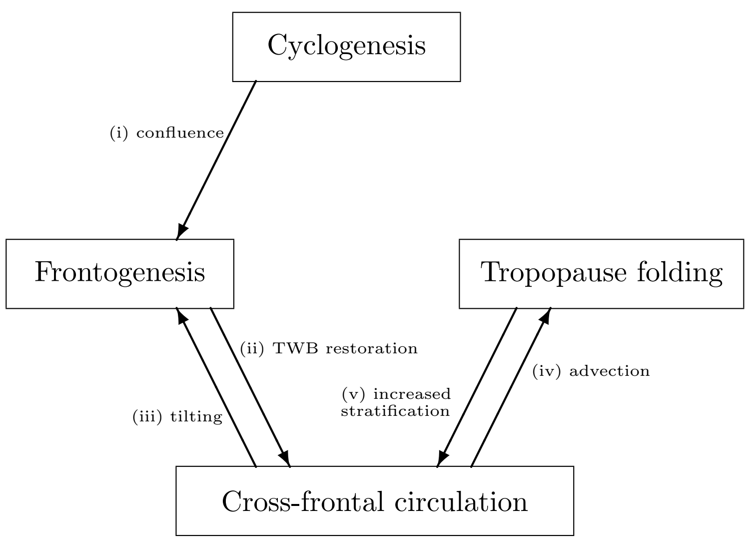


Figure 37 – Schematic overview showing physical relations between different dynamical structures on various scales. Only relations considered in this thesis are shown

In the introduction, it was said that the challenge for a researcher is to recognize 'cause' and 'effect' relations in the complex nonlinear atmosphere.

We focused here on the process of frontogenesis in the upper atmosphere, in particular on the cross-frontal circulation component and its dynamical *effect* on the tropopause and near the surface front (illustrated by the arrows (ii) and (iv) in figure 37). In the theoretic part, we also described the other relations in order to give a comprehensive overview on the position of the CFC with respect to cyclogenesis and frontogenesis, as shown in figure 37.

We considered cyclogenesis as a constant factor in our study. In chapter 2, we described how a steady configuration of (anti)-cyclones on synoptical scale results in confluence. In other words, a geostrophic wind field that transports two air masses towards each other. Frontogenesis takes place where both air masses meet and involves compression across-front and stretching along-front. During frontogenesis, the atmosphere is continuously brought out of thermal wind balance.

In chapters 4 and 5, we used the balanced theories of quasi-geostrophy and semi-geostrophy to interpret the cross-frontal circulation as the balance restoring component of a frontal system. Both the QG and SG set of equations were separated into an advection equation for q', related to frontogenesis and the Sawyer-Eliassen equation, describing the response of the CFC. In the QG case, the advection equation only contains frontogenesis due to the large-scale confluence field and *excludes* front deformation by the CFC. Hence, in this case arrows (iii) and (iv) in figure 37 are absent. In the SG case, advection by the CFC is included and it is related to the additional tilting effect. This increases the rate of frontogenesis and has an 'action at distance' effect on the tropopause.

We introduced the conserved PV field in the theoretic part and used it to understand the (non)linear interplay between two PV perturbations, representing a front in TWB with a jet stream q'_{jet} and the tropopause q'_{ts}. The description of frontogenesis and tropopause folding in terms of PV has several advantages, especially in the SG case. A positive PV field represents a balanced atmosphere and negative PV values indicate areas of (symmetric) instability. Moreover, it is possible to derive all other variables diagnostically from the PV field. We studied the dynamical role of the CFC in terms of PV by examining the interaction of q'_{jet} (including the CFC implicitly) with itself and with q'_{ts}.

We discussed the transformation to geostrophic coordinates shortly in chapter 6. The basic idea is that the nonlinear semi-geostrophic system of equations reduces to a linear system of equations in geostrophic space, similar to the quasi-geostrophic equations. One can understand this by the fact that only geostrophic motion is shown in geostrophic space, whereas advection by ageostrophic motions has become implicit.

11.2 Numerical model

We used a two-dimensional model to study the role of the CFC in tropopause folding. The ingredients for the model were discussed in section 7. The model consists of two steady background fields, namely a stretching deformation field and a vertical reference temperature profile. This reference state is represented by a 'constant-N' background temperature profile, which corresponds to a linear increase of the background potential temperature $\overline{\theta}$ with height. The dynamics are represented by the superposition of two PV anomalies, namely an initial jet structure in TWB with a pole-to-equator frontal zone and a continuous troposphere-stratosphere transition. The latter PV 'anomaly' depends only on height and is a simple but effective way to model the additional stratification in the stratosphere.

Given these ingredients, the flow diagram in figure 17 illustrates how a PV anomaly, q'_{QG} or q'_{SG}, is processed in the model. Both QG and SG algorithms are based on the derived equations in the theoretic part and are discussed in chapter 8.

An important challenge in this chapter is the use of the SOR and RK4 techniques to solve the SG set of equations for *two* PV anomalies. This is important because it determines how a frontal system and a tropopause (i.e. the two PV anomalies) are interacting with each other in the model. The numerical routines for SGPV-inversion and PV advection play a central role here.

Starting with the SGPV-inversion, we saw that the procedure involves an additional iteration loop to derive the two nonlinear terms (see thick box in figure 22). The nonlinear terms are responsible for the difference in the solution for Ψ_g, when we apply SGPV-inversion (I) on both PV anomalies separately and (II) on the PV field as a whole.

(I) We applied method (I) in our idealized model runs. This gives the correct results for the temperature and velocity fields. We checked this by comparing the PV inversion results with the initial fields in table 4.

(II) With method (II), we noticed that the temperature and velocity fields differ from the initial fields. We note that the nonlinear terms are small or at most comparable with respect to the other terms. It thus might be an alternative for the idealized model runs in chapter 9. For our case-study, we imported the total PV field from ECMWF archive data. We simply applied SGPV-inversion on the whole (scaled) PV field here.

Secondly, one has to consider which PV anomalies are advected or deformed by advection. For our theoretic model runs, we applied RK4 on the local time derivative of q'_{jet} and centered differences on the spatial derivatives of q'_{jet}. It is a simplification such that q'_{ts} remains a steady PV component describing a stratosphere at rest. Our choice to include advection of q'_{ts} is still a topic of discussion. The two discussed alternatives were:

1. We did a thought experiment on the deformation of q'_{ts} by the CFC in the advection equation. It seems more realistic when q'_{ts} (large stratification) is also advected vertically. But then q'_{ts} would deform in time and this would result in a nonlinear interaction between both PV anomalies. Moreover, we would have an attribution problem for the CFC, which has been studied in more detail by Thorpe [1997].

2. The other alternative would be to insert the total PV as a single field into the inversion relation. In the QG case, this works because the inversion equation is linear. But in the SG case, nonlinear problems arise as discussed previously. We advect the total scaled PV field for our case-study model runs.

Our next step to improve our results would be to include the advection of q'_{ts} in a dynamical consistent way.

We experienced that the SGPV-inversion routine only works in a symmetrically stable domain ($q'_{SG} + f > 0$). However, local symmetric unstable areas seem to be tolerated occasionally and more strangely, converge did not occur under some stable circumstances.

11.3 Results

Idealized model runs

Three idealized model runs were used, namely `QGsymjet`, `SGsymjet` and `SGcomjet`. The results of the first two model runs show the differences between the QG and SG shape of the CFC and their interaction with the tropopause. The results of the last two model runs show how a more realistic jet structure results in a deeper tropopause fold.

We repeat the main differences between the QG and the SG results. The most striking region is the stratosphere where the CFC is unrealistically strong. We concluded that QG theory does not 'feel' the presence of the stratification in the stratosphere. This stratification is present in the SG form of the Sawyer-Eliassen equation, resulting in more realistic reduced velocities. This difference is the result of our definition of the stratosphere, namely a PV anomaly. Furthermore, the QG model run does not show a sign of tropopause folding as expected.

The results of the idealized model runs `SGsymjet` and `SGcomjet` show the dynamical influence of the CFC. It deforms the tropopause which has descended a few hundred meters over 24 hours, which is rather short. The main reason is that we did not include the advection of q'_{ts} yet. Moreover, a stronger and more local deformation field would be more realistic and will increase the process of frontogenesis, and therefore will strengthen the CFC in time.

The idealized model runs did not gave a satisfying answer on the second research question, but it enabled us to try out several configurations and to perform a sensitivity analysis. This knowledge was of great help for interpreting the following case-study results.

Case-study

The case that we studied concerns a very deep tropopause fold over Europe from the 9th until the 11th of November. The deepest fold was observed over the Mediterranean Sea. We focused here on the shallower fold over France due to its approximate two-dimensional structure. Our hypothesis was that the tropopause fold is the result of deformation frontogenesis and resulting CFC. Our goal here was to answer the second research question and to produce a tropopause fold as realistic as possible with the model.

The cross-section data revealed some interesting features. The short-lived double fold structure showed many similarities with another case in Donnadille et al. [2001]. The deepest fold was observed when the two extensions merged. The fold extended downward to around 4 km and developed slightly under the jet stream in western direction. The curving under the jet is a clear signature of the effect of a cross-frontal circulation.

We performed a rescaling of Ertel's PV, obtained from the ECMWF operational data archive, in order to remove the dominant background stratification in the stratosphere. We chose an isothermal reference atmosphere as the scaling factor. This resulted in a scaled PV ($\hat{q}$) field in which we recognized a troposphere-stratosphere transition and a notable deep PV anomaly. An ideal input for our two-dimensional model without considering both components separately. The following results are based on a constant stretching deformation field.

1. We compared QGPV-inversion and SGPV-inversion for one specific time. We saw that from $\hat{q}_{QG}$ all basic properties of an initiating fold

could be derived. A strong(er) jet core west of the fold, an area of reduced static stability mainly in the east and a double circulation pattern. The latter results from the double front structure. However, the vertical velocities are a factor 10 too small. The SG results were similar, but showed a more local CFC with similar velocities.

2. The SG results for the 18 hour long model run are also based on a constant deformation field. But it is capable of reproducing the observed downward advection of the PV anomaly by the double CFC structure. Moreover, the western CFC seems stronger and begins to advect the PV anomaly right under the jet core, making it officially a tropopause fold after 18 hours. The fold evolves too slow compared with the ECMWF analysis as the simulated CFC is too weak.

These results are in accordance with results from other case-studies (Donnadille et al. [2001]), where stretching deformation alone could not explain the deep tropopause folds. The effect of cold wind-shear and curvature also contribute to the folding process. However, the input of a more realistic stretching deformation field from archived data seems to be promising. The first results show that stretching deformation is by far the largest contributor to the folding process over France. In southern direction, we see that the intrusion becomes deeper, the contribution from other sources larger (e.g. curvature) and the two-dimensional approximation will no longer hold.

Acknowledgements

This thesis has been the result of an inspiring year at the KNMI during my internship. The atmosphere at the weather research department was great and the informal talks and discussions were very interesting and informative. I would like to thank my supervisors Wim Verkley and Aarnout van Delden for their availability and willingness to help me whenever possible. Thanks to you, I have become even more enthusiastic for dynamical meteorology. Also, I would like to thank Emiel van der Plas for providing me with useful Python scripts. Furthermore, Thoas Hooning van Duyvenbode for making a beautiful front cover. And of course my roommates at the KNMI (under which Niels, Michou and Valentijn) and other colleagues for having a great time!

En ik wil tenslotte mijn familie, die me afgelopen studiejaren hebben bijgestaan, duizendmaal bedanken voor hun liefdevolle steun.

Appendix

A Combining stretching and shearing deformation

The stretching and shearing deformation only differ by a 45° rotation so it is convenient to combine the two terms A and F and replace them by a total deformation term. This measure of deformation will have a simple form if we use a coordinate system that is rotated (w.r.t. the original one) such that one of its axes lies along-front and the other across-front.

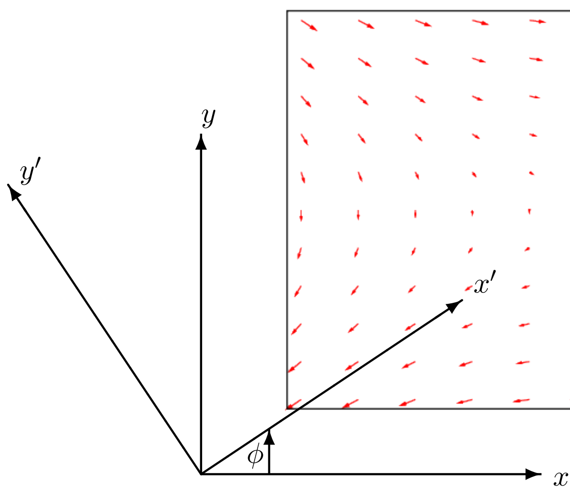


Figure 38 – Rotation of coordinate system (x, y) to (x', y') with x' being the axis of dilatation positioned such that the stretching effect is maximum.

Assume $\zeta = D = 0$. Now rotate the coordinate axis from (x, y) to (x', y') with an angle ϕ such that there is only stretching deformation A' along this axis (which is defined as the *axis of dilatation*, see figure A). The relative velocity in the (x, y) and (x', y') coordinate systems are given by:

$$\delta u = \frac{1}{2}(Ax + Fy) \qquad \delta v = \frac{1}{2}(Fx - Ay) \tag{120a}$$

$$\delta u' = \frac{1}{2}(A'x' + F'y) \qquad \delta v' = \frac{1}{2}(F'x - A'y') \tag{120b}$$

The transformation equations are given by:

$$\delta u' = \cos\phi\delta u + \sin\phi\delta v \tag{121a}$$

$$\delta v' = -\sin\phi\delta u + \cos\phi\delta v \tag{121b}$$

$$\delta u = \cos\phi \delta u' - \sin\phi \delta v' \tag{121c}$$

$$\delta v = \sin\phi \delta u' + \cos\phi \delta v' \tag{121d}$$

and similarly for $\vec{x}' \leftrightarrow \vec{x}$ (or any other vector).

Now use (121) to rewrite (120a) as follows:

$$\cos\phi\delta u' - \sin\phi\delta v' = \frac{1}{2}\left[A(\cos\phi x' - \sin\phi y') + F(\sin\phi x' + \cos\phi y')\right] \tag{122a}$$

$$\sin\phi\delta u' + \cos\phi\delta v' = \frac{1}{2}\left[F(\cos\phi x' - \sin\phi y') - A(\sin\phi x' + \cos\phi y')\right] \tag{122b}$$

Solve these equations for $\delta u'$ and $\delta v'$ by multiplying the equations with $\sin\phi$ or $\cos\phi$ and using $\sin^2\phi + \cos^2\phi = 1$. This results in:

$$\delta u' = \frac{1}{2}\left[x'(A\cos 2\phi + F\sin 2\phi) + y'(-A\sin 2\phi + F\cos 2\phi)\right] \tag{123a}$$

$$\delta v' = \frac{1}{2}\left[x'(-A\sin 2\phi + F\cos 2\phi) + y'(-A\cos 2\phi - F\sin 2\phi)\right] \tag{123b}$$

Now we can compare the last two expressions with (120b) resulting in the following equations for A' and F'.

$$A' = A\cos 2\phi + F\sin 2\phi \tag{124a}$$

$$F' = F\cos 2\phi - A\sin 2\phi \tag{124b}$$

From here we can find the following two useful equations:

$$A'^2 = A^2 + F^2 \tag{125a}$$

$$\tan 2\phi = \frac{F}{A} \tag{125b}$$

where we have assumed that $F' = 0$, i.e. ϕ is the angle of rotation from x towards the axis of dilatation x' along which there is only stretching deformation A'!

B Derivation of alternative frontogenesis function

The alternative form of the frontogenesis function will be derived here, based on notes in Smith [2007], starting from:

$$\frac{D|\nabla_h\theta|^2}{Dt} = 2\nabla_h\theta \cdot \frac{D}{Dt}\nabla_h\theta \tag{126}$$

Writing out this inner product, using that $\frac{D\theta}{Dt} = \dot{q}$, we obtain:

$$\begin{aligned}\frac{D}{Dt}|\nabla_h\theta|^2 =& 2\left[(\theta_x\dot{q}_x + \theta_y\dot{q}_y) - (\theta_x w_x + \theta_y w_y)\theta_z\right] \\ & - 2\left[(u_x\theta_x^2 + v_y\theta_y^2) + (v_x + u_y)\theta_x\theta_y\right]\end{aligned} \tag{127}$$

Now we can substitute the definitions of the derivatives of u, v in terms of D,A,F and ζ:

$$\begin{aligned}\frac{D}{Dt}|\nabla_h\theta|^2 &= 2\left[(\theta_x\dot{q}_x + \theta_y\dot{q}_y) - (\theta_x w_x + \theta_y w_y)\theta_z\right] \\ &\quad - ((D + A)\theta_x^2 + (D - A)\theta_y^2) - (F + \zeta + F - \zeta)\theta_x\theta_y \\ &= 2(\theta_x\dot{q}_x + \theta_y\dot{q}_y) - 2(\theta_x w_x + \theta_y w_y)\theta_z \\ &\quad - D(\theta_x^2 + \theta_y^2) - \left(A\theta_x^2 + 2F\theta_x\theta_y - A\theta_y^2\right)\end{aligned} \tag{128}$$

We see that the rotation part ζ drops out. It does not play a role in frontogenetic forcing directly as explained earlier. From the equation above, it follows that:

$$\begin{aligned}\frac{D}{Dt}|\nabla_h\theta| = &\left[\frac{(\theta_x\dot{q}_x + \theta_y\dot{q}_y)}{|\nabla_h\theta|}\right]_{T_1} - \left[\frac{(\theta_x w_x + \theta_y w_y)\theta_z}{|\nabla_h\theta|}\right]_{T_2} \\ & - \left[\frac{\frac{1}{2}D(\theta_x^2 + \theta_y^2)}{|\nabla_h\theta|}\right]_{T_3} - \left[\frac{\frac{1}{2}\left(A\theta_x^2 + 2F\theta_x\theta_y - A\theta_y^2)\right)}{|\nabla_h\theta|}\right]_{T_4}\end{aligned} \tag{129}$$

Now let us go through all terms and examine the underlying kinematics. Let $\hat{n}$ be the unit vector in the direction of $|\nabla_h\theta|$ and use the relation: $\hat{n}|\nabla_h\theta| = \nabla_h\theta$.

$$T_1 = \frac{\nabla_h\theta \cdot \nabla_h\dot{q}}{|\nabla_h\theta|} = \frac{\nabla_h\theta}{|\nabla_h\theta|} \cdot \nabla_h\dot{q} = \hat{n} \cdot \nabla_h\dot{q} \tag{130a}$$

$$T_2 = -\frac{(\nabla_h\theta \cdot \nabla_h w)\theta_z}{|\nabla_h\theta|} = -\left(\frac{\nabla_h\theta}{|\nabla_h\theta|} \cdot \nabla_h w\right)\theta_z = -\theta_z\hat{n} \cdot \nabla_h w \tag{130b}$$

$$T_3 = -\frac{\frac{1}{2}D|\nabla_h\theta|^2}{|\nabla_h\theta|} = -\frac{1}{2}D|\nabla_h\theta| \tag{130c}$$

$$T_4 = -\frac{\frac{1}{2}\left[A\theta_x^2 + 2F\theta_x\theta_y - A\theta_y^2)\right]}{|\nabla_h\theta|} \tag{130d}$$

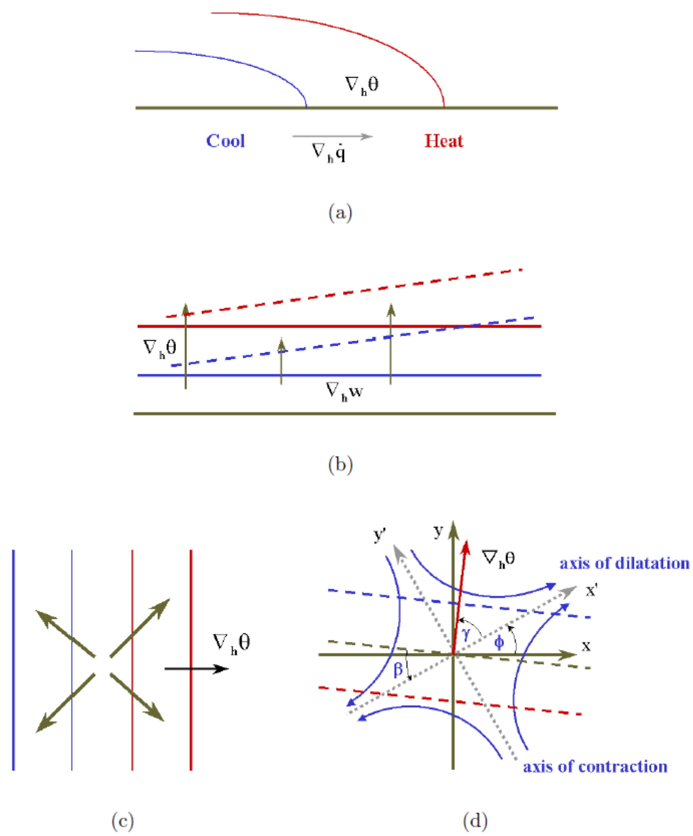


Figure 39 – Sketches of four frontogenetic processes: (a) differential diabatic heating in the direction of the thermal gradient, (b) differential vertical motion acting on a vertical temperature field, (c) convergence in an existing horizontal temperature field and (d) the interplay between a horizontal deformation field and temperature field. Source: Smith [2007]

Examples of differential diabatic heating T_1 are differences in latent heat release (frontogenetic), differences in heat capacity (e.g. a front over land-sea surface) and cloud formation. The tilting effect T_2 was already discussed and only plays a role in middle (upper) troposphere where vertical velocities (vertical gradient in temperature) are largest. Convergence on an existing thermal gradient T_3 is characterized by shrinking of isotherms in all directions. The direction of the thermal gradient is not important here. It is of importance in T_4 representing deformation effects by both stretching A and shear F.

We will now neglect T_1 and T_2 and combine T_3 and T_4. Term T_4 will be rewritten in a simpler way:

First, we transform the derivatives in the (x, y) coordinate system to (x', y') derivatives with (x', y') being the coordinate system with an axis of dilatation and axis of contraction:

$$\theta_x = \cos\phi\theta'_x - \sin\phi\theta'_y \tag{131a}$$

$$\theta_y = \sin\phi\theta'_x + \cos\phi\theta'_y \tag{131b}$$

and substitute these in the equation for T_4, (130d):

$$\begin{aligned}
T_4 =& \frac{-1}{2|\nabla_h\theta|}\left[A(\cos\phi\theta'_x - \sin\phi\theta'_y)^2 + 2F(\cos\phi\theta'_x - \sin\phi\theta'_y)(\sin\phi\theta'_x + \cos\phi\theta'_y)\right.\\
&\left.-A(\sin\phi\theta'_x + \cos\phi\theta'_y)^2\right]\\
=& \frac{-1}{2|\nabla_h\theta|}\left[A\cos^2\phi\theta'^2_x + A\sin^2\phi\theta'^2_y - 2A\cos\phi\sin\phi\theta'_x\theta'_y\right.\\
&+ 2F\cos\phi\sin\phi\theta'^2_x + 2F\cos^2\phi\theta'_x\theta'_y - 2F\sin^2\phi\theta'_x\theta'_y - 2F\cos\phi\sin\phi\theta'^2_y\\
&\left.-A\sin^2\phi\theta'_x - A\cos^2\phi\theta'^2_y - 2A\cos\phi\sin\phi\theta'_x\theta'_y\right]\\
=& \frac{-1}{2|\nabla_h\theta|}\left[A(\theta'^2_x - \theta'^2_y)(\cos^2\phi - \sin^2\phi) - 4A\cos\phi\sin\phi\theta'_x\theta'_y\right.\\
&\left.+2F\cos\phi\sin\phi(\theta'^2_x - \theta'^2_y) + 2F\theta'_x\theta'_y(\cos^2\phi - \sin^2\phi)\right]\\
=& \frac{-1}{2|\nabla_h\theta|}\left[A(\theta'^2_x - \theta'^2_y)\cos 2\phi - 2A\sin 2\phi\theta'_x\theta'_y\right.\\
&\left.+F\sin 2\phi(\theta'^2_x - \theta'^2_y) + 2F\theta'_x\theta'_y\cos 2\phi\right]\\
=& \frac{-1}{2|\nabla_h\theta|}\left[(\theta'^2_x - \theta'^2_y)(A\cos 2\phi + F\sin 2\phi)\right.\\
&\left.+2\theta'_x\theta'_y(F\cos 2\phi - A\sin 2\phi)\right]
\end{aligned} \tag{132}$$

Now use the definitions of $A' = A\cos 2\phi + F\sin 2\phi$ and $F = A\tan 2\phi$ (see

Appendix A)

$$\begin{aligned} T_4 =& \frac{-1}{2|\nabla_h\theta|}\left[(\theta_x'^2-\theta_y'^2)A' + 2\theta_x'\theta_y'(A\sin 2\phi - A\sin 2\phi)\right] \\ =& \frac{-A'}{2|\nabla_h\theta|}(\theta_x'^2-\theta_y'^2) \end{aligned} \tag{133}$$

Next, we perform another coordinate transformation from (x', y') to $(\hat{n}, \hat{tr})$ where $\hat{n}$ is the unit vector in the direction of $\nabla_h\theta$ and $\hat{tr}$ in the perpendicular direction. Now let's calculate the components of $\hat{n}$ by performing the transformation:

$$\frac{\partial\theta}{\partial x'} = \frac{\partial n}{\partial x'}\frac{\partial\theta}{\partial n} = \cos\gamma|\nabla_h\theta| \tag{134a}$$

$$\frac{\partial\theta}{\partial y'} = \frac{\partial n}{\partial y'}\frac{\partial\theta}{\partial n} = \sin\gamma|\nabla_h\theta| \tag{134b}$$

where $\frac{\partial\theta}{\partial tr}$ is zero by definition such that $\frac{\partial\theta}{\partial n} = |\nabla_h\theta|$ and γ is the angle from the axis of dilatation x' to the thermal gradient axis $\hat{n}$ (see figure 39). Now we can write $(\theta_x'^2 - \theta_y'^2)$ as:

$$\theta_x'^2 - \theta_y'^2 = \cos^2\gamma|\nabla_h\theta|^2 - \sin^2\gamma|\nabla_h\theta|^2 = \cos 2\gamma|\nabla_h\theta|^2 \tag{135}$$

and T_4 can be written as:

$$\begin{aligned} T_4 =& \frac{-A'}{2}|\nabla_h\theta|\cos 2\gamma \\ =& \frac{-A'}{2}|\nabla_h\theta|\cos\left(2(\frac{1}{4}\pi - \phi)\right) \\ =& \frac{A'}{2}|\nabla_h\theta|\cos 2\phi \end{aligned} \tag{136}$$

where A' is the total deformation which is purely stretching deformation in coordinate system (x', y'), ϕ is the angle from the axis of dilatation to the isotherms (or $\hat{tr}$ where $\nabla_h\theta = 0$).

If T_3 is included to the expression for T_4 then the final deformation form of the frontogenesis function becomes:

$$\frac{D|\nabla_h\theta|^2}{Dt} = |\nabla_h\theta|^2(A'\cos 2\phi - D) \tag{137}$$

where we recall that D represents horizontal divergence.

C Basics of SOR theory

We demonstrate here the basics behind the Successive Over-Relaxation method using our simplest elliptical equation. This is the QG form of the Sawyer-Eliassen equation given by (28):

$$N^2\psi_{yy} + f^2\psi_{zz} = N_2(\psi) = 2Q_2 \tag{138}$$

where N_2 is a linear operator on ψ. ψ can be derived by solving the problem $\psi = N_2^{-1}(2Q_2)$

This equation can be interpreted as a final equilibrium state of the following partial differential equation:

$$\frac{\partial\psi}{\partial t} = N^2\psi_{yy} + f^2\psi_{zz} - 2Q_2 \tag{139}$$

In other words, an initial well-chosen solution *relaxes* towards the equilibrium state ($\frac{\partial\psi}{\partial t} = 0$) given in (138) for $t \to \infty$.

Jacobi's method

We start by applying Forward-Time-Centered-Space (hereafter FTCS) differencing on (139):

$$\psi_{j,l}^{r+1} = \psi_{j,l}^{r} + \Delta t\left[N^2\frac{\psi_{j+1,l}^{r} - 2\psi_{j,l}^{r} + \psi_{j-1,l}^{r}}{h^2} + f^2\frac{\psi_{j,l+1}^{r} - 2\psi_{j,l}^{r} + \psi_{j,l-1}^{r}}{h^2} - 2Q_2\right] \tag{140}$$

where h is the grid size in both horizontal and vertical direction, r is used for iteration steps in the relaxation process ('time' steps), j for discrete horizontal steps and l for discrete vertical steps. The same indices are used in the code.

It can be shown from numerical stability analysis that FCTS is stable in two-dimensional space when $\Delta t/h^2 \leq \frac{1}{4}$. Now if we choose the largest possible 'time' step $\Delta t = \frac{1}{4}h^2$ than the equation above can be rewritten as:

$$\begin{aligned}\psi_{j,l}^{r+1} =& \psi_{j,l}^{r} + \frac{1}{4}\left[N^2(\psi_{j+1,l}^{r} - 2\psi_{j,l}^{r} + \psi_{j-1,l}^{r}) + \right.\\ &\left. f^2(\psi_{j,l+1}^{r} - 2\psi_{j,l}^{r} + \psi_{j,l-1}^{r}) - \frac{1}{2}h^2Q_2\right]\end{aligned} \tag{141}$$

If $N^2 = f^2 = 1$ for simplicity (to show the basic idea of relaxation):

$$\psi_{j,l}^{r+1} = \frac{1}{4}(\psi_{j+1,l}^{r} + \psi_{j-1,l}^{r} + \psi_{j,l+1}^{r} + \psi_{j,l-1}^{r}) - \frac{1}{2}h^2Q_2 \tag{142}$$

then we can directly see that the ψ field at the 'r'th iteration step is calculated by taking the spatial average (first term) plus a contribution from the source (second term). The procedure is repeated until convergence occurs. This is Jacobi's relaxation method. It converges too slowly to be practically useful.

Let us now rewrite the analysis above by splitting up the matrix operator N_2. The same approach in terms of matrices is eventually applied in the SOR method. In matrix terms, the following equation has to be solved:

$$\mathbf{A} \cdot \psi = \mathbf{b} \tag{143}$$

where we use standard matrix notations $\mathbf{A}$ for linear operator N_2 and $\mathbf{b}$ for the forcing $2Q_2$. Now we can split the matrix operator $\mathbf{A}$ into three parts:

$$\mathbf{A} = \mathbf{L} + \mathbf{D} + \mathbf{U} \tag{144}$$

where $\mathbf{L}$ is the lower triangle of $\mathbf{A}$, $\mathbf{U}$ the upper triangle and $\mathbf{D}$ the diagonal. This matrix splitting can be used to write the Jacobi's method as:

$$\mathbf{D} \cdot \psi^{(r)} = -(\mathbf{L} + \mathbf{U}) \cdot \psi^{(r-1)} + \mathbf{b} \tag{145}$$

Convergence occurs for diagonally dominant matrices. This means that we are in a situation close to equilibrium given by eq. (143). The diagonal part $\mathbf{D}$ on the lhs is dominant. The first term on the rhs, $(\mathbf{L} + \mathbf{U})$, corresponds to deviations from equilibrium and is relatively small.

The convergence of this method can be investigated by the following relation which gives the number of iterations r needed to reduce the overall error by a factor 10^{-p}:

$$r \approx \frac{p \ln 10}{-\ln \rho_{Jac}} \tag{146}$$

where ρ_{Jac} is the so-called spectral radius for the Jacobi method. It is the modulus (between 0 and 1) of the slowest decaying eigenmode which corresponds to the largest rate of convergence. We take $p = 3$ such that an error smaller than 10^{-3} is the end point of the iteration loop.

The general two-dimensional expression for ρ_{Jac} is given by:

$$\rho_{Jac} = \frac{\cos\frac{\pi}{J} + \left(\frac{\Delta y}{\Delta z}\right)^2 \cos\frac{\pi}{L}}{1 + \left(\frac{\Delta y}{\Delta z}\right)^2} \tag{147}$$

The value of ρ_{Jac} goes asymptotically to 1 as the grid size increases to infinity. In our case, we use very large grids ($J = L = 201$ grid points) and we can simplify (147) to:

$$\rho_{Jac} \approx \cos\left(\frac{\pi}{J}\right) \approx 1 - \frac{\pi^2}{2J^2} \tag{148}$$

after performing a Taylor expansion. The number of iterations r given in (146) finally becomes:

$$r \approx \frac{1}{2} p J^2 \tag{149}$$

thus the number of iterations increases by the number of grid points *squared*. For our model this would yield $r \approx 0.5 * 3 * 200^2 = 60.000!$ iteration steps in order to evaluate the cross-frontal circulation, which takes far too much time.

SOR

A better algorithm can be devised by the implementation of an overcorrection in such a way that we anticipate on future corrections towards equilibrium. We would like to implement such a correction in the following way:

$$\psi_{new} = \psi_{old} - \omega * f(\psi_{old}) \tag{150}$$

such that (i) the solution converges towards a new ψ_{new} by the physical processes captured inside a function of the old ψ_{old} which are overcorrected by the overrelaxation parameter ω.

It is easy to rewrite (145) in the form of (150). We obtain:

$$\psi^{(r)} = \psi^{(r-1)} - (\mathbf{D})^{-1} \cdot \left[(\mathbf{U} + \mathbf{L} + \mathbf{D}) \cdot \psi^{(r-1)} - \mathbf{b}\right] \tag{151}$$

where the term between brackets is the residual vector $\xi^{(r-1)}$, i.e. we may write:

$$\psi^{(r)} = \psi^{(r-1)} - (\mathbf{D})^{-1} \cdot \xi^{(r-1)} \tag{152}$$

Now we include the overcorrection ω:

$$\psi^{(r)} = \psi^{(r-1)} - \omega(\mathbf{D})^{-1} \cdot \xi^{(r-1)} \tag{153}$$

The following holds:

1. Only overrelaxation ($1 < \omega < 2$) gives faster convergence than the Jacobian method

2. If ρ_{Jac} is the spectral radius of the Jacobi iteration, then the *optimal* choice for ω is given by:

$$\omega = \frac{2}{1 + \sqrt{1 - \rho_{Jac}^2}} \tag{154}$$

3. The number of iterations to reduce error by a factor 10^{-p} is now given by:

$$r = \frac{1}{3} pJ \tag{155}$$

The third statement means that for our model we only need $r \approx 0.33 * 3 * 200 = 200$ iteration steps! We conclude that we have to choose optimal values for ρ_{Jac} and ω in order to gain faster convergence. The optimal values are calculated in code lines 902 - 906

D From scaled to balanced PV

It will be shown here that if we scale Ertel's PV properly by $S_{ref}(\theta)$ corresponding to an isothermal reference atmosphere, we can derive an equation for the scaled PV which has a similar shape as the semi-geostrophic PV.

We start with $\hat{q}$, given in (119):

$$\hat{q} = \frac{-g(\zeta + f - \frac{\partial v}{\partial \theta}\frac{\partial \theta}{\partial x} + \frac{\partial u}{\partial \theta}\frac{\partial \theta}{\partial y})\frac{\partial \theta}{\partial p}}{\frac{gT_0\kappa}{p_r}\left(\frac{\theta}{T_0}\right)^{\frac{\kappa+1}{\kappa}}} \tag{156}$$

which can be rewritten as:

$$\hat{q} = -\left(\zeta + f - \frac{\partial v}{\partial \theta}\frac{\partial \theta}{\partial x} + \frac{\partial u}{\partial \theta}\frac{\partial \theta}{\partial y}\right)\frac{\partial \theta}{\partial p}\frac{p_r}{T_0\kappa}\left(\frac{\theta}{T_0}\right)^{-1-\frac{1}{\kappa}} \tag{157}$$

The second factor can be written as a derivative with respect to p (like factor II in Ertel's PV).

$$\begin{aligned}
\frac{\partial \theta}{\partial p}\frac{p_r}{T_0\kappa}\left(\frac{\theta}{T_0}\right)^{-1-\frac{1}{\kappa}} &= \frac{\partial}{\partial p}\left(\theta\frac{p_r}{T_0\kappa}\left(\frac{\theta}{T_0}\right)^{-1-\frac{1}{\kappa}}\right) \\
&\quad - \theta\frac{p_r}{T_0\kappa}\frac{\partial}{\partial p}\left[\left(\frac{\theta}{T_0}\right)^{-1-\frac{1}{\kappa}}\right] \\
&= \frac{\partial}{\partial p}\left(\frac{p_r}{\kappa}\left(\frac{T_0}{\theta}\right)^{\frac{1}{\kappa}}\right) \\
&\quad - \frac{\theta p_r}{T_0\kappa}\left(-1 - \frac{1}{\kappa}\right)\frac{1}{T_0}\left(\frac{\theta}{T_0}\right)^{-2-\frac{1}{\kappa}}\frac{\partial \theta}{\partial p} \\
&= \frac{\partial}{\partial p}\left(\frac{p_r}{\kappa}\left(\frac{T_0}{\theta}\right)^{\frac{1}{\kappa}}\right) \\
&\quad + \frac{p_r}{T_0\kappa}\left(1 + \frac{1}{\kappa}\right)\left(\frac{\theta}{T_0}\right)^{-1-\frac{1}{\kappa}}\frac{\partial \theta}{\partial p}
\end{aligned} \tag{158}$$

The term on the lhs now also appears on the rhs. Subtracting this term from the whole equation and multiplying by κ results in:

$$-\frac{p_r}{T_0\kappa}\left(\frac{\theta}{T_0}\right)^{-1-\frac{1}{\kappa}}\frac{\partial \theta}{\partial p} = \frac{\partial}{\partial p}\left(p_r\left(\frac{T_0}{\theta}\right)^{\frac{1}{\kappa}}\right) \tag{159}$$

such that (157) simplifies to

$$\hat{q} = \frac{\partial r}{\partial p}\left(\zeta + f - \frac{\partial v}{\partial \theta}\frac{\partial \theta}{\partial x} + \frac{\partial u}{\partial \theta}\frac{\partial \theta}{\partial y}\right) \tag{160}$$

where r is defined as:

$$r = p_r\left(\frac{T_0}{\theta}\right)^{\frac{1}{\kappa}} \tag{161}$$

We see that when we use $\theta = T(p_r/p)^\kappa$, we can also write:

$$r = p\left(\frac{T_0}{T}\right)^{\frac{1}{\kappa}} \tag{162}$$

Thus, $r = p$ for an isothermal atmosphere, in which case $\hat{q}$ does not contain a stratification term at all.

Let us assume that the isothermal atmosphere is a good approximation and that deviations from it are small. We can split up r in a main and perturbation part, i.e. $r = \overline{r} + r'$ where $\overline{r} = \overline{p}$ and $r' \ll \overline{r}$. r' is calculated using a Taylor expansion around the reference state $\overline{\theta}$ by:

$$\begin{aligned} r &\approx \overline{p} + \left.\frac{\partial r}{\partial \theta}\right|_{\overline{\theta}}(\theta - \overline{\theta}) \\ &\approx \overline{p} - p_r\frac{T_0}{\kappa\overline{\theta}^2}\left(\frac{T_0}{\overline{\theta}}\right)^{\frac{1}{\kappa}-1}\theta' \\ &\approx \overline{p} - \frac{p_r}{\kappa}\left(\frac{T_0}{\overline{\theta}}\right)^{\frac{1}{\kappa}}\frac{\theta'}{\overline{\theta}} \\ &\approx \overline{p} - \frac{p_r}{\kappa}\left(\frac{T_0}{T_0\left(\frac{p_r}{\overline{p}}\right)^\kappa}\right)^{\frac{1}{\kappa}}\frac{\theta'}{\overline{\theta}} \\ &\approx \overline{p}\left(1 - \frac{1}{\kappa}\frac{\theta'}{\overline{\theta}}\right) \end{aligned} \tag{163}$$

We now expressed r in terms of temperature deviations from an isothermal reference profile. For the vertical derivative with respect to p, we then have:

$$\begin{aligned}
\frac{\partial r}{\partial p} &\approx \frac{\partial \overline{p}}{\partial p} - \frac{\partial \overline{p}}{\partial p}\frac{1}{\kappa}\frac{\theta'}{\overline{\theta}} - \frac{\overline{p}}{\kappa}\frac{\partial}{\partial p}\left(\frac{\theta'}{\overline{\theta}}\right) \\
&\approx \frac{\partial \overline{p}}{\partial p} - \frac{\partial \overline{p}}{\partial p}\frac{1}{\kappa}\frac{\theta'}{\overline{\theta}} - \frac{1}{\kappa}\frac{\partial}{\partial p}\left(\overline{p}\frac{\theta'}{\overline{\theta}}\right) + \frac{\partial \overline{p}}{\partial p}\frac{1}{\kappa}\frac{\theta'}{\overline{\theta}} \\
&\approx \frac{\partial \overline{p}}{\partial p} - \frac{1}{\kappa}\frac{\partial z}{\partial \overline{p}}\frac{\partial \overline{p}}{\partial p}\frac{\partial}{\partial z}\left(\overline{p}\frac{\theta'}{\overline{\theta}}\right) \\
&\approx \frac{\partial \overline{p}}{\partial p}\left[1 + \frac{1}{\kappa}\frac{1}{\overline{\rho} g}\frac{\partial}{\partial z}\left(\overline{p}\frac{\theta'}{\overline{\theta}}\right)\right] \\
&\approx \frac{\partial \overline{p}}{\partial p}\left[1 + \frac{1}{\kappa \overline{\rho} g}\frac{\partial}{\partial z}\left(\overline{\rho} R T_0 \frac{\theta'}{\overline{\theta}}\right)\right] \\
&\approx \frac{\partial \overline{p}}{\partial p}\left[1 + \frac{R T_0}{\kappa \overline{\rho} g^2}\frac{\partial}{\partial z}\left(g\overline{\rho}\frac{\theta'}{\overline{\theta}}\right)\right]
\end{aligned} \tag{164}$$

We can use the definitions for buoyancy $b = g\frac{\theta'}{\overline{\theta}}$ and for static stability:

$$N^2 = \frac{g}{\overline{\theta}}\frac{\partial \overline{\theta}}{\partial z} = \frac{g}{\overline{\theta}}\frac{\partial \overline{\theta}}{\partial \overline{p}}\frac{\partial \overline{p}}{\partial z} = \frac{g}{\overline{\theta}}\left(-\frac{\kappa}{\overline{p}}\overline{\theta}\right)(-\overline{\rho} g) = \frac{g^2 \kappa}{R T_0} \tag{165}$$

such that (164) becomes:

$$\frac{\partial r}{\partial p} \approx \frac{\partial \overline{p}}{\partial p}\left[1 + \frac{1}{N^2 \overline{\rho}}\frac{\partial}{\partial z}(g\overline{\rho} b)\right] \tag{166}$$

Next, we neglect the vertical dependence of $\overline{\rho}$:

$$\frac{\partial r}{\partial p} \approx \frac{\partial \overline{p}}{\partial p}\left[1 + \frac{1}{N^2}\frac{\partial b}{\partial z}\right] \tag{167}$$

Returning to our expression for $\hat{q}$, we substitute the result above into (160):

$$\hat{q} = \frac{\partial \overline{p}}{\partial p}\left(1 + \frac{1}{N^2}\frac{\partial b}{\partial z}\right)\left(\zeta + f - \frac{\partial v}{\partial \theta}\frac{\partial \theta}{\partial x} + \frac{\partial u}{\partial \theta}\frac{\partial \theta}{\partial y}\right) \tag{168}$$

We can split the four terms $\frac{\partial v}{\partial \theta}$, $\frac{\partial \theta}{\partial x}$, $\frac{\partial u}{\partial \theta}$ and $\frac{\partial \theta}{\partial y}$ into main parts ($\overline{\theta}$) and perturbation parts (b).

Starting with $\frac{\partial v}{\partial \theta}$:

$$\frac{\partial v}{\partial \theta} = \frac{\partial v}{\partial z}\frac{\partial z}{\partial \overline{\theta}}\frac{\partial \overline{\theta}}{\partial \theta} = \frac{\partial v}{\partial z}\frac{g}{\overline{\theta} N^2}\frac{\partial \overline{\theta}}{\partial \theta} \tag{169}$$

For $\frac{\partial\theta}{\partial x}$:

$$\frac{\partial\theta}{\partial x} = \frac{\partial\overline{\theta}}{\partial x} + \frac{\partial\theta'}{\partial x} = \frac{\overline{\theta}}{g}\frac{\partial b}{\partial x} \tag{170}$$

And similarly for the other product term:

$$\frac{\partial u}{\partial\theta} = \frac{\partial u}{\partial z}\frac{g}{\overline{\theta}N^2}\frac{\partial\overline{\theta}}{\partial\theta} \tag{171}$$

$$\frac{\partial\theta}{\partial y} = \frac{\overline{\theta}}{g}\frac{\partial b}{\partial y} \tag{172}$$

As a result, the scaled PV begins to have a familiar form:

$$\hat{q} = \frac{\partial\overline{p}}{\partial p}\left(1 + \frac{1}{N^2}\frac{\partial b}{\partial z}\right)\left[\zeta + f - \frac{\partial\overline{\theta}}{\partial\theta}\frac{1}{N^2}\frac{\partial v}{\partial z}\frac{\partial b}{\partial x} + \frac{\partial\overline{\theta}}{\partial\theta}\frac{1}{N^2}\frac{\partial u}{\partial z}\frac{\partial b}{\partial y}\right] \tag{173}$$

If we substitute Ψ_g into (173) where $\left(\frac{\partial\Psi_g}{\partial x}, \frac{\partial\Psi_g}{\partial y}, \frac{\partial\Psi_g}{\partial z}\right) \equiv \left(v_g, -u_g, \frac{b}{f}\right)$:

$$\begin{aligned}\hat{q} =& \frac{\partial\overline{p}}{\partial p}\left(1 + \frac{f}{N^2}\frac{\partial^2\Psi_g}{\partial z^2}\right)\\ &\left[\nabla^2\Psi_g + f - \frac{\partial\overline{\theta}}{\partial\theta}\frac{f}{N^2}\left(\frac{\partial^2\Psi_g}{\partial x\partial z}\right)^2 - \frac{\partial\overline{\theta}}{\partial\theta}\frac{f}{N^2}\left(\frac{\partial^2\Psi_g}{\partial y\partial z}\right)^2\right]\end{aligned} \tag{174}$$

The two-dimensional version of (174) almost equals the SGPV equation (52) that we use. The only final assumptions to be made are:

$$\frac{\partial\overline{p}}{\partial p} = \frac{\partial\overline{\theta}}{\partial\theta} = 1 \tag{175}$$

This is not straightforward. We will show the required conditions here:

$$\frac{\partial\overline{p}}{\partial p} = \frac{\frac{\partial\overline{p}}{\partial z}}{\frac{\partial\overline{p}}{\partial z} + \frac{\partial p'}{\partial z}} = \frac{1}{1 - \frac{1}{\rho g}\frac{\partial p'}{\partial z}} \tag{176}$$

$$\frac{\partial\overline{\theta}}{\partial\theta} = \frac{\frac{\partial\overline{\theta}}{\partial z}}{\frac{\partial\overline{\theta}}{\partial z} + \frac{\partial\theta'}{\partial z}} = \frac{1}{1 + \frac{1}{N^2}\frac{\partial b}{\partial z}} \tag{177}$$

where in (177), we approximated the definitions for static stability and buoyancy by $N^2 = \frac{g}{\theta_0}\frac{\partial\overline{\theta}}{\partial z}$ (instead of (165)) and $b = g\frac{\theta'}{\theta_0}$.

Thus, as long as the vertical differences in pressure and potential temperature are small compared to the main state (isothermal reference profile), we can state that the scaled PV is analogous to the semi-geostrophic PV formulation. Mesoscale phenomena such as tropopause folds and intense fronts are in a gray zone where $\hat{q}$ and q_{SG} start to differ from each other.

E User tutorial

It was mentioned in the introduction of chapter 7 that the user can choose between three types of initial PV configurations: (a) initial jet/front system, (b) initial PV anomaly and (c) PV field from archived ECMWF data. We will give a short tutorial here on how a model run is started for the first two configurations. The code to handle (c), a PV field from archived ECMWF data, is very specific and shall not be considered in the tutorial.

Starting a model run

The program is built in Fortran90 and a Fortran compiler compatible with Fortran90 code has to be installed in your system. The Fortran compiler 'g95' is used here. The code is spread over three scripts, namely:

(i) PVinversion_main_v1.0.f90

(ii) PVinversion_modules.f90

(iii) PVinversion_numroutines.f90

Script (i) is the heart of the program with references to subroutines and functions defined in modules. These modules are combined into script (ii). Script (iii) contains several standard numerical routines that we use here (Successive Over-Relaxation and Runge-Kutta 4th order). The code is given in the Appendix for all scripts.

Once these three scripts are in the same file, a model run can start for the initial 'jet/front system' or 'PV anomaly' model configurations. Now go inside your terminal environment to the directory where the scripts are stored and compile the program as follows (in the right order!):

```
g95 PVinversion_numroutines.f90 PVinversion_modules.f90
 PVinversion_main_v1.0.f90
```

An error will be given when the program is compiled for the first time in a directory. In that case, compile the program again and it will work. The output after compilation is written to a file named 'a.out'. Now start a model run with the following command:

```
./a.out
```

The 'Input_user()' subroutine is called first and asks several input from the user on the command line. Follow the given instructions. The following three choices have to be made:
(I) Model configuration:

```
>> Please choose your initial problem to simulate:
1. Jet-front system     (type "jet")
2. Symmetric PV anomaly (type "ano")
3. Import PV field      (type "imp")
```

(II) Balanced theory:

```
>> Please choose between:
1. Quasi-Geostrophic balance (typ "QG")
2. Semi-Geostrophic balance  (typ "SG")
```

(III) And finally, including a stratosphere anomaly or not:

```
>> Would you like to include a stratosphere? (typ "y" or "n")
```

After that, the model will start the run. During the run, output will be generated and written to two data files. The two data files are available after the model run and have the following names:

```
Tropofold_PVfield_ani.dat
Tropofold_PVresult_ani.dat
```

The first file contains only PV values for every coordinate (y, z) and at every timestep t:

t	y	z	PV
...	...	...	...

The second file contains all other variables which are derived diagnostically from a given PV field. These are:

t	y	z	Ψ_g	b	u_g	$2Q$	ψ	v_a	w_a	Θ	TWB	N^2_{eff}	F^2	S^2
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

Notice how the format of data files look like (see example on next page). A single white space is used to separate between horizontal coordinates. Double white spaces are used to separate between timesteps. This data format is recognized by plotting software such as GNUplot if one wants to make a gif animation from initial to final timestep.

```
# t   y   z   value
  0   0   0   1
  0   0   1   3
  0   0   2   8

  0   1   0   2
  0   1   1   4
  0   1   2   9

  0   2   0   3
  0   2   1   5
  0   2   2   10


  1   0   0   2
  1   0   1   6
  1   0   2   16

  1   1   0   4
  1   1   1   8
  1   1   2   18

  ...
```

Model configuration

The script file PVinversion_modules.f90 consists of three modules. The first module (`Inversion_globalvar`) contains all global variables. The second module (`Inversion_support`) contains all secundary subroutines and functions. These are called inside the main program (PVinversion_main_v1.0.f90) and in the more general third module (`Inversion_general`) containing the input and initialisation procedure.

The first module is most important for the user. The values of global parameters can be changed inside the comment block found in the code lines 110 - 128. For example, by changing the reference atmosphere parameters (N, f, T_0, etc.), one can perform a sensitivity analysis. By changing the resolution and domain size, one can adjust the accuracy and the influence of the boundary on the results.

Plotting the data

The user can now choose what software he or she wants to use to plot the data from the two data files. During our research, we used GNUplot scripts to plot the data for one particular time step (as illustrated throughout this thesis). Moreover, shell scripts were made to plot the data for a *sequence* of time steps. These are stored as GIF animation files.

The shell and GNUplot scripts are not given in the Appendix, but the reader can contact the author or one of the supervisors to receive all scripts that we used. The software package includes:

- An user guide including a manual for the shell and GNUplot scripts.
- The PV-inversion program consisting of three Fortran 90 scripts. These scripts may be updated to a newer version with respect to the code given in the Appendix F.
- Several examples of shell scripts for making GIF animation files from the data sets.
- Several examples of GNUplot scripts for making high resolution EPS files from the data sets.
- A few GIF examples illustrating frontogenesis and tropopause folding for a period of 24 hours in 24 frames.

F Fortran90 code

PVinversion_main_v1.0.f90

```
!------------------------------------------------------ PV-inversion program
------------------------------------------------------------

!This program is used for studying atmospheric dynamic processes using the concept of 'PV thinking'

PROGRAM PV_inversion
USE Inversion_general
!Modules included in main_PV:
!- Inversion_globalvar
!- Inversion_support  (supporting subroutines and functions)
!- Nummethods (SOR and rk4 methods)
IMPLICIT NONE

!-------------------------------------------------------------------------------------------
!...............................Preparation............................
!-------------------------------------------------------------------------------------------

!Interaction with user who chooses the following:
!1. Choose reference atmosphere       (a. Isothermal (not included), b. linear stratified)
!2. PV perturbations:
! Initial configuration:              (a. Jet/front system, b. symmetric PV anomaly, c. PV imported)
! Balanced theory:                    (a. Quasi-geostrophic, b. semi-geostrophic)
! Include stratosphere (y/n)?
!3. Deformation field                 (a. Stretching deformation, other fields not included)
CALL Input_user()

!---------------------------------------------------------------------------------
!-------------------------------- Algorithm -----------------------------------

! -------------------------- Order of calculations: --------------------------
! Vg,va,wa --> PV --> gphi --> Ug, buoy, FF --> N2,F2,S2 --> phi --> va,wa -->
! -------------------------------------------------------------------------------

DO it=0,t1,dt
  IF (it==0) THEN
  !initialisation of variables depending on chosen model configuration
  CALL Initialisation()
  ELSE
  !Calculate new PV field at next timestep from PV conservation
    !Using Runge-Kutta 4th order
  CALL rk4_PVfield()
    !also on surface boundary
  CALL rk4_PVsurface()
    !preparations for new time step (print-to-file settings, boundary conditions)
  CALL Initialisation_dt()
  ENDIF

  !PV-inversion (QG & SG) using Successive Over-Relaxation (SOR)
  SELECT CASE (SG)
  CASE (.TRUE.)  !Semi-geostrophic case
    !Including surface and lateral boundary conditions
    !Nonlinear inversion
    IF (inimodel.NE.'imp') THEN
      !>>> Perform PV-inversion for each anomaly separately and sum up gphi
      !stratosphere anomaly
    CALL SORrelaxation_PVSG(PV_ts,PVFF_surface*0.d0,PVFF_lateral*0.d0,gphi_ts)
      !dynamic PV_anomaly
    CALL SORrelaxation_PVSG(PV-PV_ts,PVFF_surface,PVFF_lateral,gphi_ano)
    gphi = gphi_ano + gphi_ts
    ELSE
      !Imported SG anomalies cannot be separated: perform PV-inversion for whole field
    CALL SORrelaxation_PVSG(PV-f,PVFF_surface,PVFF_lateral,gphi)
    ENDIF
  CASE (.FALSE.) !Quasi-geostrophic case
    !Linear inversion
    !Including surface and lateral boundary conditions
  CALL SORrelaxation_BC(PVFF,PVFF_surface,PVFF_lateral,gphi)
  END SELECT

  !Calculate new geostrophic velocity, buoyancy and frontogenetic forcing fields from gphi
  CALL PV_circulation(buoy,U,FF,gphi)

  !New frequencies N^2, F^2, S^4 for SG case, for QG case they remain constant
```

```
    IF (SG) CALL new_frequencies()

    !Use SOR again to determine ageostrophic streamfunction
    !Linear operator for both QG and SG
    !Simple zero-valued boundary conditions
    CALL SORrelaxation(N2,F2,S2,FF,phi)

    !finally, determine ageostrophic velocity field
    CALL ageo_circulation()

    !write resulting fields to file
    IF (outputok) CALL results_to_file()
  END DO

!--------------End of program-------------------
END PROGRAM PV_inversion


!Made by:                    Marten Blaauw
!For comments/questions:     mcblaauw@gmail.com
!Date of last update:        14 Oktober 2011
```

PVinversion_modules.f90

```fortran
!QG and SG case, stratosphere or no stratosphere using PV notation
!SG case solved WITHOUT transformation, including an extra SOR iteration step to approach gphi from SGPV

!v1: input of external fields from txt files. Q, PV and background T fields are from ECMWF data


!----------------------------------------------------------------------------------------------------------------
!------------------------------------------------------- Global Variables ---------------------------------------
!----------------------------------------------------------------------------------------------------------------



MODULE Inversion_globalvar
 IMPLICIT NONE
 SAVE

 !----------------------------------------------------User settings-----------------------------------------------
  !stratosphere?, semi-geostrophic theory?, print-to-file?
 LOGICAL :: stratos, SG, outputok=.TRUE.
  !Configuration: Jet/front system, symmetric PV anomaly or imported PV field
 CHARACTER(LEN=3) :: inimodel

  !#gridpoints y and z, domain size y and z, timestep, endtime, print-to-file interval,
 INTEGER, PARAMETER :: ygrid=201,zgrid=201,domain_y=5d6,domain_z=5e4,dt=300,t1=86400,intervalit=3600
  !Constants of nature
 DOUBLE PRECISION, PARAMETER :: pi=3.14159, g=9.81, R=286.9d0, cp=1005.d0
  !Coriolis parameter, surface temperature, static frequency reference atmosphere, surface pressure, height scale of troposphere-stratosphere transition
 DOUBLE PRECISION, PARAMETER :: f=1.d-4, T0=300.d0, N=1.2d-2, p0 = 100000.d0, alpha=5.d-4
  !stretching deformation field ( A = -d(vg)/dy = constant)
 DOUBLE PRECISION, PARAMETER :: Aconstant = 1.d-5
  !Strength, position and structure of jet
 DOUBLE PRECISION, PARAMETER :: Ujet0=50.d0, y0=0.d0, z0=1.d4, Yscale1=5.d5, Yscale2=5.d5, Zscale=6.d3
  !Strength, position and structure of PV anomaly
 DOUBLE PRECISION, PARAMETER :: PVano0=2.d-4, PVy0=0.d0, PVz0=1.d4, PVYscale1=5.d5, PVYscale2=5.d5, PVZscale=5.d3
 !----------------------------------------------------------------------------------------------------------------

  !coordinates
 DOUBLE PRECISION, DIMENSION(ygrid) :: y
 DOUBLE PRECISION, DIMENSION(zgrid) :: z
!---BASIC variables      (print-to-file)
  !-potential vorticity, geostrophic streamfunction, buoyancy, total potential temperature, geostrophic velocity, frontogenetic forcing
  !-ageostrophic streamfunction, ageostrophic velocity components, thermal wind balance, deformation field
 DOUBLE PRECISION, DIMENSION(ygrid, zgrid) :: PV, gphi, buoy, U, FF, phi, va, wa, Theta_bg,TWB, A
!---Additional variables (not stored-to-file)
  !Boundaries and PV components
 DOUBLE PRECISION, DIMENSION(2,zgrid)      :: PVFF_lateral
 DOUBLE PRECISION, DIMENSION(ygrid)        :: PVFF_surface
 DOUBLE PRECISION, DIMENSION(ygrid,zgrid)  :: PV_ts, PV_ano, gphi_ts, gphi_ano
  ! static, inertial and baroclinic frequencies and PVFF=PV*N^2 (only for semigeostrophic case) and vg=A*y
 DOUBLE PRECISION, DIMENSION(ygrid, zgrid) :: N2, F2, S2, PVFF, vg

!---Imported PV field parameters
  !importing cross data sets
 DOUBLE PRECISION, DIMENSION(ygrid, zgrid) :: PVin, Ain, surfaceBCin, lateralBCin
  !import dates
 CHARACTER(13), DIMENSION(16)              :: darray
  !import domain sizes y and z
 INTEGER                                   :: y_imp, z_imp

!---Other
 INTEGER :: i,j, k=0, it=0, countnegPV=0        !x,y,t_import,t, # of negative PV
 DOUBLE PRECISION :: dy, dz, beta, rhoJ, w      !dy,dz,beta=dy/dz, 2 optimal SOR parameters
 DOUBLE PRECISION, DIMENSION(6)  :: filedata_ini      !Write PV fields to file
 DOUBLE PRECISION, DIMENSION(15) :: filedata_result   !Write from-PV-derived fields to file
END MODULE Inversion_globalvar






!----------------------------------------------------------------------------------------------------------------

```

```fortran
!------------------------------------------------ Supporting Functions ---------------------------------------------

!------------------------------------------------------------------------------------------------------------------







MODULE Inversion_support
 USE Inversion_globalvar
 USE nummethods
 IMPLICIT NONE

 CONTAINS
  SUBROUTINE caseselect(input)
   IMPLICIT NONE

   CHARACTER(LEN=*), INTENT(IN) :: input

   SELECT CASE (input)
    CASE ("jet")
      inimodel = input
      PRINT *, 'Simulation jet-front system chosen:'
      PRINT *, '>> Symmetric jet structure is centered at (y,z) = (', y0, ',', z0, ') m'
      PRINT *, '>> Horizontal length scale : yscale = ', yscale1, ' m'
      PRINT *, '>> Vertical length scale   : zscale = ', zscale, ' m'
      PRINT *, '>> Thermal front is in thermal wind balance with chosen jet structure'
    CASE ("ano")
      inimodel = input
      PRINT *, 'Simulation PV anomaly chosen'
      PRINT *, '>> PV anomaly is centered at (y,z) = (', y0, ',', z0, ') m'
      PRINT *, '>> Horizontal length scale : yscale = ', yscale1, ' m'
      PRINT *, '>> Vertical length scale   : zscale = ', zscale, ' m'
    CASE ("imp")
      inimodel = input
      PRINT *, 'UNDER CONSTRUCTION!! Not user-friendly enough at the moment'
      PRINT *, 'PV field shall be imported from data files'
      PRINT *, '>> Note that the following files are needed:'
      DO i=1,len(darray)
      PRINT *, i, 'wref_param_date.txt'
      END DO
      PRINT *, '>> Where param is 60 for PV, 167 for surface temperature and 131 for jet stream on lateral boundaries'

    CASE ("QG")
      SG=.FALSE.
    CASE ("SG")
      SG=.TRUE.

    CASE ("y")
      stratos=.TRUE.
    CASE ("n")
      stratos=.FALSE.
    CASE DEFAULT
      PRINT *, 'Wrong input: please try again.'
      STOP
   END SELECT
  END SUBROUTINE caseselect

  FUNCTION Data_in_date(no,time)
   IMPLICIT NONE

   DOUBLE PRECISION, DIMENSION(ygrid,zgrid) :: Data_in_date
   CHARACTER(LEN=*)                         :: no, time

   OPEN (25, file='Wrefs/wref_'//no//'_'//time//'.txt')
   DO j=1,81
    IF (no=='167' .AND. (j.NE.1)) THEN
     Data_in_date(:,j) = 0.d0
    ELSEIF (no=='1311') THEN
     READ (25, *) (Data_in_date(i,j),i=1,2)
     Data_in_date(3:101,j) = 0.d0
    ELSE
     READ (25, *) (Data_in_date(i,j),i=1,101)
```

```
    ENDIF
   END DO
   CLOSE(25)
  END FUNCTION Data_in_date

  SUBROUTINE smoothing(param)
   IMPLICIT NONE

   DOUBLE PRECISION, DIMENSION(ygrid,zgrid) :: param
   DOUBLE PRECISION, DIMENSION(ygrid,zgrid) :: oldparam

   oldparam = param
   DO i=2,size(y)-1
    DO j=2,size(z)-1
     param(i,j) = (oldparam(i-1,j+1)+oldparam(i,j+1)+oldparam(i+1,j+1)+oldparam(i-1,j)+4.d0*oldparam(i,j)+ &
                  oldparam(i+1,j)+oldparam(i-1,j-1)+oldparam(i,j-1)+oldparam(i+1,j-1))/12.d0
    END DO
   END DO
  END SUBROUTINE smoothing

  SUBROUTINE allowoutput()
   IMPLICIT NONE
   IF (int(it/intervalit) == (it*1.d0/intervalit)) THEN
    outputok = .TRUE.
   ELSE
    outputok = .FALSE.
   ENDIF
  END SUBROUTINE allowoutput

!------------------------------------------------------------------------
!------------------------------- PV-inversion routines --------------------
!------------------------------------------------------------------------

  SUBROUTINE SORrelaxation_PVSG(PVlocal,PVFF_surfacelocal,PVFF_lat,func)
    !This subroutine is called twice:
    !- one time for total PV >> SORrelaxation_PVSG(PV,PVFF_surface+g/f,gphi)
    !- one time for background PV >> SORrelaxation_PVSG(PV_bg,g/f,gphi_bg)
    IMPLICIT NONE
    !extra iteration step for estimation phi including nonlinear terms
    DOUBLE PRECISION, DIMENSION(ygrid, zgrid), INTENT(IN) :: PVlocal
    DOUBLE PRECISION, DIMENSION(ygrid), INTENT(IN) :: PVFF_surfacelocal
    DOUBLE PRECISION, DIMENSION(2,zgrid), INTENT(IN) :: PVFF_lat

    DOUBLE PRECISION, DIMENSION(ygrid, zgrid), INTENT(OUT) :: func

    DOUBLE PRECISION, DIMENSION(ygrid, zgrid) :: func_old
    INTEGER :: ij, imax=5000
    DOUBLE PRECISION :: err=1.d-4, anorm, anorm_old, r

    !These relaxation parameters work best for the three configurations:
    IF (inimodel.NE.'imp') THEN
     r = 0.8d0
    ELSE
     r = 0.2d0
    ENDIF

    !initial streamfunction
    func_old = 0.d0
    DO i=1,size(y)
     DO j=1,size(z)
      PVFF(i,j) = PVlocal(i,j)*N**2
     END DO
    END DO
    CALL SORrelaxation_BC(PVFF,PVFF_surfacelocal,PVFF_lat,func)

    !SOR above is only QG >> test gphi field here before influence of nonlinear terms come into play!
    !>>Solution for func is okay here (what you expect from QG)
    !CALL Testplotting(func-gphi_ts) !gphi_ano is what remains

    !start iterating until nonlinear terms are also represented well enough
    DO ij=1,imax
     !correct func using r to avoid overestimation of func. The new func value is a factor r larger than func_old
     IF (ij .NE. 1) THEN
      func = func_old + r*(func-func_old)
     ENDIF
     !new forcing term including nonlinear terms
```

```
      anorm = 0.d0
      anorm_old = 0.d0
      DO i=1,size(y)
       DO j=1,size(z)-1
         !Two important notes here
         !1. second term is questionable >> use of PVFF_surface which is a parameter that does not change (but is small anyway?)
         !2. Lateral boundary conditions have been included for QG case. Also change this part???????????????????????????
         ! >> probably not: lateral BC are arranged in SORrelaxation_BC with input of PVFF on ALL gridpoints!!
         ! >> PVFF comes from PV and need not to be changed (except z=0) hereunder for guessing the nonlinear terms.
         ! >> nonlinear terms of PV only make a difference at center of domain!
        IF (i==1) THEN
         IF (j==1) THEN
          PVFF(i,j) = PVlocal(i,j)*N**2 - &
           f*((func(i+1,j)-func(i,j)-dy*PVFF_lat(1,j))/(.5d0*dy**2))*((func(i,j+1)-func(i,j)-dz*PVFF_surfacelocal(i))/(.5d0*dz**2)) + &
           0.d0 !f*((func(i+1,j+1)-func(i+1,j)-func(i,j+1)+func(i,j))/(dy*dz))**2
         ELSE
          PVFF(i,j) = PVlocal(i,j)*N**2 - &
           f*((func(i+1,j)-func(i,j)-dy*PVFF_lat(1,j))/(.5d0*dy**2))*((func(i,j+1)-2.d0*func(i,j)+func(i,j-1))/dz**2) + &
           0.d0 !f*((func(i+1,j+1)-func(i+1,j-1)-func(i,j+1)+func(i,j-1))/(2.d0*dy*dz))**2
         ENDIF
        ELSEIF (i==size(y)) THEN
         IF (j==1) THEN
          PVFF(i,j) = PVlocal(i,j)*N**2 - &
           f*((func(i-1,j)-func(i,j)+dy*PVFF_lat(2,j))/(.5d0*dy**2))*((func(i,j+1)-func(i,j)-dz*PVFF_surfacelocal(i))/(.5d0*dz**2)) + &
           0.d0 !f*((-func(i-1,j+1)+func(i-1,j)+func(i,j+1)-func(i,j))/(dy*dz))**2
         ELSE
          PVFF(i,j) = PVlocal(i,j)*N**2 - &
           f*((func(i-1,j)-func(i,j)+dy*PVFF_lat(2,j))/(.5d0*dy**2))*((func(i,j+1)-2.d0*func(i,j)+func(i,j-1))/dz**2) + &
           0.d0 !f*((-func(i-1,j+1)+func(i-1,j-1)+func(i,j+1)-func(i,j-1))/(2.d0*dy*dz))**2
         ENDIF
        ELSEIF (j==1 .AND. ((i.NE.1) .OR. (i.NE.size(y)))) THEN
         PVFF(i,j) = PVlocal(i,j)*N**2 - &
          f*((func(i+1,j)-2.d0*func(i,j)+func(i-1,j))/dy**2)*((func(i,j+1)-func(i,j)-dz*PVFF_surfacelocal(i))/(.5d0*dz**2)) + &
          f*((PVFF_surfacelocal(i+1)-PVFF_surfacelocal(i-1))/(2.d0*dy))**2
          !f*((func(i+1,j+1)-func(i+1,j)-func(i-1,j+1)+func(i-1,j))/(2.d0*dy*dz))**2
        ELSE
         PVFF(i,j) = PVlocal(i,j)*N**2 - &
          f*((func(i+1,j)-2.d0*func(i,j)+func(i-1,j))/dy**2)*((func(i,j+1)-2.d0*func(i,j)+func(i,j-1))/dz**2) + &
          f*((func(i+1,j+1)-func(i+1,j-1)-func(i-1,j+1)+func(i-1,j-1))/(4.d0*dy*dz))**2
        ENDIF
        anorm = anorm+abs(func(i,j)-func_old(i,j))
        anorm_old = anorm_old + abs(func_old(i,j))
       END DO
      END DO
      print *, ij, anorm/anorm_old, err
      IF(anorm .LT. err*anorm_old) RETURN

      !remember some parameters
      func_old = func
      !new relaxation step
      CALL SORrelaxation_BC(PVFF,PVFF_surfacelocal,PVFF_lat,func)
     END DO
     PAUSE 'imax exceeded in SORrelaxation_PVSG'
   END SUBROUTINE SORrelaxation_PVSG

   SUBROUTINE SORrelaxation_BC(PVFF,PVFF_surface,PVFF_lat,func)
     IMPLICIT NONE
     !version where the estimated parameters are NONZERO on the surface boundary
     DOUBLE PRECISION, DIMENSION(ygrid, zgrid), INTENT(IN) :: PVFF
     DOUBLE PRECISION, DIMENSION(ygrid), INTENT(IN) :: PVFF_surface
     DOUBLE PRECISION, DIMENSION(2,zgrid), INTENT(IN) :: PVFF_lat
     DOUBLE PRECISION, DIMENSION(ygrid, zgrid), INTENT(OUT) :: func

     DOUBLE PRECISION, DIMENSION(ygrid, zgrid) :: a,b,c,d,e,forc,g

     DO i=1,size(y)
      DO j=1,size(z)
       !Calculate coefficients of SG circulation equation
       IF (i==1) THEN                    !western lateral B.C.
        IF (j==1) THEN                    !lowerleftcorner of domain
         c(i,j) = 2.d0*f**2*beta**2
         d(i,j) = 0.d0
```

```
         forc(i,j) = (dy**2)*PVFF(i,j) + 2.d0*f**2*beta**2*dz*PVFF_surface(i) + 2.d0*N**2*dy*PVFF_lat(1,j)
        !ELSEIF (j==size(z)) THEN          !upperleftcorner of domain
        ! c(i,j) = 0.d0
        ! d(i,j) = 2.d0*f**2*beta**2
        ! forc(i,j) = (dy**2)*PVFF(i,j)
        ELSE
         c(i,j) = f**2*beta**2
         d(i,j) = f**2*beta**2
         forc(i,j) = (dy**2)*PVFF(i,j) + 2.d0*N**2*dy*PVFF_lat(1,j)
        ENDIF
        a(i,j) = 2.d0*N**2
        b(i,j) = 0.d0
       ELSEIF (i==size(y)) THEN           !eastern lateral B.C.
        IF (j==1) THEN                     !lowerrightcorner of domain
         c(i,j) = 2.d0*f**2*beta**2
         d(i,j) = 0.d0
         forc(i,j) = (dy**2)*PVFF(i,j) + 2.d0*f**2*beta**2*dz*PVFF_surface(i) - 2.d0*N**2*dy*PVFF_lat(2,j)
        !ELSEIF (j==size(z)) THEN                  !upperrightcorner of domain
        ! c(i,j) = 0.d0
        ! d(i,j) = 2.d0*f**2*beta**2
        ! forc(i,j) = (dy**2)*PVFF(i,j)
        ELSE
         c(i,j) = f**2*beta**2
         d(i,j) = f**2*beta**2
         forc(i,j) = (dy**2)*PVFF(i,j) - 2.d0*N**2*dy*PVFF_lat(2,j)
        ENDIF
        a(i,j) = 0.d0
        b(i,j) = 2.d0*N**2
       ELSEIF (j==1 .AND. ((i.NE.1) .OR. (i.NE.size(y)))) THEN   !on surface boundary (except lateral borders)
        a(i,j) = N**2
        b(i,j) = N**2
        c(i,j) = 2.d0*f**2*beta**2
        d(i,j) = 0.d0
        forc(i,j) = (dy**2)*PVFF(i,j) + 2.d0*f**2*beta**2*dz*PVFF_surface(i)
       !ELSEIF (j==size(z) .AND. ((i.NE.1) .OR. (i.NE.size(y)))) THEN    !on upper boundary (except lateral borders)
       ! a(i,j) = N**2
       ! b(i,j) = N**2
       ! c(i,j) = 0.d0
       ! d(i,j) = 2.d0*f**2*beta**2
       ! forc(i,j) = (dy**2)*PVFF(i,j)
       ELSE                                !inner domain
        a(i,j) = N**2
        b(i,j) = N**2
        c(i,j) = f**2*beta**2
        d(i,j) = f**2*beta**2
        forc(i,j) = (dy**2)*PVFF(i,j)
       ENDIF
       e(i,j) = -2.d0*(N**2+f**2*beta**2)
       g(i,j) = 0.d0
      END DO
     END DO

     !use version 5 of SOR relaxation method including boundary problem
     CALL SOR_5(a,b,c,d,e,forc,g,func,w,ygrid,zgrid)
   END SUBROUTINE SORrelaxation_BC

   SUBROUTINE PV_circulation(buoylocal,Ulocal,FFlocal,gphilocal)
    IMPLICIT NONE

    DOUBLE PRECISION, DIMENSION(ygrid,zgrid), INTENT(IN) :: gphilocal
    DOUBLE PRECISION, DIMENSION(ygrid,zgrid), INTENT(OUT) :: buoylocal, Ulocal, FFlocal

    !Calculate some fields and write resulting values to file
     !geostrophic velocity
     DO i=1,size(y)
      IF (i==1) THEN
       Ulocal(i,:) = -(gphilocal(i+1,:)-gphilocal(i,:))/dy
      ELSEIF (i==size(y)) THEN
       Ulocal(i,:) = -(gphilocal(i,:)-gphilocal(i-1,:))/dy
      ELSE
       Ulocal(i,:) = -(gphilocal(i+1,:)-gphilocal(i-1,:))/(2.d0*dy)
      ENDIF
     END DO
     !buoyancy
     DO j=1,size(z)
      IF (j==1) THEN
```

```
      buoylocal(:,j) = f*(gphilocal(:,j+1)-gphilocal(:,j))/dz
     ELSEIF (j==size(z)) THEN
      buoylocal(:,j) = f*(gphilocal(:,j)-gphilocal(:,j-1))/dz
     ELSE
      buoylocal(:,j) = f*(gphilocal(:,j+1)-gphilocal(:,j-1))/(2.d0*dz)
     ENDIF
    END DO
    !lateral conditions (to avoid lateral boundary problems!)
    !buoylocal(1,:) = buoylocal(2,:)
    !buoylocal(size(y),:) = buoylocal(size(y)-1,:)

    !next: calculate new frontogenetic forcing
    !Also, check thermal wind balance relation between Ug and buoy:
    DO i=2,size(y)-1
     DO j=1,size(z)-1
      FFlocal(i,j) = Calc_FF2(buoylocal(i-1,j), buoylocal(i+1,j), A(i,j), dy)
      IF (j.NE.1) TWB(i,j) = (-f*(U(i,j+1)-U(i,j-1))/(2.d0*dz))/((buoy(i+1,j)-buoy(i-1,j))/(2.d0*dy))
      !For case-study input
      IF ( (z(j)>20000) .AND. (abs(FFlocal(i,j))>0.5d-11) .AND. inimodel=='imp') THEN
       FFlocal(i,j) = 0.5d-11
      ELSEIF ( (z(j)<=1000) .AND. (abs(FFlocal(i,j))>0.5d-11) .AND. inimodel=='imp') THEN
       FFlocal(i,j) = 0.d0
      ENDIF
     END DO
    END DO
  END SUBROUTINE PV_circulation

  SUBROUTINE new_frequencies()
   IMPLICIT NONE
    !adjust the frequency parameters to the new U and buoy fields
    !useful for input Sawyer-Eliassen equation
    DOUBLE PRECISION, DIMENSION(ygrid,zgrid) :: instabilitycheck

    !Static frequency
    DO j=1,size(z)
     !N2 comes from isothemal ref. atmosphere + both perturbations (buoy)
     IF (j==1) THEN
      N2(:,j) = N**2 + (buoy(:,j+1)-buoy(:,j))/dz
     ELSEIF (j==size(z)) THEN
      N2(:,j) = N**2 + (buoy(:,j)-buoy(:,j-1))/dz
     ELSE
      N2(:,j) = N**2 + (buoy(:,j+1)-buoy(:,j-1))/(2.d0*dz)
     ENDIF
    END DO

    !Baroclinic and inertial frequency
    DO i=1,size(y)
      DO j=1,size(z)
       IF (i==1 .OR. i==size(y)) THEN
        S2(i,j) = 0.d0          !db /dy=0
        F2(i,j) = f**2          !dug/dy=0
       ELSE
        S2(i,j) = (buoy(i+1,j)-buoy(i-1,j))/(2.d0*dy)
        F2(i,j) = f*(f - (U(i+1,j)-U(i-1,j))/(2.d0*dy))
       ENDIF
       ! check if  SGPV = N^2*F^2-S^4 is greater than 0.
       ! Or else, Sawyer-Eliassen equation can not be solved
       instabilitycheck = Calc_PV(N2(i,j),F2(i,j),S2(i,j)**2,y(i),z(j))
      END DO
    END DO
  END SUBROUTINE new_frequencies

  SUBROUTINE SORrelaxation(N2,F2,S2,FF,phi)
    IMPLICIT NONE
    !simpler version where the estimated parameters are ZERO on the surface boundary
    DOUBLE PRECISION, DIMENSION(ygrid, zgrid), INTENT(IN) :: N2,F2,S2,FF
    DOUBLE PRECISION, DIMENSION(ygrid, zgrid), INTENT(OUT) :: phi

    DOUBLE PRECISION, DIMENSION(ygrid, zgrid) :: a,b,c,d,e,forc,g

    DO i=1,size(y)
     DO j=1,size(z)
        !Calculate coefficients of SG circulation equation
      a(i,j) = N2(i,j)
      b(i,j) = N2(i,j)
      c(i,j) = F2(i,j)*beta**2
```

```fortran
      d(i,j) = F2(i,j)*beta**2
      e(i,j) = -2.d0*(N2(i,j)+F2(i,j)*beta**2)
      forc(i,j) = (dy**2)*FF(i,j)
      g(i,j) = -.5d0*S2(i,j)*beta
     END DO
    END DO
    CALL SOR_4(a,b,c,d,e,forc,g,phi,w,ygrid,zgrid)
  END SUBROUTINE SORrelaxation

  SUBROUTINE ageo_circulation()
   IMPLICIT NONE
    !Calculate ageostrophic wind components
    !wa
    DO i=2,size(y)-1
      wa(i,:) = -(phi(i+1,:)-phi(i-1,:))/(2.d0*dy)
    END DO
    !va
    DO j=1,size(z)-1
     IF (j==1) THEN
      va(:,j) = (phi(:,j+1)-phi(:,j))/dz
     ELSE
      va(:,j) = (phi(:,j+1)-phi(:,j-1))/(2.d0*dz)
     ENDIF
    END DO
  END SUBROUTINE ageo_circulation

  SUBROUTINE results_to_file()
   IMPLICIT NONE
    DOUBLE PRECISION, DIMENSION(6)  :: filedata_ini
    DOUBLE PRECISION, DIMENSION(15) :: filedata_result

    DO i=1,size(y)
     DO j=1,size(z)-1
      !initial PV field
      filedata_ini = (/ it*1.d0, y(i)*1.d-3, z(j)*1.d-3, 1.d4*PV(i,j), 0.d0, 0.d0 /)
      CALL writetofile_small(14, 1, filedata_ini)
      !resulting field
      filedata_result = (/ it*1.d0, y(i)*1.d-3, z(j)*1.d-3, gphi(i,j), buoy(i,j), U(i,j), 1.d11*FF(i,j), &
          phi(i,j), va(i,j), 1.d2*wa(i,j), Theta_bg(i,j)+buoy(i,j)*T0/g, TWB(i,j), 1.d4*N2(i,j), 1.d4*F2(i,j), 1.d4*S2(i,j) /)
      CALL writetofile_large(15, filedata_result)
     END DO
     !additional white space
     WRITE(14,'(e17.8, e17.8, e17.8, e17.8, e17.8, e17.8)')
     WRITE(15,'(e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8)')
    END DO
    !additional white space
    WRITE(14,'(e17.8, e17.8, e17.8, e17.8, e17.8, e17.8)')
    WRITE(15, '(e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8)')

    !close data files
    IF (it==t1) CLOSE(14)
    IF (it==t1) CLOSE(15)
  END SUBROUTINE results_to_file

  SUBROUTINE writetofile_small(filename, interval, dataset)
    !generating small datasets (6 columns) used for vectorplots. Interval determines the
    !separation of these vectors.
    IMPLICIT NONE
    INTEGER, INTENT(IN) :: filename, interval
    DOUBLE PRECISION, DIMENSION(:) :: dataset
    !Reduce the dataset such that only the vector arrows on certain gridpoints (seperated by 'interval')
    !are plotted!
    IF ( (y(i)/interval == int(y(i)/interval)) .AND. (z(j)/interval == int(z(j)/interval)) ) THEN
    WRITE(filename,'(e17.8, e17.8, e17.8, e17.8, e17.8, e17.8)') dataset
    ELSE
    dataset(5:6) = 0 !set length of vector arrows to zero (????)
    WRITE(filename,'(e17.8, e17.8, e17.8, e17.8, e17.8, e17.8)') dataset
    END IF
  END SUBROUTINE writetofile_small

  SUBROUTINE writetofile_large(filename, dataset)
    IMPLICIT NONE
    INTEGER, INTENT(IN) :: filename
    DOUBLE PRECISION, DIMENSION(:) :: dataset

```

```
    WRITE(filename,&
    '(e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8, e17.8)') dataset
  END SUBROUTINE writetofile_large

  !--------------------- - - - FUNCTIONS - - - --------------------------
  ! INITIAL FUNCTIONS
  FUNCTION Ujet()
    IMPLICIT NONE
    DOUBLE PRECISION, DIMENSION(ygrid, zgrid) :: Ujet
    !use circular jet!!
    DO i=1,size(y)
     DO j=1,size(z)
       IF ((y(i)-y0)>=0) THEN
        Ujet(i,j) = Ujet0*exp(-((y(i)-y0)/Yscale1)**2)*exp(-((z(j)-z0)/Zscale)**2)
       ELSE
        Ujet(i,j) = Ujet0*exp(-((y(i)-y0)/Yscale2)**2)*exp(-((z(j)-z0)/Zscale)**2)
       ENDIF
     END DO
    END DO
  END FUNCTION Ujet

  FUNCTION ini_buoy()
    IMPLICIT NONE
    DOUBLE PRECISION, DIMENSION(ygrid, zgrid) :: ini_buoy

    !Solving TWB relation
    DO i=1,size(y)
     DO j=1,size(z)
       IF ((y(i)-y0)>=0) THEN
       ini_buoy(i,j) = &
        (f*2.d0*Ujet0*(z(j)-z0))/(Zscale**2)*exp(-((z(j)-z0)/Zscale)**2)*Yscale1*.5d0*sqrt(pi)*erf((y(i)-y0)/Yscale1)
      ELSE
       ini_buoy(i,j) = &
        (f*2.d0*Ujet0*(z(j)-z0))/(Zscale**2)*exp(-((z(j)-z0)/Zscale)**2)*Yscale2*.5d0*sqrt(pi)*erf((y(i)-y0)/Yscale2)
      ENDIF
     END DO
    END DO
  END FUNCTION ini_buoy

  ! TIME EVOLUTION FUNCTIONS

  FUNCTION Calc_Mg(U, f, y)
    IMPLICIT NONE
    DOUBLE PRECISION :: Calc_Mg, U, f, y
    ! absolute momentum calculation used for calculating a new F2
    Calc_Mg = U - f*y
  END FUNCTION Calc_Mg

  FUNCTION Calc_Q(U1, U2, A, f, T0, g, dz)
    IMPLICIT NONE
    DOUBLE PRECISION :: Calc_Q, U1, U2, A, f, T0, g, dz
    !Frontogenetic forcing at new timestep
    Calc_Q = -A*f*T0*(U2-U1)/(2.d0*g*dz)
  END FUNCTION Calc_Q

  FUNCTION Calc_FF(U1, U2, A, f, dz)
    IMPLICIT NONE
    DOUBLE PRECISION :: Calc_FF, U1, U2, A, f, dz
    !Total forcing at new timestep
    Calc_FF = A*f*(U2-U1)/dz
    !note: factor 2 in denumerator drops here
  END FUNCTION Calc_FF

  FUNCTION Calc_FF2(b1, b2, A, dy)
    IMPLICIT NONE
    DOUBLE PRECISION :: Calc_FF2, b1, b2, A, dy
    !Total forcing at new timestep
    Calc_FF2 = -2.d0*A*(b2-b1)/(2.d0*dy)
    !note: factor 2 in denumerator drops here
  END FUNCTION Calc_FF2

  FUNCTION Calc_PV(N_2,F_2,S_4, y, z)
    IMPLICIT NONE
    DOUBLE PRECISION :: Calc_PV, N_2, F_2, S_4, y, z
    !q = N^2 F^2 - S^4
    Calc_PV = N_2*F_2 - S_4
```

```fortran
    IF(Calc_PV<0) THEN
     print *, 'coord:', y, z, 'PV = ', Calc_PV, 'CANCEL PROGRAM!!'
     print *, '--------------------------------------------------'
     print *, 'N_eff^2: ', N_2
     print *, 'F^2: ', F_2
     print *, 'S^4: ', S_4
     print *, '--------------------------------------------------'
     print *, 'db/dz: ', N_2-N**2
     print *, 'dug/dy: ', -(F_2- f**2)/f
     print *, 'db/dy: ', sqrt(S_4)
     print *, '--------------------------------------------------'
    ENDIF
  END FUNCTION Calc_PV

  !version 8: PV functions ----------------------------

  !initial QGPV based on exponential structure of jet (circular)
  !using different tropopause formulation (local1)
  FUNCTION ini_PVQG()
    IMPLICIT NONE
    DOUBLE PRECISION, DIMENSION(ygrid,zgrid) :: ini_PVQG

    DOUBLE PRECISION :: local1, local2, local3

    !from analytic expression
    DO i=1,size(y)
     DO j=1,size(z)
      IF ((y(i)-y0)>=0) THEN
       local2 = -2.d0*Ujet0*exp(-((y(i)-y0)/Yscale1)**2)*exp(-((z(j)-z0)/Zscale)**2)*(y(i)-y0)/(Yscale1**2)  !dug/dy
       local3 = f*Ujet0*Yscale1*sqrt(pi)*erf((y(i)-y0)/Yscale1)*exp(-((z(j)-z0)/Zscale)**2)*(1.d0- &
                2.d0*((z(j)-z0)/Zscale)**2)/(Zscale**2)  !db/dz
      ELSE
       local2 = -2.d0*Ujet0*exp(-((y(i)-y0)/Yscale2)**2)*exp(-((z(j)-z0)/Zscale)**2)*(y(i)-y0)/(Yscale2**2)  !dug/dy
       local3 = f*Ujet0*Yscale2*sqrt(pi)*erf((y(i)-y0)/Yscale2)*exp(-((z(j)-z0)/Zscale)**2)*(1.d0- &
                2.d0*((z(j)-z0)/Zscale)**2)/(Zscale**2)  !db/dz
      ENDIF
      ini_PVQG(i,j) = (f/(N**2))*local3 - local2
     END DO
    END DO

  END FUNCTION ini_PVQG

  !initial SGPV based on exponential structure of jet
  !using different tropopause formulation (local1)
  FUNCTION ini_PVSG()
    IMPLICIT NONE
    DOUBLE PRECISION, DIMENSION(ygrid,zgrid) :: ini_PVSG

    DOUBLE PRECISION :: local2, local3, local4

    !from analytic expression
    DO i=1,size(y)
     DO j=1,size(z)
      IF ((y(i)-y0)>=0) THEN
       local2 = -2.d0*Ujet0*exp(-((y(i)-y0)/Yscale1)**2)*exp(-((z(j)-z0)/Zscale)**2)*(y(i)-y0)/(Yscale1**2)  !dug/dy
       local3 = f*Ujet0*Yscale1*sqrt(pi)*erf((y(i)-y0)/Yscale1)*exp(-((z(j)-z0)/Zscale)**2)*(1.d0- &
                2.d0*((z(j)-z0)/Zscale)**2)/(Zscale**2)  !db/dz
       local4 = 2.d0*f*Ujet0*exp(-((y(i)-y0)/Yscale1)**2)*(z(j)-z0)/(Zscale**2)*exp(-((z(j)-z0)/Zscale)**2)  !db/dy
      ELSE
       local2 = -2.d0*Ujet0*exp(-((y(i)-y0)/Yscale2)**2)*exp(-((z(j)-z0)/Zscale)**2)*(y(i)-y0)/(Yscale2**2)  !dug/dy
       local3 = f*Ujet0*Yscale2*sqrt(pi)*erf((y(i)-y0)/Yscale2)*exp(-((z(j)-z0)/Zscale)**2)*(1.d0- &
                2.d0*((z(j)-z0)/Zscale)**2)/(Zscale**2)  !db/dz
       local4 = 2.d0*f*Ujet0*exp(-((y(i)-y0)/Yscale2)**2)*(z(j)-z0)/(Zscale**2)*exp(-((z(j)-z0)/Zscale)**2)  !db/dy
      ENDIF
      ini_PVSG(i,j) = (f/(N**2))*local3 - local2 - (1.d0/(N**2))*local3*local2 - (1.d0/(f*N**2))*local4**2
     END DO
    END DO
  END FUNCTION ini_PVSG

  FUNCTION ini_PV3()
    IMPLICIT NONE
    DOUBLE PRECISION, DIMENSION(ygrid,zgrid) :: ini_PV3

    !from analytic expression
    DO i=1,size(y)
      DO j=1,size(z)
```

```fortran
       IF ((y(i)-y0)>=0) THEN
        ini_PV3(i,j) = PVano0*exp(-((y(i)-PVy0)/PVYscale1)**2)*exp(-((z(j)-PVz0)/PVZscale)**2)
       ELSE
        ini_PV3(i,j) = PVano0*exp(-((y(i)-PVy0)/PVYscale2)**2)*exp(-((z(j)-PVz0)/PVZscale)**2)
       ENDIF
      END DO
    END DO
  END FUNCTION ini_PV3

  SUBROUTINE smallest(param)
   IMPLICIT NONE
   DOUBLE PRECISION, DIMENSION(ygrid,zgrid), INTENT(IN) :: param
   DOUBLE PRECISION :: minim

   !minimal values
   minim = param(1,1)
   DO i=1,size(y)
    DO j=1,size(z)
     IF (param(i,j) < minim) minim = param(i,j)
    END DO
   END DO
   print *, 'Minimal value SGPV: ', minim
   IF (minim<=0.d0) THEN
    print *, 'PV has negative values somewhere. SGPV-inversion routine not expected to work.'
   ENDIF
  END SUBROUTINE smallest

END MODULE Inversion_support



!------------------------------------------------------------------------------------------------------------------------

!--------------------------------------------------   General subroutines   -------------------------------------------

!------------------------------------------------------------------------------------------------------------------------





MODULE Inversion_general
 USE Inversion_support
 IMPLICIT NONE

 CONTAINS

  SUBROUTINE Input_user()
   IMPLICIT NONE

   CHARACTER(LEN=3) :: input1
   CHARACTER(LEN=2) :: input2
   CHARACTER(LEN=1) :: input3


   PRINT *, 'Welcome user!'
   PRINT *, 'Current version: v1.0'
   PRINT *, 'This program is used for simulating dynamical processes from a PV perspective'
   PRINT *, 'For example: frontogenesis or tropopause fold evolution'
   PRINT *, '-----------------------------------------------------------'

   PRINT *, 'First, choose your initial problem to simulate:'
   PRINT *, '1. Jet-front system     (type "jet")'
   PRINT *, '2. Symmetric PV anomaly (type "ano")'
   PRINT *, '3. Import PV field      (type "imp")'
   READ *, input1
   PRINT *, ''
   CALL caseselect(input1)
   PRINT *, '-----------------------------------------------------------'

   PRINT *, 'All other fields may be derived numerically using Piecewise PV-Inversion when the following 3 conditions are met:'
   PRINT *, '(I) A reference state has to be chosen for an atmosphere at rest'
   PRINT *, '>> This program uses a linear stratified reference atmosphere where the static stability N^2 is constant.'
   PRINT *, '>> N = ', N, ' /s'
   PRINT *, ''
```

```
  PRINT *, '(II) A specific balance condition has to be chosen for the PV perturbations upon reference atmosphere'
  PRINT *, '>> Please choose between: '
  PRINT *, '1. Quasi-Geostrophic balance (typ "QG")'
  PRINT *, '2. Semi-Geostrophic balance  (typ "SG")'
  READ *, input2
  PRINT *, ''
  CALL caseselect(input2)
  IF (input1.NE.'imp') THEN
   PRINT *, '>> Would you like to include a tropopause? (typ "y" or "n")'
   READ *, input3
   PRINT *, ''
   CALL caseselect(input3)
  ENDIF
  PRINT *, '(III) The PV inversion problem has to be globally solved with proper boundary conditions'
  PRINT *, '>> The PV inversion algorithm includes the surface and lateral boundary conditions'
  PRINT *, ''

  PRINT *, '-----------------------------------------------------------'
  PRINT *, '>> Starting simulation <<'
  PRINT *, '-----------------------------------------------------------'
 END SUBROUTINE Input_user

 SUBROUTINE Initialisation()
  IMPLICIT NONE

  !Open data file for storing initial PV perturbation values
  OPEN (14, file='Tropofold_QGPVinitial_ani.dat', ACTION='WRITE')
  WRITE(14,'(a17,a17,a17,a17,a17,a17)') &
  "␣---␣t␣(s)␣--␣", "␣--␣y␣(km)␣--␣", "␣--␣z␣(km)␣--␣", "␣PV␣(1E12s^-4)", "␣------------␣", "␣------------␣"

  !---open data file for storing resulting fields derived from PV perturbations
  OPEN (15, file='Tropofold_QGPVresult_ani.dat', ACTION='WRITE')
  WRITE(15,'(a17,a17,a17,a17,a17,a17,a17,a17,a17,a17,a17,a17,a17,a17,a17)') "␣---␣t␣(s)␣--␣", "␣--␣y␣(km)␣--␣", &
  "␣--␣z␣(km)␣--␣", "␣gPhi␣(m^2/s)␣", "␣buoy␣(m/s^2)␣", "␣-␣U␣␣(m/s)␣-␣", "␣-␣FF␣(/s^3)-␣", "␣-␣Phi␣(/s)␣-␣", &
  "␣-␣va␣(m/s)␣-␣", "␣-␣wa␣(m/s)␣-␣", "␣␣-Theta␣(K)-␣", "␣---␣TWB␣---␣␣", "␣------------␣", "␣------------␣", "␣------------␣"

  !--------------------basic variables part I-------------------------
  PV = 0.d0
  PV_ts = 0.d0
  PV_ano = 0.d0
  gphi = 0.d0
  gphi_ts = 0.d0
  gphi_ano = 0.d0
  buoy = 0.d0
  U = 0.d0
  FF = 0.d0
  phi = 0.d0
  va = 0.d0
  wa = 0.d0
  TWB = 1.d0
  A = Aconstant

  PVFF = 0.d0
  N2 = N**2
  S2 = 0.d0
  F2 = f**2

  !Calculate the optimal spectral radius for largest convergence in SOR
  !>applying approximation based on ygrid/zgrid ratio
  rhoJ = 1.d0 - (pi**2)/(2.d0*zgrid**2) + (pi**4)/(24.d0*zgrid**4)
  !And the resulting over-relaxation parameter for faster convergence
  w = 2.d0/(1.d0 + sqrt(1.d0-rhoJ**2))

  !-------------------------------------------------------
  !----initialisation depends on chosen configuration!----
  !-------------------------------------------------------

  SELECT CASE (inimodel)
   !---------jet/front system case------------
   CASE ("jet")
    dy = domain_y/(ygrid-1.d0)
    dz = domain_z/(zgrid-1.d0)
    y = ((/ (i,i=1,size(y)) /)-.5d0*(ygrid+1.d0))*dy
    z = ((/ (j,j=1,size(z)) /)-1.d0)*dz
    beta = dy/dz
```

```
  !stratification depending on tropopause or not
  DO j=1,size(z)
   !continuous tropopause included
   IF (stratos) THEN
    PV_ts(:,j) = N**2*(1.5d0*(1.d0+erf(alpha*(z(j)-z0)))) !stratosphere PVanomaly
   ELSE
    PV_ts(:,j) = N**2
   ENDIF
  END DO

  !initial values of perturbation fields
  buoy = ini_buoy()
  U = Ujet()
  CALL new_frequencies() !check for instability
  !initial boundary conditions
  PVFF_surface = buoy(:,1)/f
  PVFF_lateral(1,:) = -U(1,:)
  PVFF_lateral(2,:) = -U(size(y),:)
  IF (SG) THEN
   PV_ano = ini_PVSG()
  ELSE
   PV_ano = ini_PVQG()
  ENDIF

  PV = PV_ano+PV_ts
 !-------------PV anomaly case----------------
 CASE ("ano")
  dy = domain_y/(ygrid-1.d0)
  dz = domain_z/(zgrid-1.d0)
  y = ((/ (i,i=1,size(y)) /)-.5d0*(ygrid+1.d0))*dy
  z = ((/ (j,j=1,size(z)) /)-1.d0)*dz
  beta = dy/dz

  !stratification depending on tropopause or not
  DO j=1,size(z)
   !continuous tropopause included
   IF (stratos) THEN
    PV_ts(:,j) = N**2*(1.5d0*(1.d0+erf(alpha*(z(j)-z0)))) !stratosphere PVanomaly
   ELSE
    PV_ts(:,j) = N**2
   ENDIF
  END DO

  !initial values of perturbation fields
  PV_ano = ini_PV3()
  !initial boundary conditions
  PVFF_surface = 0.d0
  PVFF_lateral(1,:) = 0.d0
  PVFF_lateral(2,:) = 0.d0

  PV = PV_ano+PV_ts
 !--------------PV import case--------------------
 CASE ("imp")
  !read time_array
  darray = (/ '20011108-0000','20011108-0600','20011108-1200','20011108-1800',&
              '20011109-0000','20011109-0600','20011109-1200','20011109-1800',&
              '20011110-0000','20011110-0600','20011110-1200','20011110-1800',&
              '20011111-0000','20011111-0600','20011111-1200','20011111-1800' /)

  !y_imp specifically chosen for 2001nov case-study >> change later on
  y_imp = sqrt((2.d0*pi*6.371d6*(2.315d1/3.6d2))**2+(2.d0*pi*cos((4.5d1/3.6d2)*2.d0*pi)*6.371d6*(2.508d1/3.6d2))**2)
  z_imp = 4.e4
  dy = y_imp/(ygrid-1.d0)
  dz = z_imp/(zgrid-1.d0)
  y = ((/ (i,i=1,size(y)) /)-.5d0*(ygrid+1.d0))*dy
  !yplot = ((/ (i,i=1,size(y)) /)-1.d0)*10.d0  !alternative axis: put this in gnuplot script
  z = ((/ (j,j=1,size(z)) /)-1.d0)*dz
  beta = dy/dz

  PVin = Data_in_date('601',darray(6))
  !Ain  = Data_in_date('1341',darray(1))
  surfaceBCin = Data_in_date('167',darray(6))
  lateralBCin = Data_in_date('1311',darray(6))
  DO i=1,size(y)
   DO j=1,size(z)
```

```
     !A(i,j) = dble(Ain(i,j))
     PV(i,j) = dble(PVin(i,j))
     PVFF_surface(i) = dble(surfaceBCin(i,1)/f)
     PVFF_lateral(1,j) = dble(lateralBCin(1,j))
     PVFF_lateral(2,j) = dble(lateralBCin(2,j))
     !Some modifications on data
     IF (PV(i,j)<0) print *, 'negative value for PV(i,j). Coordinates', y(i), z(j), 'PV = ', PV(i,j)
     IF (PV(i,j)<0) PV(i,j) = 0.d0
     IF (j==0) PVFF_lateral(:,j) = 0.d0  !on surface boundary: ug = 0 at lateral boundaries
    END DO
   END DO

   !get rid of PV anomalies above 20km
   DO j=41,size(z)
    PV(:,j) = sum(PV(:,j))/size(y)
    PVFF_lateral(:,j) = 0.d0
   END DO

   !Smoothing of noisy import fields by horizontal averaging
   !CALL smoothing(A)

   !calculating fields at timestep it
   print *, 'timestep = ', darray(1), '.  end time = ', darray(it)
 END SELECT

 !--------------------basic variables part II-------------------------
 !information on PV, y and z needed here
 DO i=1,size(y)
  DO j=1,size(z) !potential temperature profile for reference atmosphere only
   Theta_bg(i,j) = T0*(exp((g*z(j))/(R*T0)))**(R/cp)
   vg(i,j) = -A(i,j)*y(i)
   IF (.NOT. SG) PVFF(i,j) = PV(i,j)*N**2
   IF ((.NOT. SG) .AND. (inimodel=='imp')) PVFF(i,j) = (PV(i,j)-f)*N**2
  END DO
 END DO

 !minimal values
 CALL smallest(PV+f)

 PRINT *, 'Initialisation complete'
END SUBROUTINE Initialisation



SUBROUTINE Initialisation_dt()
 IMPLICIT NONE

 CALL allowoutput() !check if results may be printed to file for this timestep

 !----------------------------------------------------
 !----PV adjustment depends on chosen configuration!----
 !----------------------------------------------------

 SELECT CASE (inimodel)
  !----------------------------------------------------jet/front system case---------------------------------
  CASE ("jet")
   print *, 'timestep = ', it, 't1 = ', t1

   PVFF_lateral(1,:) = -U(1,:)
   PVFF_lateral(2,:) = -U(size(y),:)
  !----------------------------------------------------PV anomaly case-----------------------------------------
  CASE ("ano")
   print *, 'timestep = ', it, 't1 = ', t1

   PVFF_lateral(1,:) = -U(1,:)
   PVFF_lateral(2,:) = -U(size(y),:)
  !----------------------------------------------------PV import case------------------------------------------
  CASE ("imp")
   IF (outputok) k = (it)/(6*dt)

   !print info to screen
   IF (.NOT. outputok) print *, '------- time: ', darray(k+1), ' +', (it-k*21600)/(3600), 'hours -------'
   IF (outputok) print *, 'timestep = ', darray(k+1), '.  t1 = ', darray(16)

   !import new deformation field and boundary conditions every 6 hours (when outputok=true)
   IF (outputok) THEN
```

```
      !Ain = Data_in_date('1341',darray(k))
      surfaceBCin = Data_in_date('167',darray(k+1))
      lateralBCin = Data_in_date('1311',darray(k+1))
      DO i=1,size(y)
       DO j=1,size(z)
        !A = dble(Ain)
        PVFF_surface(i) = dble(surfaceBCin(i,1)/f)
        PVFF_lateral(1,j) = dble(lateralBCin(1,j))
        PVFF_lateral(2,j) = dble(lateralBCin(2,j))
        IF (PV(i,j)<0) print *, 'negative value for PV(i,j): ',PV(i,j)
        IF (PV(i,j)<0) PV(i,j) = 0.d0
        IF (j==0) PVFF_lateral(:,j) = 0.d0  !on surface boundary: ug = 0 at lateral boundaries
       END DO
      END DO

      !apply smoothing on noisy fields
      !CALL smoothing(A)

      !Some modifications on data
      DO j=41,size(z) !get rid of PV anomalies above 20km
       PV(:,j) = sum(PV(:,j))/size(y)
       PVFF_lateral(:,j) = 0.d0
      END DO
     ENDIF
   END SELECT

   !--------------------Some modifications-------------------------
   DO i=1,size(y)
    DO j=1,size(z)
     !Some modifications on data
      IF((PV(i,j)+f)<0 .AND. SG) THEN !nonlinear inversion equation only solvable for SGPV>=0
       print *, 'coord:', y(i), z(j), 'SGPV = ', f+PV(i,j)
       !PV(i,j) = 0.d0 !solution for negative SGPV problem
      ENDIF
     vg(i,j) = -A(i,j)*y(i)
     IF (.NOT. SG) PVFF(i,j) = PV(i,j)*N**2
     IF ((.NOT. SG) .AND. (inimodel=='imp')) PVFF(i,j) = (PV(i,j)-f)*N**2 !take absolute vorticity instead of relative
    END DO
   END DO

   !minimal values
   CALL smallest(PV+f)
  END SUBROUTINE Initialisation_dt
END MODULE Inversion_general
```

PVinversion_numroutines.f90

```fortran
MODULE nummethods
USE Inversion_globalvar
IMPLICIT NONE

 CONTAINS
!-----------------------------------------------------------------------------------------------------------------
!------------------------------------Successive Over-Relaxation---------------------------------------------------
!-----------------------------------------------------------------------------------------------------------------


!--- simplest form useful when parameter (phi) is 0 on all boundaries. Iteration takes place on whole domain EXCEPT
boundaries
  SUBROUTINE SOR_4(a,b,c,d,e,forc, g, phi, omega, gs1, gs2)
    IMPLICIT NONE
    !Routine for SOR method in simplest form, based on given example in
    !'numerical recipes' H19.
    !No Chebyshev acceleration and odd-even ordening applied here.
    !>>version 4: simple form that can be used if phi=0 on all boundaries
    !input variables
    INTEGER :: maxits, gs1, gs2 !size of grid space (ygrid, zgrid)
    !relaxation parameter omega (constant), coefficients of equation a,b,c,d,e,f
    !initial guess phi and error threshold value
    DOUBLE PRECISION:: a(gs1,gs2),b(gs1,gs2),c(gs1,gs2),d(gs1,gs2) &
    ,e(gs1,gs2),forc(gs1,gs2), g(gs1,gs2), phi(gs1,gs2), err
    !maximum iterations and threshold value
    PARAMETER (maxits=10000, err=1.d-3)
    ! Remaining variables only used in subroutine
    INTEGER :: i, ipass,j,l,n
    DOUBLE PRECISION:: anorm, anormf, omega,resid, anormf_test

    !compute initial norm of residual (with threshold value err)
    anormf = 0.d0
    DO j=2,gs1-1
     DO l=2,gs2-1
      !assume here that initial value (of phi) is zero and
      !calculate norm of total forcing over whole grid space
      anormf = anormf + abs(forc(j,l))
     END DO
    END DO

    !Now iterate until the error between the total field that corresponds to forcing f
    !and the numerical solution is less than ERR.
    DO n=1,maxits
     anorm=0.d0
     DO j=2,gs1-1
      DO l=2, gs2-1
        resid = a(j,l)*phi(j+1,l)+b(j,l)*phi(j-1,l)+c(j,l)*phi(j,l+1)+d(j,l)*phi(j,l-1) &
        + e(j,l)*phi(j,l)-forc(j,l)+g(j,l)*(phi(j+1,l+1)-phi(j+1,l-1)+phi(j-1,l-1)-phi(j-1,l+1))
        anorm = anorm+abs(resid)
        !calculate new solution based on norm of residual
        phi(j,l) = phi(j,l)-omega*resid/e(j,l)
      END DO
     END DO
     !print *, n, anorm/anormf, err
     IF(anorm .LT. err*anormf) RETURN
    END DO

    PAUSE 'maxits exceeded in SOR'
  END SUBROUTINE SOR_4

!--- Routine INCLUDING 3 or 4 boundary conditions (surface and lateral boundaries and optional: upper boundary ).
!--- Iteration takes place on whole domain INCLUDING boundaries
SUBROUTINE SOR_5(a,b,c,d,e,forc, g, phi, omega, gs1, gs2)
  IMPLICIT NONE
  !Routine for SOR method in simplest form, based on given example in
  !'numerical recipes' H19.
  !No Chebyshev acceleration and odd-even ordening applied here.
  !>>version 5: including lateral and surface and upper boundary conditions

  !input variables
  INTEGER :: maxits, gs1, gs2 !size of grid space (ygrid, zgrid)
  !relaxation parameter omega (constant), coefficients of equation a,b,c,d,e,f
  !initial guess phi and error threshold value
  DOUBLE PRECISION:: a(gs1,gs2),b(gs1,gs2),c(gs1,gs2),d(gs1,gs2) &
  ,e(gs1,gs2),forc(gs1,gs2), g(gs1,gs2), phi(gs1,gs2), err
```

```
  !maximum iterations and threshold value
  PARAMETER (maxits=100000, err=1.d-3)
  ! Remaining variables only used in subroutine
  INTEGER :: i, ipass,j,l,n
  DOUBLE PRECISION:: anorm, anormf, omega,resid

  !compute initial norm of residual (with threshold value err)
  anormf = 0.d0
  DO j=1,gs1
   DO l=1,gs2-1
    !assume here that initial value (of phi) is zero and
    !calculate norm of total forcing over whole grid space
    anormf = anormf + abs(forc(j,l))
   END DO
  END DO

  !Now iterate until the error between the total field that corresponds to forcing f
  !and the numerical solution is less than ERR.
  DO n=1,maxits
   anorm=0.d0
   DO j=1,gs1
    DO l=1,gs2-1
      IF (j==1) THEN                               !on western lateral boundary
       IF (l==1) THEN                               !lowerleftcorner of domain
        resid = a(j,l)*phi(j+1,l)+c(j,l)*phi(j,l+1)+e(j,l)*phi(j,l)-forc(j,l)
       !ELSEIF (l==gs2) THEN                        !upperleftcorner of domain
       ! resid = a(j,l)*phi(j+1,l)+d(j,l)*phi(j,l-1)+e(j,l)*phi(j,l)-forc(j,l)
       ELSE
        resid = a(j,l)*phi(j+1,l)+c(j,l)*phi(j,l+1)+d(j,l)*phi(j,l-1)+e(j,l)*phi(j,l)-forc(j,l)
       ENDIF
      ELSEIF (j==gs1) THEN                          !on eastern lateral boundary
       IF (l==1) THEN                               !lowerrightcorner of domain
        resid = b(j,l)*phi(j-1,l)+c(j,l)*phi(j,l+1)+e(j,l)*phi(j,l)-forc(j,l)
       !ELSEIF (l==gs2) THEN                        !upperrightcorner of domain
       ! resid = b(j,l)*phi(j-1,l)+d(j,l)*phi(j,l-1)+e(j,l)*phi(j,l)-forc(j,l)
       ELSE
        resid = b(j,l)*phi(j-1,l)+c(j,l)*phi(j,l+1)+d(j,l)*phi(j,l-1)+e(j,l)*phi(j,l)-forc(j,l)
       ENDIF
      ELSEIF (l==1 .AND. ((j.NE.1) .OR. (j.NE.gs1))) THEN       !on surface boundary (except lateral corners)
       resid = a(j,l)*phi(j+1,l)+b(j,l)*phi(j-1,l)+c(j,l)*phi(j,l+1)+e(j,l)*phi(j,l)-forc(j,l)
      !ELSEIF (l==gs2 .AND. ((j.NE.1) .OR. (j.NE.gs1))) THEN    !on upper boundary (except lateral corners)
      ! resid = a(j,l)*phi(j+1,l)+b(j,l)*phi(j-1,l)+d(j,l)*phi(j,l-1)+e(j,l)*phi(j,l)-forc(j,l)
      ELSE                                          !inner domain
       resid = a(j,l)*phi(j+1,l)+b(j,l)*phi(j-1,l)+c(j,l)*phi(j,l+1)+d(j,l)*phi(j,l-1)+e(j,l)*phi(j,l)-forc(j,l)
      ENDIF
      anorm = anorm+abs(resid)
      !calculate new solution based on norm of residual
      phi(j,l) = phi(j,l)-omega*resid/e(j,l)
    END DO
   END DO
   !print *, n, anorm/anormf, err
   IF(anorm .LT. err*anormf) RETURN
  END DO
  PAUSE 'maxits exceeded in SOR'
 END SUBROUTINE SOR_5

!--------------------------------------------------------------------------------------------------------------------
!-----------------------------------Runge-Kutta 4th order------------------------------------------------------------
!--------------------------------------------------------------------------------------------------------------------

!Central domain
 SUBROUTINE rk4_PVfield()
  IMPLICIT NONE
  !Given a parabolic (time-dependent) p.d.e. solve it through Runge-Kutta
  !for func from t0 till t1. The tendency of func equals rhs (i.e. the
  !righthand side of function)

  !The fourth order method consists of four phases
  !After each phase the rhs of the p.d.e. needs to be updated.

  !temporary
  DOUBLE PRECISION, DIMENSION(ygrid,zgrid) :: k1, k2, k3, k4, rhs, PV_old

  PV_old = PV-PV_ts
  !rhs at t0
  rhs = Calc_PVrhs(PV_old)
```

```
  !first phase
  k1 = dt*rhs
  !calculate new rhs for next phase
  rhs = Calc_PVrhs(PV_old+.5d0*k1)

  !second phase
  k2 = dt*rhs
  !calculate new rhs for next phase
  rhs = Calc_PVrhs(PV_old+.5d0*k2)

  !third phase
  k3 = dt*rhs
  !calculate new rhs for next phase
  rhs = Calc_PVrhs(PV_old+k3)

  !fourth phase
  k4 = dt*rhs

  !calculate new function at t=t0+tau=t1 using intermediate time evaluations
  !given by the four k's
  PV = PV_ts + PV_old + (1.d0/6.d0)*(k1+2.d0*k2+2.d0*k3+k4)
 END SUBROUTINE rk4_PVfield

!Surface boundary
 SUBROUTINE rk4_PVsurface()
  IMPLICIT NONE
  !Given a parabolic (time-dependent) p.d.e. solve it through Runge-Kutta
  !for func from t0 till t1. The tendency of func equals rhs (i.e. the
  !righthand side of function)

  !The fourth order method consists of four phases
  !After each phase the rhs of the p.d.e. needs to be updated.

  !temporary
  DOUBLE PRECISION, DIMENSION(ygrid) :: k1, k2, k3, k4, rhs, PVFF_surface_old

  PVFF_surface_old = PVFF_surface
  !rhs at t0
  rhs = Calc_PVFFsurfacerhs(PVFF_surface_old)

  !first phase
  k1 = dt*rhs
  !calculate new rhs for next phase
  rhs = Calc_PVFFsurfacerhs(PVFF_surface_old+.5d0*k1)

  !second phase
  k2 = dt*rhs
  !calculate new rhs for next phase
  rhs = Calc_PVFFsurfacerhs(PVFF_surface_old+.5d0*k2)

  !third phase
  k3 = dt*rhs
  !calculate new rhs for next phase
  rhs = Calc_PVFFsurfacerhs(PVFF_surface_old+k3)

  !fourth phase
  k4 = dt*rhs

  !calculate new function at t=t0+tau=t1 using intermediate time evaluations
  !given by the four k's
  PVFF_surface = PVFF_surface_old + (1.d0/6.d0)*(k1+2.d0*k2+2.d0*k3+k4)
 END SUBROUTINE rk4_PVsurface

 !Function calculating right-hand-side of PV conservation equation
 FUNCTION Calc_PVrhs(dPV)
  IMPLICIT NONE
  DOUBLE PRECISION, DIMENSION(:,:), INTENT(IN) :: dPV
  DOUBLE PRECISION, DIMENSION(size(y),size(z)) :: Calc_PVrhs

  DOUBLE PRECISION :: SGterm, QGterm

  ! q does not change on y-boundaries and at z=50km. It does! at z=0
  !on surface z=0
  DO i=2,size(y)-1
   QGterm = -vg(i,j)*(dPV(i+1,1)-dPV(i-1,1))/(2.d0*dy)
```

```
   SGterm = -va(i,1)*(dPV(i+1,1)-dPV(i-1,1))/(2.d0*dy) - wa(i,1)*(dPV(i,2)-dPV(i,1))/dz
   IF (SG) THEN
    Calc_PVrhs(i,1) = QGterm + SGterm
   ELSE
    Calc_PVrhs(i,1) = QGterm
   ENDIF
  END DO

  DO i=2,size(y)-1
   DO j=2,size(z)-1
    QGterm = -vg(i,j)*(dPV(i+1,j)-dPV(i-1,j))/(2.d0*dy)
    SGterm = -va(i,j)*(dPV(i+1,j)-dPV(i-1,j))/(2.d0*dy) - wa(i,j)*(dPV(i,j+1)-dPV(i,j-1))/(2.d0*dz)
    IF (SG) THEN
     Calc_PVrhs(i,j) = QGterm + SGterm
    ELSE
     Calc_PVrhs(i,j) = QGterm
    ENDIF
   END DO
  END DO
 END FUNCTION Calc_PVrhs

 FUNCTION Calc_PVFFsurfacerhs(PV_surface)
  IMPLICIT NONE
  DOUBLE PRECISION, DIMENSION(:), INTENT(IN) :: PV_surface
  DOUBLE PRECISION, DIMENSION(size(y)) :: Calc_PVFFsurfacerhs

  DOUBLE PRECISION :: SGterm, QGterm

  DO i=2,size(y)-1
   QGterm = -vg(i,1)
   SGterm = -va(i,1)
   IF (SG) THEN
    Calc_PVFFsurfacerhs(i) = (QGterm+SGterm)*(PV_surface(i+1)-PV_surface(i-1))/(2.d0*dy)
   ELSE
    Calc_PVFFsurfacerhs(i) = QGterm*(PV_surface(i+1)-PV_surface(i-1))/(2.d0*dy)
   ENDIF
  END DO
 END FUNCTION Calc_PVFFsurfacerhs
END MODULE nummethods
```

References

T. Bergeron. Uber die dreidimensionale verknupfende wetteranalyse. *Geofys. Publ.*, 5(6):111 pp., 1928.

J. Bjerknes. On the structure of moving cyclones. *Mon. Wea. Rev.*, 47: 95–99, 1919.

J. Bjerknes and H. Solberg. Life cycles of cyclones and he polar front theory of atmospheric circulation. *Geofys. Publ.*, 3(1):1–18, 1922.

J.G. Charney. The dynamics of long waves in a baroclinic westerly current. *J. Meteor.*, 4:135–162, 1947.

E.F. Danielsen. Stratospheric-tropospheric exchange based on radioactivity, ozone aand potential vorticity. *J. Atmos. Sci.*, 25:502–518, 1968.

H.C. Davies and A.M. Rossa. Pv frontogenesis and upper-tropospheric fronts. *Mon. Wea. Rev.*, 126(6):1528–1539, 1998.

J. Donnadille, J Cammas, P Mascart, D. Lambert, and R. Gall. Fastex iop 18: A very deep tropopause fold. i: Synoptic description and modelling. *Quart. J. R. Met. Soc*, 127:2247–2268, 2001.

E.T. Eady. Long waves and cyclone waves. *Tellus*, 1 (3):33–52, 1949.

A. Eliassen. On the formation of fronts in the atmosphere. *The atmosphere and the sea in motion*, pages B. Bolin, Ed. Rockefeller Institute Press, 277–287, 1959.

A. Eliassen. On the vertical circulation in frontal zones. *Geofys. Publikasjoner*, 24(4):147–160, 1962.

A. Eliassen. Motions of intermediate scale: fronts and cyclones. *Advances in Earth Science*, pages P.M. Hurley, Ed. MIT Press, 111–138, 1966.

T.J. Galarneau Jr. `http://www.atmos.albany.edu/student/tomjr/images/realtime/Qvect700-NorAmer/res.html`, 2011.

B.J. Hoskins. The geostrophic momentum approximation and the semi-geostrophic equations. *J. Atmos. Sci.*, 32(2):233–242, 1975.

B.J. Hoskins. The mathematical theory of frontogenesis. *Ann. Rev. Fluid Mech.*, 14:131–151, 1982.

B.J. Hoskins and F.P. Bretherton. Atmospheric frontogenesis models: mathematical formulation and solution. *J. Atmos. Sci*, 29:11–37, 1972.

B.J. Hoskins, I. Draghici, and H.C. Davies. A new look at the ω-equation. *Quart. J. R. Met. Soc*, 104:31–38, 1978.

B.J. Hoskins, M.E. McIntyre, and A.W. Robertson. On the use and significance of isentropic potential vorticity maps. *Quart. J. R. Met. Soc*, 111: 877–946, 1985.

M. Juckes. The structure of idealized upper-tropospheric shear lines. *J. Atmos. Sci*, 56:2830–2845, 1999.

L.R. Lait. An alternative form of potential vorticity. *J. Atmos. Sci*, 51(12): 1754–1759, 1994.

C. Lu, P.E. Ciesielski, and W.H. Schubert. Geostrophic and ageostrophic circulations in midlatitude squall lines. *J. Atmos. Sci*, 54:1218–1227, 1997.

P. Markowski and Y. Richardson. *Mesoscale meteorology in midlatitudes.* John Wiley & Sons, Inc., May 2010.

S. McMillan. `http://einstein.drexel.edu/courses/Comp_Phys/Integrators/rk4.html`, 2011.

J.E. Miller. On the concept of frontogenesis. *J. Meteorol.*, 5:169–171, 1948.

G.W.K. Moore. The development of tropopause folds in two-dimensional models of frontogenesis. *J. Atmos. Sci*, 50(15):2321–2334, 1993.

S.E. Mudrick. A numerical study of frontogenesis. *J. Atmos. Sci*, 31:869–892, 1974.

V. Ostdiek and W. Blumen. Deformation frontogenesis: obseration and theory. *J. Atmos. Sci*, 52(9):1487–1500, 1995.

W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical recipes in Fortran 77 - second edition.* Bob Pirtle, 2005.

R.J. Reed. Advances in knowledge and understanding of extratropical cyclones during the past quarter century: an overview. *Extratropiccal cyclones: the Erik Palmen Memorial Volume*, pages C.W. Newton and E.O. Holopainen, Eds., Amer. Meteor. Soc., 27–45, 1990.

R.J. Reed and R.K. Sanders. An investigation of the development of a midtropospheric frontal zone and its associated vorticity field. *Ibid.*, 10: 338–349, 1953.

J.S. Sawyer. The vertical circulation at meteorological fronts and its relation to frontogenesis. *Proc. R. Soc. London Ser. A*, 234:346–362, 1956.

W.H. Schubert. Semigeostrophic theory. *J. Atmos. Sci*, 42(16):1770–1772, 1985.

W.H. Schubert, S.R. Fulton, and R.F.A. Hertenstein. Balanced response to squall lines. *J. Atmos. Sci.*, 46:2478–2483, 1989.

M.A. Shapiro. Turbulent mixing within tropopause folds as a mechanism for the exchange of chemical constituents between the stratosphere and troposphere. *J. Atmos. Sci.*, 37:994–1004, 1980.

R. Smith. Lectures on dynamical meteorology. `http://www.meteo.physik.uni-muenchen.de/~roger/Dm_Lectures/DM2007.pdf`, 2007.

A.J. Thorpe. Attribution and its application to mesoscale structure associated with tropopause folds. *Q. J. R. Meteorol. Soc.*, 123:2377–2399, 1997.

A.J. Thorpe and K.A. Emanuel. Frontogenesis in the presence of small stability to slantwise convection. *J. Atmos. Sci.*, 42:1809–1824, 1985.

G.J. Tripoli, C.M. Medaglia, S. Dietrich, A. Mugnai, G. Panegrossi, S. Pinori, and E.A. Smith. The 9-10 november 2001 algerian flood. *Bull. Amer. Meteorol. Soc.*, 86:1229–1235, 2005.

A. van Delden. The synoptic setting of thunderstorms in western europe. *Atmos. Res.*, 56:89–110, 2001.

W.T.M. Verkley. Salmon's hamiltonian approach to balanced flow applied to a one-layer isentropic model of the atmosphere. *Q. J. R. Meteorol. Soc.*, 127:579–600, 2001.

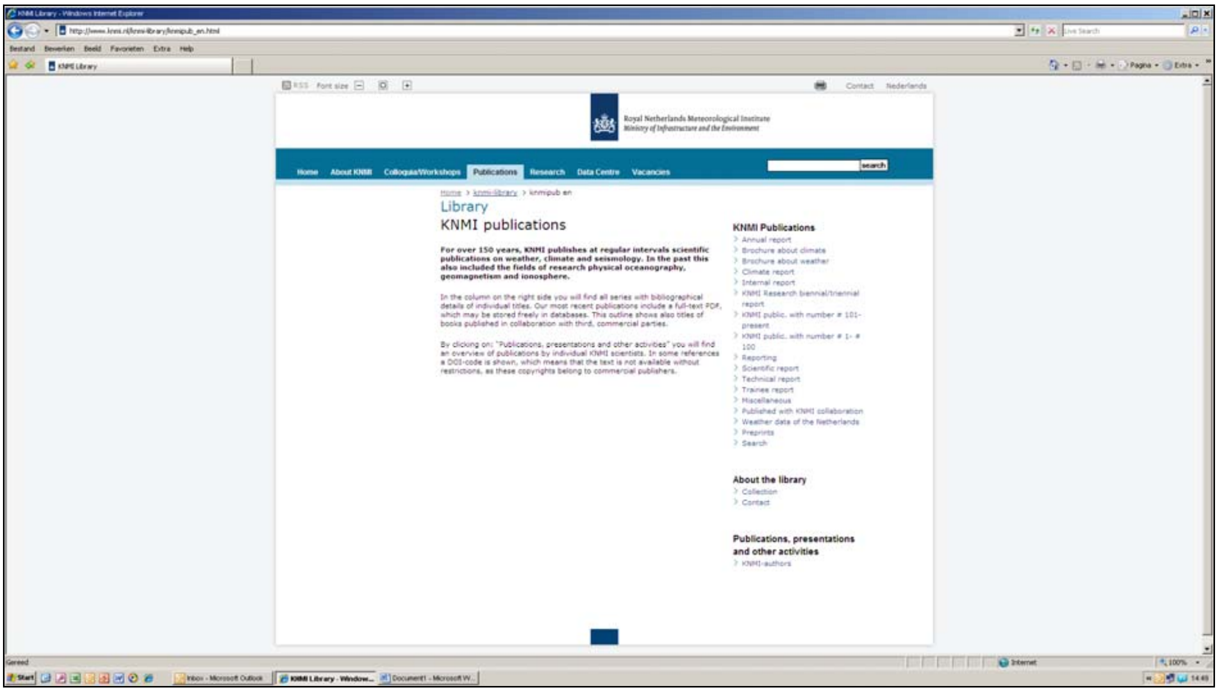
M.S. Wandishin, J.W. Nielsen-Gammon, and D. Keyser. A potential vorticity diagnsotic approach to upper-level frontogenesis within a developing baroclinic wave. *J. Atmos. Sci.*, 57(24):3918–3938, 2000.

R.T. Williams. Atmospheric frontogenesis: A numerical experiment. *J. Atmos. Sci.*, 24:627–641, 1967.

R.T. Williams. Numerical simulation of steady-state fronts. *J. Atmos. Sci.*, 31(5):1286–1296, 1974.

A complete list of all KNMI-publications (1854 – present) can be found on our website

www.knmi.nl/knmi-library/knmipub_en.html



The most recent reports are available as a PDF on this site.