Geoscientific
Model Development
Discussions

Open Access

3-D visualization of ensemble weather forecasts – Part 1: The visualization tool Met.3D (version 1.0)

M. Rautenhaus[1], M. Kern[1], A. Schäfler[2], and R. Westermann[1]

[1]Computer Graphics & Visualization Group, Technische Universität München, Garching, Germany
[2]Deutsches Zentrum für Luft- und Raumfahrt, Institut für Physik der Atmosphäre, Oberpfaffenhofen, Germany

Correspondence to: M. Rautenhaus (marc.rautenhaus@tum.de)

GMDD

8, 2101–2160, 2015

3-D visualization of ensemble weather forecasts – Part 1: Met.3D

M. Rautenhaus et al.

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

|◄ | ►|

◄ | ►

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Abstract

We present *Met.3D*, a new open-source tool for the interactive 3-D visualization of numerical ensemble weather predictions. The tool has been developed to support weather forecasting during aircraft-based atmospheric field campaigns, however, is applicable to further forecasting, research and teaching activities. Our work approaches challenging topics related to the visual analysis of numerical atmospheric model output – 3-D visualization, ensemble visualization, and how both can be used in a meaningful way suited to weather forecasting. Met.3D builds a bridge from proven 2-D visualization methods commonly used in meteorology to 3-D visualization by combining both visualization types in a 3-D context. We address the issue of spatial perception in the 3-D view and present approaches to using the ensemble to allow the user to assess forecast uncertainty. Interactivity is key to our approach. Met.3D uses modern graphics technology to achieve interactive visualization on standard consumer hardware. The tool supports forecast data from the European Centre for Medium Range Weather Forecasts and can operate directly on ECMWF hybrid sigma-pressure level grids. We describe the employed visualization algorithms, and analyse the impact of the ECMWF grid topology on computing 3-D ensemble statistical quantitites. Our techniques are demonstrated with examples from the T-NAWDEX-Falcon 2012 campaign.

1 Introduction

Weather forecasting requires meteorologists to explore large amounts of numerical weather prediction (NWP) data, and to assess the reliability of the predictions. Visualization methods that facilitate fast and intuitive exploration of the data hence are of particular importance. In practice, the forecasting process for the most part relies on two-dimensional (2-D) visualization methods. Meteorologists use weather maps, vertical cross-sections and a multitude of meteorological diagrams to depict the data. From these image sources, they build "mental models" of the three-dimensional (3-D), time-

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

◄◄ | ►►

◄ | ►

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

varying forecast atmosphere inside their heads (Hoffman and Coffey, 2004; Trafton and Hoffman, 2007).

Despite the 3-D nature of the atmosphere, 3-D visualization methods have not found widespread usage, even though there have been promising attempts in the 1990s and early 2000s that suggested added value (Treinish and Rothfusz, 1997; Koppert et al., 1998; McCaslin et al., 2000). Various hindering factors are discussed in the literature, including resistence of forecasters to adapt to new 3-D visualization methods that are decoupled from their "familiar" 2-D products (Koppert et al., 1998; Szoke et al., 2003), problems with spatial perception in 3-D renderings (Szoke et al., 2003), as well as issues due to limited performance (Treinish and Rothfusz, 1997) and the need for dedicated graphics workstation hardware (Koppert et al., 1998).

In addition to 3-D space and time, forecast visualization has in recent years become more challenging through the increased use of ensemble weather predictions (sets of forecast runs whose distribution provides information on forecast uncertainty, e.g. Gneiting and Raftery, 2005; Leutbecher and Palmer, 2008). Ensemble products have become a major tool to assess forecast reliability. The development of visualization methods that depict the uncertainty derived from ensemble data is an active topic of research not only for weather forecast ensembles (Obermaier and Joy, 2014). Yet again, ensemble visualization techniques related to weather forecasting published so far mainly focus on two dimensions as well (e.g. Potter et al., 2009; Sanyal et al., 2010).

In this article we introduce a new open-source visualization tool, *Met.3D*, that provides interactive 3-D visualization techniques for ensemble prediction data. There has been an immense progress in mainstream graphics hardware capabilities in recent years. Making use of these developments, Met.3D facilitates interactive visualization of present-day NWP datasets on consumer hardware. The tool has been developed as a new effort to demonstrate the feasibility of using 3-D visualization for forecasting, this time also considering uncertainty information from ensemble datasets. It is intended to be used for actual forecasting tasks, as well as a platform to implement and evaluate new 3-D and ensemble visualization techniques.

The work presented in this paper has been inspired by a particular application, forecasting the weather situation to plan research flight routes during aircraft-based field campaigns. We focus on this application throughout the paper at hand. However, Met.3D is applicable to a broader range of forecasting and visual data analysis tasks. Both fast exploration and uncertainty assessment play a major role in campaign forecasting:

1. When investigating suitable meteorological conditions to specify the route of a research flight (that is, waypoints in 3-D space and time), the forecaster is required to examine the NWP data in a short period of time. Atmospheric features relevant to the flight have to be identified quickly, and findings have to be communicated to colleagues. Upper-level features typically important to research flights with high-flying aircraft are of an inherently three-dimensional nature (for example, clouds, jet streams, or the tropopause). From our experience in campaigns with DLR (German Aerospace Centre) involvement, visualization used during campaigns has been solely based on 2-D methods, typically with limited interactivity. We are hence interested in investigating how 3-D visualization methods and interactivity (to quickly navigate the data space) can be used to aid the exploration.

2. Assessing the forecast's uncertainty has become indispensable as flights frequently have to be planned multiple days before take-off (typically three to seven days; the medium forecast range) to obtain the required approval from air traffic authorities. To the best of our knowledge (concerning field campaigns with DLR involvement), ensemble predictions have not been used for flight planning until now. However, they provide valuable information; for example, 3-D probability fields for the occurrence of a targeted atmospheric process or feature can be derived. Potential flight routes can be planned in regions in which the probability is high. Ensemble forecasts are readily available, for example, from the European Centre for Medium Range Weather Forecasts (ECMWF). An open question, how-

3-D visualization of ensemble weather forecasts – Part 1: Met.3D

M. Rautenhaus et al.

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

◄◄ | ►►

◄ | ►

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

ever, is how the ensemble data can be visualized to improve flight planning in the medium forecast range.

Our objective for the work presented in this paper is to use interactive 3-D visualization of ECMWF predictions to improve the forecast process for field campaigns. The work has been stimulated by the forecast requirements of a specific field campaign, the international T-NAWDEX-Falcon campaign (THORPEX – North Atlantic Waveguide and Downstream Impact Experiment – Falcon, hereafter TNF). TNF took place in October 2012 with the objective to take in-situ measurements in warm conveyor belts (WCBs), airstreams in extratropical cyclones that lift warm and moist air from near the surface to the upper troposphere (Browning and Roberts, 1994). Schäfler et al. (2014) provide details on the campaign and its flight planning. The major forecasting challenge was to predict the likelihood of WCB occurrence within aircraft range. This was expressed by a number of forecast questions that guided the development of Met.3D (the forecaster needs to be able to answer these questions with the tool):

  – FQ-A: How will the large scale weather situation develop over the next week, and will conditions occur that favour WCB formation?

  – FQ-B: How reliable are the weather predictions?

  – FQ-C: Where and when, in the medium forecast range and within the range of the aircraft, is a WCB most likely to occur?

  – FQ-D: How reliable is the forecast of WCB occurrence?

  – FQ-E: Where will the WCB be located relative to cyclonic and dynamic features?

In a recent ECMWF Newsletter article (Rautenhaus et al., 2014), we provided a brief overview of our work. It is the purpose of this publication to describe the techniques we have developed in detail and to present our solutions to particular challenges.

We split our work into two parts, structured as follows. In the paper at hand, we introduce Met.3D. We discuss challenges related to interactive 3-D visualization and present techniques that address (FQ-A) and (FQ-B).

To put our work in the context of the literature, we review recent works in meteorological and ensemble visualization in Sect. 2. Section 3 presents Met.3D's visualization capabilities. When introducing 3-D visualization to forecasting, we need to consider that the 2-D visualization methods commonly used in meteorology provide many advantages (for example, spatial perception) and that meteorologists are used to working with them. In a 3-D forecast tool to be used in practice, we hence have to be careful not to replace proven 2-D methods, but to put them into a 3-D context and to use 3-D visualization to add value. We address the challenges of creating such a "bridge" from 2-D to 3-D visualizations, of improving spatial perception of 3-D renderings and of designing interactive methods that provide fast and easy visual access to ensemble information. For 3-D depictions, we propose *normal curves* to visualize the structure inside a transparent 3-D isosurface. The method provides an intermediate means between a 2-D section and a 3-D isosurface.

Section 3 contains short examples of the proposed visualization techniques. A supplementary video shows real-time screen recordings. The examples focus on the visualization capabilities of Met.3D and demonstrate its performance on mid-range consumer hardware.

Sections 4 and 5 address technical aspects. To avoid time consuming preprocessing of the forecast data prior to visualization, Met.3D operates directly on the ECMWF terrain-following model grid. The characteristics of the ECMWF data and resulting challenges for visualization are discussed along with Met.3D's visualization algorithms and system architecture in Sect. 4. Section 5 discusses a challenge that arises from aiming at interactive ensemble visualization: the efficient yet accurate computation of statistical quantities from the ensemble predictions. For our application, the ECMWF model grid has an unfavourable property. When computing statistical quantities on a per-grid-point-basis an error is introduced, since the vertical positions of the grid points vary between members. Regridding to a common grid is a solution, albeit time consuming and hence undesirable for real-time visualization. We analyse the error introduced when ignoring such a regridding and provide advice on how to handle the issue.

3-D visualization of ensemble weather forecasts – Part 1: Met.3D

M. Rautenhaus et al.

Section 6 provides information on code availability, before the article is concluded in Sect. 7.

In the second part of this study (Rautenhaus et al., 2015, hereafter R15P2), we address (FQ-C) to (FQ-E). A method to compute 3-D WCB probabilities from Lagrangian particle trajectories is introduced and evaluated, and Met.3D is extended by a technique to visually analyse the derived probabilities. To demonstrate the added value of 3-D visualization for forecasting, we present a comprehensive case study with detailed meteorological interpretations of a forecast case of TNF. The case study uses methods from both papers and illustrates how Met.3D can be used in practice. Readers primarily interested in the application of Met.3D should read Sect. 3 in this part, skip the technical sections and proceed to the case study in R15P2.

2 3-D and ensemble visualization in meteorology

Our work is related to 3-D visualization in meteorology and to uncertainty and ensemble visualization. While for the latter a large body of articles is available in the visualization literature, only little has been published pertaining to meteorological 3-D visualization. This is particularly true with respect to application in forecasting.

2.1 3-D visualization in meteorology

Visualization tools in meteorology can be distinguished with respect to application in a research setting and application in an operational forecast setting. As Koppert et al. (1998) point out, a tool in an operational setting should offer techniques tailored to the specific forecasting task and not confuse the forecaster with large sets of parameters that need to be configured. A research setting, on the other hand, demands a tool that is flexible to adapt to different exploration tasks and data formats. Its visualizations should be highly configurable by the user.

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

◄ | ►

◄ | ►

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

In forecasting, 2-D visualization systems prevail. With respect to field campaigns with DLR involvement, the Mission Support System (MSS) is frequently used, a tool that generates horizontal and vertical 2-D sections of the forecast data upon user request (Rautenhaus et al., 2012). This tool motivated the design of our proposed bridge from 2-D to 3-D that we describe in Sect. 3. Further 2-D systems that have been applied include the German Weather Service (DWD) NinJo workstation (Heizenrieder and Haucke, 2009) and the ECMWF Metview software (Russell et al., 2010).

The few reports on the usage of 3-D visualization in forecasting date to the 1990s and early 2000s. Treinish and Rothfusz (1997) and Treinish (1998) reported on experiments with 3-D visualization for local forecasting during the 1996 Olympic Games in Atlanta. They concluded that an advantage of their 3-D methods was "that they virtually eliminated the need to laboriously evaluate numerous two-dimensional images", however, noted a lack of interactivity due to limitations in computational performance. Lux and Frühauf (1998) and Koppert et al. (1998) presented *RASSIN* and its successor *VISUAL*, a 3-D forecasting system for usage within the DWD. Discussing their experience with an operational test of the software, they, too, point out the importance of system performance for user acceptance. They furthermore highlight the need for common concepts of operations (user interface and workflow) when forecasters are asked to transition from a 2-D to a 3-D environment.

McCaslin et al. (2000) presented *D3D*, a 3-D software built at the United States Forecast Systems Laboratory (FSL) on top of the *Vis5D* tool (Hibbard and Santek, 1990). D3D's user interface was designed to match that of the 2-D *D2D* software in use at the National Weather Service Weather Forecast Offices (WFOs). "Real-time forecast exercises" were conducted to evaluate the value of 3-D visualization, and the software was installed at a number of WFOs. Szoke et al. (2003) report on experiences gained with the system. They discuss the reluctance of forecasters to switch from 2-D to 3-D, but also confidently state that for forecasters trained with D3D it is "hard to deny that examining the atmosphere using a 3-D tool is not more effective and complete than using 2-D displays". Szoke et al. (2003) also positively report on the

3-D visualization of ensemble weather forecasts – Part 1: Met.3D

M. Rautenhaus et al.

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

⏮ | ⏭

◀ | ▶

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

interactivity introduced by their system. Interactively moveable vertical soundings and cross sections, for example, were very well perceived by the forecasters.

With respect to research environments, 3-D visualization is more frequently used. Vis5D, mentioned above, was widely used into the 2000s, however, its development was discontinued. More recently, prominent tools include *Vapor* (Norton and Clyne, 2012; Clyne et al., 2007) and the Unidata Integrated Data Viewer IDV (Murray and McWhirter, 2007; Murray et al., 2009). Vapor is an open-source 3-D visualization software developed at the United States National Centre for Atmospheric Research. It features a number of 3-D visualization techniques to view time varying gridded datasets, however, does not provide techniques for ensemble data or forecasting functionality. IDV is a comprehensive Java application for the analysis and visualization of geosciences data. It supports a variety of visualization methods, including some 3-D support. For example, Yalda et al. (2012) use IDV's 3-D capabilities for interactive immersion learning. On a broader scope, *Paraview* (Henderson et al., 2004) is a general-purpose visualization tool that can also be used with meteorological data. In the context of a graduate university course, Dyer and Amburn (2010) investigated how Paraview can be used in a meteorological setting.

A major reason why 2-D methods are often preferred in the atmospheric sciences is that they are well suited to convey quantitative information, as Middleton et al. (2005) point out in a survey of visualization in meteorology. 2-D contour lines and colour mappings can be used to convey a large range of data values. In a 3-D depiction, only a small number of isosurfaces can be displayed without cluttering and occlusion. However, a 3-D image is able to convey spatial structure in all three dimensions, a distinct advantage compared to 2-D methods. On the downside, spatial perception is more challenging in 3-D. Determining the location of a data feature displayed in a 2-D image is usually not an issue. In a 3-D projection, achieving good spatial perception can be difficult. Major influencing factors are, for example, shadows (Wanger et al., 1992) and illumination models (e.g. Weigle and Banks, 2008; Lindemann and Ropinski, 2011, and references therein). The issue is also noted by Szoke et al. (2003). As an approach,

Title Page

Abstract   Introduction

Conclusions   References

Tables   Figures

◄◄   ►►

◄   ►

Back   Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

they have implemented a switch to an overhead view and a vertically moveable map in D3D to enable the forecaster to better judge the spatial position of a 3-D feature.

2.2 Ensemble visualization

Ensemble visualization aims at identifying variability, similarities, and differences among ensemble members. It is closely related to uncertainty visualization, of which Pang et al. (1997) and Johnson and Sanderson (2003) provide early overviews. In the atmospheric sciences, 2-D visualizations of statistical quantities that summarize the ensemble distribution or that represent relative frequencies for events are frequently used. Wilks (2011, Ch. 7.6.6) lists a number of techniques. For example, current products provided in ECMWF's *ecCharts* system (Lamy-Thépaut et al., 2013) include maps of mean and standard deviation (SD), maps of threshold probabilities (for example, the probability of precipitation exceeding a critical threshold) and of derived statistical measures (for example, the extreme forecast index, Lalaurette, 2003).

In a recent survey–also including applications outside the atmospheric domain–, Obermaier and Joy (2014) classify ensemble visualization methods described in the literature into *location-based methods* and *feature-based methods*. Location-based methods compare ensemble properties at fixed locations in the dataset. In the simplest case, this includes the ensemble mean, SD, or probability as computed at a given grid point. Such statistical quantities have been visualized via colour maps, opacity, texture, and animation (Djurcilov et al., 2002; Rhodes et al., 2003; Lundstrom et al., 2007). Also, glyphs have been used to display, for example, uncertainty in wind fields (Wittenbrink et al., 1996). Feature-based methods, on the other hand, extract features from each ensemble member and aim at visually comparing the detected features. Examples include spaghetti plots (where the isolines are the features), the joint display of detected cyclonic features (Hewson and Titley, 2010), and visualization techniques for the prediction of hurricane tracks (Cox et al., 2013). Recently, Whitaker et al. (2013) have generalised boxplots to contour boxplots to enable an improved quantitative and qualitative analysis of ensembles of 2-D isocontours and level-sets. In 3-D, the effect of

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

|◄ | ►|

◄ | ►

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

uncertainty on the position of 3-D isosurfaces has been the topic of a number of studies. It has been approached with, for instance, geometric displacements (Grigoryan and Rheingans, 2004) and surface animation (Brown, 2004). In a study concerning the reconstruction of the earth's subsurface model, Zehner et al. (2010) visualize confidence intervals around an isosurface using additional transparent surfaces as well as lines connecting the surfaces. Recently, techniques have used stochastic modelling of uncertainty in scalar ensembles to quantify and visualize the possible occurrences of isosurfaces (Pöthkow and Hege, 2011; Pöthkow et al., 2011; Pfaffelmoser et al., 2011; Pfaffelmoser and Westermann, 2012). The latter studies all include examples from the atmospheric domain.

A few articles in the visualization literature have presented software tools that put special emphasis on ensembles in earth-science applications. Potter et al. (2009) present the *Ensemble-Vis* tool and investigate the usage of multiple linked views to visualize 2-D weather simulation ensembles. They conclude that the combination of standard statistical displays (spaghetti plots, maps of mean and SD) with user interaction facilitates clearer presentation and simpler exploration of the data. In their *Noodles* tool, Sanyal et al. (2010) enhance spaghetti plots by glyphs and confidence ribbons to highlight the Euclidean spread of 2-D contour ensembles. They describe the usage of their methods by atmospheric researches investigating different parametrisations in the Weather Research and Forecasting (WRF) model. Sanyal et al. also highlight the positive effect of interactivity and linked views on the user and note the challenge of potential generalization of their work to three dimensions. Recently, Höllt et al. (2014) have presented *Ovis*, a system for the visualization of 2-D ocean heightfield ensemble data. They again use linked views of maps, statistical plots and 3-D renderings and demonstrate the use of time-series glyphs for the comparative visualization of the ensembles at two different positions over time. Höllt et al. discuss the application of their tool to off-shore oil operations and the planning of underwater glider paths.

3 The 3-D ensemble visualization tool Met.3D

Met.3D has been developed to support ensemble data exploration during forecasting; at the time of writing in particular for field campaigns. Beside this primary objective, we have designed the software in a way that it can be used as a framework into which new ensemble visualization techniques can be implemented and evaluated with respect to their use in forecasting. We note that Met.3D is not intended to be a full-featured meteorological workstation; this would be beyond the scope of our work.

At the time of writing, Met.3D supports forecast data from the ECMWF Ensemble Prediction System (ENS), comprising 50 perturbed forecast runs and an unperturbed control run (Buizza et al., 2006; Miller et al., 2010). These 51 forecast members approximate the distribution of possible future weather scenarios (Leutbecher and Palmer, 2008).

The visualization examples shown in this paper use data from the TNF forecast case of 19 October 2012. The satellite image in Fig. 1 provides a real-world observation of major features that appear in the visualizations: a distinct narrow trough was located to the west of the British Isles. Upstream of the trough the former Hurricane *Rafael* transformed into a strong midlatitude cyclone. East of the trough, ascending WCB airmasses formed a cloud band extending from Spain to the British Isles. The clouds further stretch along a jet stream over southern Scandinavia and the Baltic Sea.

The static images shown in the following sections are complemented by video clips contained in the Supplement to the paper, helping to illustrate the interactive capabilities of Met.3D. The videos are screen recordings realised on hardware consisting of a consumer-class six-core Intel Xeon running at 2.67 GHz, equipped with 24 GB of RAM, a 512 GB solid state drive and an Nvidia GeForce GTX 560Ti graphics card with 2 GB of video memory.

3.1 User interface

Figure 2 shows the graphical user interface (GUI) of Met.3D. The forecast data fields can be displayed in multiple 3-D views (Fig. 2a, b, c). In the horizontal, a cylindrical longitude–latitude projection is used. As common in meteorology, the logarithm of pressure serves as the vertical coordinate. Vertical scale, i.e. the proportion of vertical to horizontal units, can be specified for each view individually. Time navigation is provided for the forecast initialisation (or base) time and the forecast valid time (Fig. 2d). This way, subsequent forecast runs can be checked for consistency by keeping the valid time fixed and changing the initialisation time. A distinct feature is the ensemble navigation. The user can select a specific forecast member for exploration, animate over members and toggle the ensemble mean for all currently displayed data fields (Fig. 2e).

Visual entities such as a horizontal or vertical cross-section, the base map or a 3-D isosurface are represented by *actors* and are assigned to a *scene*. A scene, in other words a collection of actors, can be assigned to one of the views for rendering. An actor can be part of multiple scenes. For example, a cross-section could be viewed as a traditional 2-D image in one view, and be combined with a 3-D isosurface in another. If the section is relocated, its position is updated in both views. To keep the user interface simple, properties that the user can modify for a particular actor (e.g. the isovalue of an isosurface, the forecast variable displayed by an actor, the associated colour palette) are arranged in a tree-like structure on the left of the Met.3D window and are easily accessible (Fig. 2f). If used in a forecast setting, only the uppermost tree nodes are required by the user to, for instance, load predefined forecast products.

Trafton and Hoffman (2007) point out the importance of visual comparisons in the forecasting process. Met.3D's actors can be synchronized in time and ensemble dimension, its views can be synchronized to the same camera viewpoint. Thus, side-by-side comparison of different datasets is facilitated.

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

◄◄ | ►►

◄ | ►

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

3.2 A bridge from 2-D to 3-D

To help forecasters transition to the 3-D visualization environment, we have implemented horizontal and vertical 2-D sections. The sections reproduce the look of the corresponding products in the DLR MSS (Rautenhaus et al., 2012), providing filled and line contours, wind barbs, coast lines and graticule. In Met.3D, the sections are embedded into the 3-D context and can be interactively moved in space by the user in real-time. This provides a very fast means to explore the atmosphere's vertical structure (by sliding a horizontal section up and down), or the change in forecast variables along a flight track when a waypoint is relocated (by moving a vertical section). Also, the camera can be moved interactively to zoom in, pan, or tilt the view – for instance, to view multiple sections stacked on each other from an angled viewpoint. Figure 3 illustrates the concept. The forecast wind field is visualized by means of a horizontal and vertical section. The horizontal map – largely resembling the corresponding product from the MSS – is stacked on top of surface level contours displaying the mean sea level pressure (Fig. 3b). The vertical section is augmented by a 3-D isosurface of wind speed (Fig. 3c); the isovalue is chosen such that the strongest winds of the jet stream, an important indicator for the large scale flow of the upper troposphere, are captured. The 3-D display allows us to locate the vertical section in space and additionally provides information on the spatial structure of the jet.

We approach the challenge of spatial perception by drawing projections of all rendered structures to the surface to imitate shadows generated by a light source above the scene. As illustrated in Fig. 3b and c, the shadows help to qualitatively judge the elevation of a feature, and also show its horizontal location. To improve the quantitative judgement of elevation, the user can colour the isosurface according to pressure elevation, and place vertical poles in the scene that provide labelled pressure axes (Fig. 3c). The poles can be interactively moved in the scene, so that different locations can be probed.

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

|◄ | ►|

◄ | ►

Back | Close

Full Screen / Esc

Vertical sections can be drawn along an arbitrary number of waypoints (Fig. 3c). They can also be moved synchronously in multiple scenes, as illustrated in Fig. 4. Displayed are sections of potential vorticity (Fig. 4a, the red colours around values of 2 PVU show the dynamic tropopause) and cloud cover fraction (Fig. 4b). Wind barbs overlain on a horizontal section can be configured to automatically scale in size and density. In Fig. 5, the horizontal section of equivalent potential temperature shows the different character of airmasses transported by Rafael. When the user zooms into the view, Met.3D increases the density of the wind barbs (Fig. 5b). The frontal zone along which the typical change in wind direction occurs can now be well perceived.

With respect to colours used in the visualizations, it is important to address perceptual issues (Hoffman et al., 1993). To map scalar value to colour, we have implemented the perceptually-based Hue-Chroma-Luminance (HCL) colour space. Following Zeileis et al. (2009) and Stauffer et al. (2013), the user can create colour palettes by specifying ranges in hue, chroma and luminance. Alternatively, colours can be explicitly specified to reproduce colour bars the user is familiar with. An example is the colour palette for potential vorticity shown in Fig. 4.

3.3 Ensemble support

Met.3D enables the forecaster to explore variation in the ensemble, to identify regions in which the forecast is uncertain, and to explore possible forecast scenarios. The user can interactively navigate through the ensemble members to judge the variability in the forecast. Each member can also be explored individually. Statistical measures including threshold probabilities, mean, minimum, maximum and SD can be derived on-demand. For threshold probabilities (for example, wind speed exceeding $45\,\mathrm{m\,s^{-1}}$ or cloud cover fraction being below 0.2) the threshold value can be adjusted interactively.

Figure 6 shows an example of exploring the upper level ensemble wind field of the forecast from Monday, 15 October 2012, 00:00 UTC, valid at Friday, 19 October 2012, 18:00 UTC. To visualize the jet stream, two wind speed isosurfaces are rendered. The large variation of the ensemble regarding position, structure, and strength of the jet

stream over the Atlantic highlights high uncertainty in this area. On the other hand, the strong jet extending from Spain to Scandinavia is predicted with higher certainty: while in the mean wind field the $45\,\mathrm{m\,s^{-1}}$ signal over the Atlantic is largely smoothed out, it is present over Europe (Fig. 6d). However, adding a horizontal section of wind speed SD (Fig. 6e) to the isosurface of mean wind speed reveals that the position of the jet is uncertain in particular on its northern side.

Figure 7 shows the probability of wind speed exceeding $45\,\mathrm{m\,s^{-1}}$. A high probability of over 70 % can again be found over northern Europe (Fig. 7a). The large horizontal extent of the area of low (10 %) probability above the Atlantic reflects the uncertainty. The actual jet can occur anywhere in this region. Two days later, with decreasing forecast lead time, the ensemble has significantly converged and the uncertainty has decreased (Fig. 7b).

Figure 7c and d shows the probability of the Schmidt–Appleman criterion (Schumann, 1996), an indicator for the occurrence of contrails (aircraft-induced clouds that also have been the target of research flights; Voigt et al., 2010; Kaufmann et al., 2014). Visualization of the probability of the Schmidt–Appleman criterion being fulfilled shows that contrails, in the example, can only occur between about 400 and 200 hPa. In the given case, a high probability can be observed on the leading downstream edge of the jet.

3.4 Normal curves

In the volume visualizations shown in Figs. 6 and 7, the structure of the scalar fields inside the transparent isosurfaces cannot easily be inferred. As stated in Sect. 2.1, this is a disadvantage of 3-D visualization: While an isosurface allows inference on the three-dimensional spatial structure of the displayed data field, it only displays a single data value. Although two or three isosurfaces can be rendered in a single image using transparency, the image quickly becomes illegible when more surfaces are used. *Normal curves* were suggested by Pfaffelmoser et al. (2011) to estimate the spatial

distance between two isosurfaces. For our application, we propose to use *3-D normal curves* to visualize the structure of scalar fields in the interior of an isosurface.

The curves are started on a transparent isosurface and proceed along the field's gradient direction, i.e. normal to the isosurface. We colour the curves according to the scalar value. This way, we achieve a visual sampling of a subdomain of the volume. In contrast to a 2-D section that samples a planar subdomain, the normal curves sample a 3-D subdomain enclosed by an isosurface via a discrete set of lines. Following the gradient, the curves converge at local extrema of the data field. This way, the user can at a glance identify the locations and strengths of present extrema, and judge the strength and direction of the gradient between an extremum and the outer isosurface.

Figure 8 illustrates the approach. The goal is to identify regions of maximum probability of cloud ice water content exceeding $0.01\,\mathrm{g\,kg^{-1}}$, and to track the regions' evolution over time. The normal curves immediately show a maximum in the upper part of the transparent 40 % isosurface (Fig. 8b and c). The corresponding shadows reveal that the maximum is approximately located above the Pyrenees. Interaction with the vertical axis shows a vertical position between 300 and 200 hPa. Further visual aids can now be added to obtain more quantitative information. In the example, the horizontal section can be immediately placed in the region of interest, without the need to search the entire vertical extent of the model atmosphere (Fig. 8d).

While extrema can also be identified with an inner opaque isosurface (cf. Fig. 7) or by interacting with 2-D sections, the normal curve approach requires less interaction steps. This is advantageous if the absolute values of the extrema are not known beforehand (with isosurfaces the user needs to search over isovalues), and if the extrema shall be visually tracked over ensemble members or time. Concerning time, in particular probability values tend to decrease with increasing forecast lead time, hence a fixed isosurface is not well suited to visualize the temporal evolution of a maximum.

In Fig. 2c (also shown in the video at 05:40 min), the method is applied to the upper level wind field shown in Fig. 6. Here, the normal curves inside the $45\,\mathrm{m\,s^{-1}}$ isosurface

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

◄◄ | ►►

◄ | ►

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

converge to the string-like line of local maxima in the wind field – the curves are used to identify the position of the jet core and its strength.

4 Visualization algorithms and system architecture

Response time, the time required to display a new image after the user has interacted with, for example, camera or timestep, is crucial to the acceptance of an interactive visualization tool, as Szoke et al. (2003) and Hibbard (2004) emphasize. To achieve low response times, we make extensive use of modern graphics processing units (GPUs). These highly parallel processors provide high computational throughput and memory bandwidth and are well suited to accelerate visualization algorithms.

GPU acceleration is implemented with OpenGL 4 and the OpenGL Shading Language (GLSL)[1], using vertex, geometry, fragment and compute shaders. These small GPU programs allow the parallel execution of operations on the level of a graphics vertex or of an output fragment (i.e. a single pixel in the generated image), the generation of new geometry by the graphics subsystem, or the general parallel execution of operations. We will not go into detail of graphics technology here, for an introduction to GPU based visualization we refer the reader to, for example, Bailey (2009, 2011, 2013) or Engel et al. (2006). On the CPU side, Met.3D is implemented in C++.

A second important factor influencing response time is the way data is read from disk and whether and how it needs to be processed prior to visualization. We have designed an ensemble data pipeline to handle this task efficiently.

In this section, we discuss the methods used to achieve high visualization performance in Met.3D. After describing the data that can be handled by the tool (Sect. 4.1), we discuss the ensemble data pipeline (Sect. 4.2) and the GPU-based visualization algorithms (Sects. 4.3 and 4.4).

---

[1]https://www.opengl.org/documentation/glsl/

4.1 Forecast data

The data upon which we have based our visualization methods are obtained from the ECMWF global ensemble weather prediction system ENS and the high-resolution deterministic integrated forecast system IFS. One of our system design goals was to support the forecast data in the format they can be retrieved from the ECMWF Meteorological Archive and Retrieval System (MARS). MARS outputs the data interpolated in the horizontal to a regular latitude/longitude grid. In the vertical, the data is available on either a set of pre-defined pressure levels (PL), or, higher resolved and thus better suited for 3-D visualization, on the native model grid levels (ML). For the latter, the model uses terrain following hybrid sigma-pressure coordinates, as illustrated in Fig. 9. The vertical pressure coordinate p_k of a grid point at level k is defined by a set of fixed coefficients a_k and b_k and the surface pressure p_{sfc} below the grid point (Untch and Hortal, 2004): $p_k = a_k + b_k \times p_{sfc}$. With increasing altitude the influence of p_{sfc} decreases. During TNF, the operational ensemble forecast was available with 62 levels (91 levels for the deterministic forecast, increased by the time of writing to 137 levels). At this resolution, levels are constant in pressure above approximately 64 hPa (70 hPa)[2]. In the horizontal, a spherical truncation of T639 (T1279) is available, corresponding to a regular latitude/longitude grid of approx. 0.28° by 0.28° (0.15° by 0.15°). Forecasts are available twice daily (starting at 00:00 and 12:00 UTC) at a time step of three hours up to 144 h forecast lead time and six hours up to 240 h forecast lead time.

For the examples in this article, we use ENS data interpolated horizontally to 1° × 1° and to 0.25° × 0.25°. 1° × 1° is the resolution we were able to operationally retrieve during TNF, as permitted by the available internet bandwidth and interpolation time required by MARS. Deterministic data is used at 0.15° × 0.15° resolution. In the vertical, all 62 and 91, respectively, levels are used.

The forecast domain used in the examples encompasses 100° in longitude by 40° in latitude, resulting in 101 × 41 × 62 grid points for ENS data fields at 1° × 1° resolution,

---

[2]http://old.ecmwf.int/products/data/technical/model_levels/

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

◀ | ▶

◀ | ▶

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

401 × 161 × 62 points at 0.25° × 0.25° resolution, and 669 × 268 × 91 points for the deterministic forecast at 0.15° × 0.15° resolution. Using floating point precision (4 bytes per value), the data fields require approximately 1, 16 and 62 MB per member, timestep, and forecast parameter in graphics memory. For visualizations using multiple forecast parameters and the entire ensemble, the required memory quickly adds up.

Forecast data can be read directly from GRIB files output by MARS or from NetCDF-CF[3] files. Our goal was to minimise the time span between data availability at ECMWF and visualization. Hence, no preprocessing of the data prior to usage in Met.3D is required. Forecast parameters not output by the ECMWF model, however, need to be computed first. For this purpose, Met.3D can be connected to the data processing system of the DLR MSS, which derives additional quantities (for example, relative humidity and potential vorticity) from the forecast parameters output by ECMWF.

4.2 Ensemble processing pipeline

To process the ensemble data prior to rendering, we have designed a data processing pipeline composed of modules (*data sources*) that create, read or process data and that can be combined in flexible ways. Figure 10 illustrates the concept. Algorithms in the data sources (for example, ensemble statistics or trajectory filtering, cf. R15P2) can be implemented to execute on either CPU or GPU (the latter via compute shaders). All data sources are connected to a memory manager that caches intermediate results. The actors that implement the visualization methods are placed at the end of a pipeline. They send *requests* into the pipeline to obtain a specific data item. These requests are composed of multiple key/value pairs similar to the Web Map Service requests used in the MSS (see Rautenhaus et al., 2012, for details). A request emitted into a pipeline propagates from data source to data source. Each data source interprets the keys it requires. If the requested operation has been executed before and the result has been cached, no action is taken. Otherwise, the data source defines a processing task to

---

[3]http://cfconventions.org/

perform the requested operation. The task, however, is not executed immediately. If applicable, remaining keys are passed on to the data source's input(s). If a data source requires additional input, it can also append keys to the request.

All processing tasks defined this way are assembled into a task graph that is passed to a scheduler for execution. Based on the dependencies provided by the graph structure and information carried by the tasks, the scheduler can process the tasks. For example, tasks that have to be performed for all members of the ensemble can be executed in parallel.

As an example, consider the pipeline depicted in Fig. 10b. The volume actor at the end of the pipeline emits a request for a scalar field containing the probability of horizontal wind speed exceeding $45\,\mathrm{m\,s^{-1}}$. The module computing the probability field requires the wind field of each ensemble member, regridded to a common grid. Hence, requests for regridded data fields containing the members' wind speed are emitted and a task is set up to compute the probability from these fields. The regridding module, in turn, requests that the wind speed fields are read from disk by the reader module. For an ensemble of size M, the resulting task graph (Fig. 10c) contains M tasks to read the wind field of a single member, M tasks to regrid these fields to a common grid, and one task to compute the probabilities. The regridding tasks are well suited to be executed in parallel.

To indicate an order of magnitude of the response times that Met.3D achieves when the displayed data field is changed, Table 1 lists timings for changing the forecast time in the horizontal section in Fig. 3. Timings are provided for displaying a single member of the ensemble and for displaying the ensemble mean, both when data needs to be read from disk and when it is available in cache. We note that comprehensive optimisations of the system performance were outside the scope of this project and are left for future work. However, the architecture is prepared to implement, for example, pre-loading techniques to hide disk access.

4.3 GPU based visualization algorithms

Met.3D's visualization algorithms support data fields on both hybrid sigma-pressure levels and on pressure levels. The difference is how the data fields are sampled on the GPU to obtain a value at a particular position in longitude–latitude-pressure space – an operation required by all visualization algorithms. In the horizontal, data fields on a regular longitude–latitude grid are supported.

To use the data on the GPU, a single forecast variable of a single member is stored in a 3-D texture (i.e. a 3-D data array) in GPU memory. We assume that these data fields fit into GPU memory. Longitude–latitude axes, as well as pressure levels for PL grids, are stored in an additional 1-D texture. For ML grids, the corresponding 2-D p_{sfc} field and the coefficients a_k and b_k are stored. This allows to compute the pressure coordinate of a grid point on-the-fly, without the need to use additional graphics memory for a 3-D texture with pressure values.

Horizontal 2-D sections on a pressure surface p are rendered by placing the vertices of a grid of triangles horizontally at the positions of the data grid points and vertically at p (Fig. 11a). Data sampling only needs to be done when p is changed. Executed in parallel for each vertex, a binary search in the vertex shader yields the model levels (or pressure levels) k and $k+1$ enclosing p in the corresponding grid column. Following the ECMWF FULLPOS interpolation routines (Yessad, 2014), interpolation between these two levels is done linearly in $\ln(p)$. The results are cached in a 2-D texture. Filled contours are rendered by assigning colour to each fragment within a triangle in the fragment shader, using the horizontally hardware-interpolated scalar value. To obtain a colour, colour palettes (cf. Sect. 3.2) are stored as 1-D transfer functions in 1-D textures. These textures are used as lookup tables (LUTs), mapping a scalar value to a colour. Line contours are generated by a marching squares (e.g. Hansen and Johnson, 2005, Chap. 1) implementation in a geometry shader. Each grid cell of the cached 2-D cross-section texture is examined in parallel and, if applicable, a line segment is drawn. Graticule, coast and border lines are overlain on each horizontal section to im-

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

◄◄ | ►►

◄ | ►

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

prove spatial perception (cf. Fig. 3b). Wind barbs are also generated in a geometry shader. It takes the horizontal wind field's u and v components as input and generates the geometry of the barbs, again exploiting GPU parallelism.

Vertical sections are rendered with a similar grid of triangles. A triangle vertex is drawn for each vertical (model or pressure) level and each of a number of intermediate horizontal points along a line connecting the waypoints the user has specified (Fig. 9b). The distance between the intermediate points can be specified. A vertex shader computes the vertical position of each vertex and places it accordingly. This operation is a simple lookup for PL data and involves interpolation of p_{sfc} and computation of the model level pressure for ML grids. Scalar values are interpolated horizontally, also in the vertex shader, on the level on which the vertex is placed. They are also cached in a 2-D texture that is updated if a waypoint is moved. Filled and line contours are generated equivalently to those in the horizontal sections.

3-D isosurfaces are rendered with front-to-back raycasting (Krüger and Westermann, 2003; Engel et al., 2006) implemented in the fragment shader. For each fragment (pixel) of the output image, a ray is cast through the data volume, sampling it at regular intervals and thus finding isosurface crossings. For this type of visualization algorithm, sampling the scalar volume is more expensive, as we need to interpolate in all three spatial dimensions to an arbitrary position in longitude–latitude-pressure space. For PL data, the grid is rectilinear (Fig. 11b) and can be sampled using texture mapping (e.g. Bailey, 2009), thus benefiting from the fast trilinear hardware interpolation provided by modern GPUs. By mapping the longitude-latitude-pressure coordinates of the sampling position to texture coordinates $(t_{\text{lon}}, t_{\text{lat}}, t_p)$ on the unit cube, the GPU interpolates the 3-D texture at an arbitrary position. For regular grids, this mapping is a simple linear scaling. Since, however, PL grids retrieved from MARS are irregularly spaced in the vertical, we need a method to map pressure to t_p. This is realised by means of an LUT stored in an additional 1-D texture. The level indices k can be linearly scaled to $t_{p,k} \in (0 \ldots 1)$. Since we know the pressure values p_k at the levels k, we can compute a continuous $\widetilde{k}$ for intermediate p by linearly interpolating in $\ln(p)$ (Fig. 11b). $\widetilde{k}$

can subsequently be scaled to t_p. These mappings from p to t_p are precomputed for a number, say 2048, of pressure values and stored in the LUT that can be accessed in the shader.

ML grids are not rectilinear and thus sampling becomes more complicated. As illustrated in Fig. 11b, the continuous level index $\tilde{k}$ in general is not the same for adjacent grid columns. In the worst case, a given p is located between different model levels in its four surrounding grid columns. Trilinear hardware interpolation requires $\tilde{k}$ to be the same in all surrounding grid columns, it hence cannot be used. Consequently, we need to split the interpolation into four vertical interpolations in the grid columns and a subsequent bilinear horizontal interpolation. A naïve approach is to use the binary search used for the horizontal sections for the vertical interpolations, however, our experiments showed that rendering times can be reduced by a factor of about two when again making use of an LUT approach for hardware interpolation. However, the horizontal interpolation needs to be implemented in software. ML sampling is hence over four times more expensive than PL sampling.

To use hardware interpolation for ML in the vertical, we need to extend the LUT approach. First, the horizontal texture coordinates t_{lon} and t_{lat} are set to the horizontal position of the grid columns. Since the model level pressure varies with p_{sfc}, we in principle need to precompute one LUT for every p_{sfc} value that occurs in the forecast field. We instead make use of a 2-D LUT, containing LUTs for discrete values of p_{sfc} reflecting the expected range of p_{sfc} in the data. Using bilinear hardware interpolation, this LUT is used to interpolate in both p_{sfc} and $\ln(p)$ to obtain a mapping from $\ln(p)$ to t_p. The additional memory requirement is reasonable: For an LUT using 2048 entries in the vertical and 600 entries for p_{sfc} between 1050 and 450 hPa, approximately 9 MB of GPU memory are required in float precision (i.e. 4 bytes/value). The table can be shared among variables on the same grid.

The traversal of the data volume is accelerated with an empty-space skipping strategy (Krüger and Westermann, 2003). The longitude-latitude-pressure space covered by a data field is divided uniformly into a regular grid of $N_i \times N_j \times N_k$ cells. For each

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

cell, minimum and maximum data values are computed. In the shader, the information is used to skip cells in which an isosurface cannot possibly be located. Due to the different horizontal and vertical scales, care has to be taken when choosing the step size for traversing non-empty cells. Depending on the factor that is used to scale $\ln(p)$ to a z coordinate in visualization space, the vertical distance between two grid points often is considerably smaller than the horizontal distance. The step size needs to be chosen small enough to ensure that no grid point is skipped during traversal.

Once an isosurface crossing has been identified, the isosurface normal (equivalent to the gradient of the scalar field at the crossing position) is computed via central differences. The pixel colour is subsequently determined using the commonly used Blinn–Phong lighting model (e.g Engel et al., 2006). Colour can be predefined or obtained from a transfer function. Also, a second scalar field can be mapped to the isosurface to colour, for example, a wind speed isosurface by temperature.

Table 1 lists typical rendering times for images shown in this article. Note that the performance of the raycaster depends on the visualized data as well as on camera viewpoint. In particular the effectiveness of the empty-space skipping strategy for a selected isovalue depends strongly on the spatial distribution of the data values. During user interaction, the step size used by the raycaster to sample the data fields can be reduced (cf. Table 1). While this temporarily reduces image quality, rendering time is also reduced. We note that as for the data pipeline, comprehensive optimisations of the algorithms were outside the scope of our work. In particular with respect to the raycaster, further optimisations are possible, for example, by integrating an adaptive step size strategy.

4.4 Computation of normal curves

Normal curve computation is implemented in a compute shader. Figure 12 illustrates the proposed normal curve algorithm. To generate a set of seed points, rays aligned with the three world space axes (longitude, latitude, pressure) are cast through the data volume. The rays are started at regularly spaced points (grey arrows). To avoid the reg-

ular pattern of these initial start points being reflected by the normal curves, we disturb the ray positions by a random factor (black arrows). The intersection points of the rays with the selected outer isosurface are then used as initial seed points for the normal curves (green dots). In particular in regions of high curvature, multiple rays can hit the isosurface at close-by points on the surface. To prevent normal curves to be started close together, a regular volume with a grid size of the average initial ray distance is placed over the scene (yellow grid). Only one seed is allowed per grid cell. Hence, if a seed point falls into a cell already occupied, it is discarded (illustrated in the orange grid cell). The normal curves are integrated in parallel in the direction of the scalar field's gradient, using a 4th-order Runge–Kutta scheme. The gradient is computed with the same method used for isosurface shading. If present, the integration can be stopped at an inner opaque isosurface (illustrated by the red isosurface in Fig. 12).

5 Impact of (not) regridding on ensemble statistical quantities

We compute statistical quantities per grid point. Probabilities, for example, are computed by evaluating for every member and for each grid point a given probability criterion (for instance, wind speed exceeding a given threshold). The evaluation of the criterion yields for every member a binary volume, with the bits set when the criterion is fulfilled. Probabilities are computed by counting the number of members with a set bit for each grid point. Other statistical measures are computed similarly for each grid point over the ensemble dimension.

For 2-D grids, this is common procedure (Wilks, 2011) and also for 3-D grids not an issue as long as a given grid point is located at the same spatial position in all members. However, due to surface pressure varying between ensemble members, this is not the case for data on ML grids. Hence, depending on the vertical gradient of the forecast variable from which a statistical quantity is computed, an error is introduced. One approach to this issue is to vertically regrid all ensemble members to a common grid, for example, the one defined by the mean surface pressure (as done in the exam-

ple pipeline in Fig. 10). This, however, introduces an additional interpolation step and demands computational resources.

In this section, we investigate the visual and quantitative differences between statistical quantities computed from the original ML grids and those computed from data fields regridded to a common grid. The differences are compared to an additional error that is introduced by linearly interpolating the statistical quantities. At ECMWF, maps of statistical quantities on pressure levels are computed from the individual member's forecast data on these pressure levels. This implies that a forecast meteorological variable is first interpolated to the target vertical position for each member (using linear interpolation in p or $\ln(p)$, cf. Yessad, 2014), followed by the computation of the statistical quantity. If, on the contrary, we first compute the statistical quantity on the 3-D model grid and then linearly interpolate to the target vertical position, an error is introduced due to the non-linear nature of most statistical measures. The same problem arises in the horizontal dimensions.

In the following, we analyse regridding and interpolation error for the forecast data we had available from TNF. We present results from the forecast initialised at 00:00 UTC on 15 October 2012 and valid at 114 h lead time at 18:00 UTC on 19 October 2012. This case is representative for the dataset, results for other timesteps of the TNF dataset are similar.

5.1 Variation in grid point pressure

First, we estimate typical vertical grid point displacements that can be observed between ensemble members. Figure 13a shows the SD of p_{sfc} for the example case. It reaches values of 8 to 10 hPa in the uncertain regions of the forecast. This particularly applies to the low pressure systems over the Atlantic and the northern British Isles. Figure 13b shows a vertical cross-section of the maximum pressure difference between any two members per grid point in these two areas. Close to the surface, the difference reaches 40 hPa, corresponding (at low altitudes) to an elevation offset of about 400 m.

In most other regions, however, differences are smaller. Also, as expected from the model grid topology, differences vanish in upper atmospheric levels.

5.2 Difference due to vertical regridding

Vertical regridding is implemented as a data source that can be integrated into the Met.3D ensemble processing pipeline (cf. Fig. 10). The user can toggle between visualizations from original and from regridded data fields, and, if required, permanently enable regridding. If statistical quantities are computed from the original member grids, the resulting field is interpreted on a grid defined by the mean surface pressure.

On our test hardware (cf. Sect. 3), the cost of single-threaded CPU regridding on average is about 60 ms per member and variable for the TNF ENS forecast at 1° resolution (256 742 grid points per 3-D field) and about 1 s at 0.25° resolution (4 997 262 grid points). Even though multiple ensemble members can be processed in parallel on a multicore machine and the regridding process could be further sped up using the GPU, there is a delay in particular for high-resolution datasets and visualizations using multiple variables.

We have visually inspected a number of 2-D and 3-D renderings of statistical quantities of several meteorological variables. As expected, the largest visual differences appear close to the surface. They become most manifest in horizontal sections, which are most sensitive to vertical variations in a 3-D data field. Figure 14 shows two typical low-altitude examples, the probability of horizontal wind speed exceeding $20\,\mathrm{m\,s^{-1}}$, $p(|\boldsymbol{v}| > 20\,\mathrm{m\,s^{-1}})$, and the SD of relative humidity, $\sigma(\mathrm{RH})$. From our inspection we find that differences tend to be larger for variables that depend on moisture and variables derived thereof, however, we could not find any examples in which visualized structures were significantly altered. For example, while there is some visible difference in $\sigma(\mathrm{RH})$ along Rafael's warm front, the structure itself is not significantly altered.

Visual differences strongly depend on the employed colour palette and visualized data range. Depending on the range of values covered by a single colour, small changes might simply not be reflected in the visualization. To ensure that differences in

general are small, we have performed a statistical analysis of the entire TNF dataset. Figure 15 shows results for three statistical quantities computed from the wind field of the example case: mean $\mu(|\boldsymbol{v}|)$, SD $\sigma(|\boldsymbol{v}|)$, and $p(|\boldsymbol{v}| > 20\,\mathrm{m\,s^{-1}})$. The scatter plots show that for all three quantities the largest differences appear at lower altitudes (higher model level indices). Also, differences mostly are small compared to absolute values of the quantities. For example, at only few grid points the difference in $\sigma(|\boldsymbol{v}|)$ and $p(|\boldsymbol{v}| > 20\,\mathrm{m\,s^{-1}})$ exceeds $1\,\mathrm{m\,s^{-1}}$ and $10\,\%$, respectively. The range of differences observed in Fig. 14 is well reflected in the histogram.

Larger differences appear for statistical quantities computed from moist variables (Fig. 16). Again, the histogram for $\sigma(\mathrm{RH})$ confirms the range of differences shown in Fig. 14 (Fig. 16d). For probabilities of potential vorticity and cloud cover, differences of up to $30\,\%$ can occur (Fig. 16e and f). However, for most grid points, differences are smaller.

Figure 17 shows a histogram of $\sigma(p_\mathrm{sfc})$ of the example case, overlain with the bin-averaged difference in $\sigma(|\boldsymbol{v}|)$. As can be expected, larger differences on average occur in regions with high $\sigma(p_\mathrm{sfc})$. However, even for large $\sigma(p_\mathrm{sfc})$, most differences are small (not shown). We hence cannot state that large $\sigma(p_\mathrm{sfc})$ in general accounts for large differences.

5.3 Error due to vertical interpolation of statistical quantities

The error introduced by vertical linear interpolation of a statistical quantity depends on the quantity. Consider the example given in Table 2. Due to the linear nature of the ensemble mean, there is no difference whether we first compute the mean at the grid points and then interpolate to the sample location or vice versa. For non-linear quantities including SD and probability, the results are different.

Figure 18 shows distributions of the interpolation errors for $\sigma(|\boldsymbol{v}|)$ and $p(|\boldsymbol{v}| > 20\,\mathrm{m\,s^{-1}})$. Note that in contrast to the differences caused by regridding, the largest errors due to interpolation occur in upper atmospheric levels, where the vertical distance

between model levels becomes larger. Between the surface and approximately model level 10 (approximately 100 hPa), the order of magnitude of the interpolation errors is comparable to that of the differences due to regridding. At middle atmospheric levels, both errors are at a minimum, as shown by the vertical profile of horizontally aver-
aged differences. At the upper boundary of the model atmosphere, interpolation errors become significantly larger, These regions, however, are not relevant for the forecast cases we are interested in.

5.4 Discussion

The examples show that the errors introduced by computing the statistical quantities from the original member grids are of comparable magnitude to the errors introduced by vertically interpolating the computed quantities. For most grid points, both are negligible and result in only little difference in the visualization. However, for some variables and cases (in particular moist variables), differences can be of the same order of magnitude as the statistical quantity itself.

We conclude that for general exploration of the forecast data it is sufficient for the user to use the "fast" option and visualize quantities computed from the original member grids. However, if the result is crucial for an important decision, our advice is to switch to regridded quantities and accept the additional compute time. The "best" results and those most comparable to products obtained from ECMWF can be achieved by first interpolating each member to the desired vertical pressure and then computing the statistical quantities. In this case, neither regridding nor vertical interpolation of the quantity corrupts the result. In Met.3D, this is possible for horizontal sections.

6 Code availability

To facilitate ease of deployment and of future research and developments, we make the source code of Met.3D available as open-source under the GNU General Public

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

|◀ | ▶|

◀ | ▶

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

7 Conclusions

We have presented Met.3D, a new open-source tool that provides interactive 3-D visualization techniques for numerical ensemble weather prediction data in a way suitable for weather forecasting. The development of Met.3D has been motivated by the application of forecasting during aircraft-based atmospheric field campaigns, in particular, by the requirements of the T-NAWDEX-Falcon 2012 campaign. However, we see the tool applicable to a wider range of applications, including the analysis of ensemble simulation output in atmospheric research and the usage of Met.3D to support teaching in meteorology classes.

Our work is concerned with meaningful 3-D depiction and ensemble visualization, two challenging topics of weather forecast visualization. We have addressed a number of challenges that have been discussed in the literature, including prevention of a decoupling between commonly used 2-D and new 3-D visualization methods, spatial perception in 3-D scenes, suitable uncertainty visualization techniques, and system performance. Interactivity is key to our approach. It is facilitated by exploiting the computational power provided by modern graphics processing units and by means of a flexible, modular system architecture. We have built a bridge from proven 2-D visualization methods commonly used in meteorology to 3-D visualization. 2-D products are rendered in a 3-D context, a product's position can be changed interactively. When 3-D elements are visualized, spatial perception is improved by displaying shadows on the Earth's surface, enabling the user to judge the horizontal position and relative elevation of an element. Quantitative height information can be obtained by means of interactive vertical axes. We have proposed normal curves, a novel visualization technique to

3-D visualization of ensemble weather forecasts – Part 1: Met.3D

M. Rautenhaus et al.

reveal the structure inside a transparent 3-D isosurface of a scalar field. With normal curves, the locations and magnitudes of local extrema in the visualized data can be identified at a glance. To visually provide information on forecast uncertainty, Met.3D implements support for ensemble forecasts. The tool is designed to allow integration of both feature-based and location-based ensemble visualization techniques. In the presented version, forecast products can be animated over the ensemble dimension, and statistical quantities can be derived and visualized on demand. Concerning the computation of statistical quantities from forecast data on hybrid sigma-pressure grids, we have shown that ignoring the variation in grid point pressure between the ensemble members has little impact on the visualization.

The article at hand is the first of a two-part study. We have focussed on Met.3D's functionality, system architecture and visualization algorithms. In R15P2, the second part, we focus on the specific forecast requirements of T-NAWDEX-Falcon and use Met.3D to predict warm conveyor belt situations. Ensemble particle trajectories are employed to predict a probability of warm conveyor belt occurrence. In particular, a case study, revisiting a forecast case from T-NAWDEX-Falcon, demonstrates the practical application of Met.3D and highlights the potential of the software to improve the weather forecasting process.

Future work needs to include careful evaluation of the presented visualization techniques to study their impact on tasks performed by meteorologists and atmospheric researchers in their daily work. We discuss our point of view on the added value of interactive 3-D ensemble visualization for forecasting after the presentation of the case study in the conclusions of R15P2. For example, in our experience, the provided interactivity for 2-D sections and the ability to add features as 3-D elements helps to much faster build a mental model of the atmosphere. This, of course, reflects our personal perception. We plan to evaluate the issue with a user study in the near future.

We will actively use Met.3D during upcoming field campaigns, including a future NAWDEX campaign scheduled for 2016. We also see much potential for further research in meteorological visualization: With respect to 3-D visualization, further im-

3-D visualization of ensemble weather forecasts – Part 1: Met.3D

M. Rautenhaus et al.

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

provement of spatial perception is very important. In the Met.3D version presented here, shadows are only rendered on the Earth's surface. Global illumination techniques (e.g. Jönsson et al., 2014) that, for example, allow 3-D elements to mutually cast shadows on each other, may further improve the user's judgement of spatial relationships. Also, the impact of different projections on perceived spatial distance needs to be studied. Met.3D currently is restricted to a cylindrical map projection in the horizontal. Additional challenges include the efficient rendering from further native model grid topologies and real-time placement of text labels to convey quantitative information. The latter applies in particular to 2-D and 3-D contour lines and surfaces. Due to the employed GPU implementation of the 2-D marching squares contouring algorithm, continuous line geometry is not easily available. Hence, it is difficult to compute positions for labels.

With respect to ensemble and uncertainty visualization, open questions are abundant, as reflected by the literature surveyed in Sect. 2. In R15P2, we introduce a feature-based approach for WCBs. Further approaches, both feature-based and location-based, can be implemented in Met.3D to study their feasibility and applicability in meteorology.

With the development of Met.3D, we have demonstrated how we envision 3-D and ensemble techniques to become a part of standard meteorological visualization. The tool provides a solid software infrastructure that opens the door to investigate the above listed and other research questions, thus enabling the further advancement of meteorological visualization.

The Supplement related to this article is available online at doi:10.5194/gmdd-8-2101-2015-supplement.

References

Bailey, M.: Using GPU shaders for visualization, IEEE Comput. Graph., 29, 96–100, 2009. 2118, 2123

Bailey, M.: Using GPU shaders for visualization, Part 2, IEEE Comput. Graph., 31, 67–73, 2011. 2118

Bailey, M.: Using GPU shaders for visualization, Part 3, IEEE Comput. Graph., 33, 5–11, 2013. 2118

Brown, R.: Animated visual vibrations as an uncertainty visualisation technique, in: Proceedings of the 2nd International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia, GRAPHITE '04, ACM, New York, NY, USA, 84–89, 2004. 2111

Browning, K. A. and Roberts, N. M.: Structure of a frontal cyclone, Q. J. Roy. Meteor. Soc., 120, 1535–1557, 1994. 2105

Buizza, R., Bidlot, J.-R., Wedi, N., Fuentes, M., Hamrud, M., Holt, G., Palmer, T., and Vitart, F.: The ECMWF Variable Resolution Ensemble Prediction System VAREPS, ECMWF Newsletter, 108, 14–20, 2006. 2112

Clyne, J., Mininni, P., Norton, A., and Rast, M.: Interactive desktop analysis of high resolution simulations: application to turbulent plume dynamics and current sheet formation, New J. Phys., 9, 301, doi:10.1088/1367-2630/9/8/301, 2007. 2109

Cox, J., House, D., and Lindell, M.: Visualizing uncertainty in predicted hurricane tracks, Int. J. Uncertainty Quant., 3, 143–156, 2013. 2110

Djurcilov, S., Kim, K., Lermusiaux, P., and Pang, A.: Visualizing scalar volumetric data with uncertainty, Comput. Graph., 26, 239–248, 2002. 2110

Dyer, J. and Amburn, P.: Desktop visualization of meteorological data using paraview, Kitware Source, 14, 7–10, 2010. 2109

Engel, K., Hadwiger, M., Kniss, J., Rezk-Salama, C., and Weiskopf, D.: Real-Time Volume Graphics, 1st Edn., AK Peters, Wellesley, Mass, 2006. 2118, 2123, 2125

Gneiting, T. and Raftery, A. E.: Weather forecasting with ensemble methods, Science, 310, 248–249, 2005. 2103

5 Grigoryan, G. and Rheingans, P.: Point-based probabilistic surfaces to show surface uncertainty, IEEE T. Vis. Comput. Gr., 10, 564–573, 2004. 2111

Hansen, C. D. and Johnson, C.: The Visualization Handbook, 1st Edn., Academic Press, Burlington, MA, 2005. 2122

Heizenrieder, D. and Haucke, S.: Das meteorologische Visualisierungs- und Produktionssystem
10 NinJo, promet, 35, 57–69, 2009. 2108

Henderson, A., Ahrens, J., and Law, C.: The ParaView Guide, Kitware, Clifton Park, NY, 2004. 2109

Hewson, T. D. and Titley, H. A.: Objective identification, typing and tracking of the complete life-cycles of cyclonic features at high spatial resolution, Met. Apps, 17, 355–381, 2010. 2110

15 Hibbard, B.: The top five problems that motivated my work, IEEE Comput. Graph., 24, 9–13, 2004. 2118

Hibbard, B. and Santek, D.: The VIS-5D system for easy interactive visualization, in: Proceedings of the 1st Conference on Visualization '90, VIS '90, San Francisco, CA, USA, 23–26 October 1990, IEEE Computer Society Press, Los Alamitos, CA, USA, 28–35, 1990. 2108

20 Hoffman, R. R. and Coffey, J. W.: Weather forecasting and the principles of complex cognitive systems, in: Proceedings of the Human Factors and Ergonomics Society Annual Meeting, Vol. 48, New Orleans, Louisiana, 20–24 September 2004, SAGE Publications, 315–319, doi:10.1177/154193120404800309, 2004. 2103

Hoffman, R. R., Detweiler, M., Conway, J. A., and Lipton, K.: Some considerations in using color
25 in meteorological displays, Weather Forecast., 8, 505–518, 1993. 2115

Höllt, T., Magdy, A., Zhan, P., Chen, G., Gopalakrishnan, G., Hoteit, I., Hansen, C. D., and Hadwiger, M.: Ovis: a framework for visual analysis of ocean forecast ensembles, IEEE T. Vis. Comput. Gr., 20, 1114–1126, doi:10.1109/TVCG.2014.2307892, 2014. 2111

Jönsson, D., Sundén, E., Ynnerman, A., and Ropinski, T.: A survey of volumetric illumination
30 techniques for interactive volume rendering, Comput. Graphics Forum, 33, 27–51, 2014. 2133

Johnson, C. R. and Sanderson, A. R.: A next step: visualizing errors and uncertainty, IEEE Comput. Graph., 23, 6–10, 2003. 2110

Title Page

Abstract   Introduction

Conclusions   References

Tables   Figures

Back   Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Kaufmann, S., Voigt, C., Jeßberger, P., Jurkat, T., Schlager, H., Schwarzenboeck, A., Klinge-biel, M., and Thornberry, T.: In situ measurements of ice saturation in young contrails, Geo-phys. Res. Lett., 41, 702–709, doi:10.1002/2013gl058276, 2014. 2116

Koppert, H. J., Schröder, F., Hergenröther, E., Lux, M., and Trembilski, A.: 3-D visualisation in daily operation at the DWD, in: Proceedings of the 6th ECMWF Workshop on Meteorological Operational Systems, Reading, England, 17–21 November 1997, 119–142, 1998. 2103, 2107, 2108

Krüger, J. and Westermann, R.: Acceleration techniques for GPU-based volume render-ing, in: Proceedings of the 14th IEEE Visualization 2003 (VIS'03), Seattle, Washing-ton, 22–24 October 2003, 287–292, IEEE Computer Society, Washington, DC, USA, doi:10.1109/VIS.2003.10001, 2003. 2123, 2124

Lalaurette, F.: Early detection of abnormal weather conditions using a probabilistic extreme forecast index, Q. J. Roy. Meteor. Soc., 129, 3037–3057, 2003. 2110

Lamy-Thépaut, S., Sahin, C., and Raoult, B.: ecCharts service, ECMWF Newsletter, 134, 7–9, 2013. 2110

Leutbecher, M. and Palmer, T.: Ensemble forecasting, J. Comput. Phys., 227, 3515–3539, 2008. 2103, 2112

Lindemann, F. and Ropinski, T.: About the influence of illumination models on image compre-hension in direct volume rendering, IEEE T. Vis. Comput. Gr., 17, 1922–1931, 2011. 2109

Lundstrom, C., Ljung, P., Persson, A., and Ynnerman, A.: Uncertainty visualization in medical volume rendering using probabilistic animation, IEEE T. Vis. Comput. Gr., 13, 1648–1655, 2007. 2110

Lux, M., and Frühauf, T.: A visualization system for operational meteorological use, in: Pro-ceedings of the Sixth International Conference in Central Europe on Computer Graphics and Visualization (WSCG'98), Plzen, Czech Republic, 9–13 February 1998, 525–534, 1998. 2108

McCaslin, P. T., McDonald, P. A., and Szoke, E. J.: 3-D visualization development at NOAA forecast systems laboratory, Comp. Graph., 34, 41–44, 2000. 2103, 2108

Middleton, D., Scheitlin, T., and Wilhelmson, B.: Visualization in weather and climatic research, in: The Visualization Handbook, Chap. 44, edited by: Hansen, C. D. and Johnson, C., Aca-demic Press, 845–871, 2005. 2109

Miller, M., Buizza, R., Haseler, J., Hortal, M., Janssen, P., and Untch, A.: Increased resolution in the ECMWF deterministic and ensemble prediction systems, ECMWF Newsletter, 124, 10–16, 2010. 2112

Murray, D. and McWhirter, J.: Evolving IDV – creating better tools for the community, in: 23th Conference on International Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology, 15–18 January 2007, San Antonio, TX, American Meteorological Society, 3B.5, 2007. 2109

Murray, D., McWhirter, J., Ho, Y., and Whittaker, T. M.: IDV at 5: new features and future, in: 25th Conference on International Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology, 10–15 January 2009, Phoenix, AZ, American Meteorological Society, 7B.5, 2009. 2109

Norton, A. and Clyne, J.: The VAPOR visualization application, in: High Performance Visualization, Chap. 20, edited by: Bethel, E. W., Childs, H., and Hansen, C., CRC Press, Boca Raton, FL, 415–428, 2012. 2109

Obermaier, H. and Joy, K. I.: Future challenges for ensemble visualization, IEEE Comput. Graph., 34, 8–11, 2014. 2103, 2110

Pang, A. T., Wittenbrink, C. M., and Lodha, S. K.: Approaches to uncertainty visualization, Visual Comput., 13, 370–390, 1997. 2110

Pfaffelmoser, T. and Westermann, R.: Visualization of global correlation structures in uncertain 2-D scalar fields, Comput. Graph. Forum, 31, 1025–1034, 2012. 2111

Pfaffelmoser, T., Reitinger, M., and Westermann, R.: Visualizing the positional and geometrical variability of isosurfaces in uncertain scalar fields, Comput. Graph. Forum, 30, 951–960, 2011. 2111, 2116

Pöthkow, K. and Hege, H. C.: Positional uncertainty of isocontours: condition analysis and probabilistic measures, IEEE T. Vis. Comput. Gr., 17, 1393–1406, doi:10.1109/tvcg.2010.247, 2011. 2111

Pöthkow, K., Weber, B., and Hege, H.-C.: Probabilistic marching cubes, Comput. Graph. Forum, 30, 931–940, 2011. 2111

Potter, K., Wilson, A., Bremer, P. T., Williams, D., Doutriaux, C., Pascucci, V., and Johnson, C. R.: Ensemble-Vis: a framework for the statistical visualization of ensemble data, in: Int. Conference on Data Mining Workshops, Miami, FL, 6 December 2009, IEEE Computer Society, Los Alamitos, CA, USA, 233–240, doi:10.1109/ICDMW.2009.55, 2009. 2103, 2111

3-D visualization of ensemble weather forecasts – Part 1: Met.3D

M. Rautenhaus et al.

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

◄◄ | ►►

◄ | ►

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Rautenhaus, M., Bauer, G., and Dörnbrack, A.: A web service based tool to plan atmospheric research flights, Geosci. Model Dev., 5, 55–71, doi:10.5194/gmd-5-55-2012, 2012. 2108, 2114, 2120

Rautenhaus, M., Grams, C. M., Schäfler, A., and Westermann, R.: GPU based interactive 3-D visualization of ECMWF ensemble forecasts, ECMWF Newsletter, 138, 34–38, 2014. 2105

Rautenhaus, M., Grams, C. M., Schäfler, A., and Westermann, R.: 3-D visualization of ensemble weather forecasts – Part 2: Forecasting warm conveyor belt situations for aircraft-based field campaigns, Geosci. Model Dev. Discuss., 8, 2161–2212, doi:10.5194/gmdd-8-2161-2015, 2015 2107

Rhodes, P. J., Laramee, R. S., Bergeron, R. D., and Sparr, T. M.: Uncertainty visualization methods in isosurface rendering, in: Proceedings Eurographics 2003, Granada, Spain, 1–5 September 2003, 83–88, 2003. 2110

Russell, I., Siemen, S., Ii, F., Kertész, S., Lamy-Thépaut, S., and Karhila, V.: Metview 4 – ECMWF's latest generation meteorological workstation, ECMWF Newsletter, 126, 23–27, 2010. 2108

Sanyal, J., Zhang, S., Dyer, J., Mercer, A., Amburn, P., and Moorhead, R.: Noodles: a tool for visualization of numerical weather model ensemble uncertainty, IEEE T. Vis. Comput. Gr., 16, 1421–1430, 2010. 2103, 2111

Schäfler, A., Boettcher, M., Grams, C. M., Rautenhaus, M., Sodemann, H., and Wernli, H.: Planning aircraft measurements within a warm conveyor belt, Weather, 69, 161–166, 2014. 2105

Schumann, U.: On conditions for contrail formation from aircraft exhausts, Meteorol. Z., 5, 4–23, 1996. 2116

Stauffer, R., Mayr, G. J., Dabernig, M., and Zeileis, A.: Somewhere over the Rainbow: How to Make Effective Use of Colors in Meteorological Visualizations, Tech. rep., Faculty of Economics and Statistics, University of Innsbruck, 2013. 2115

Szoke, E. J., Grote, U. H., McCaslin, P. T., and McDonald, P. A.: D3D update: is it being used?, in: Proceedings of the 19th IIPS Conference, Long Beach, CA, 9–13 February 2003, P1.10, 2003. 2103, 2108, 2109, 2118

Trafton, J. G. and Hoffman, R. R.: Computer-aided visualization in meteorology, in: Expertise Out of Context: Proceedings of the Sixth International Conference on Naturalistic Decision Making, edited by: Hoffman, R. R., Chap. 15, Psychology Press, 337–357, 2007. 2103, 2113

3-D visualization of ensemble weather forecasts – Part 1: Met.3D

M. Rautenhaus et al.

Title Page

Abstract	Introduction
Conclusions	References
Tables	Figures

|◄ | ►|

◄ | ►

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Treinish, L. A.: Task-specific visualization design: a case study in operational weather fore-
casting, in: Proceedings Visualization '98, Research Triangle Park, NC, 24 October 1998,
405–409, doi:10.1109/VISUAL.1998.745330, 1998. 2108

Treinish, L. A. and Rothfusz, L. P.: Three-dimensional visualization for support of operational
forecasting at the 1996 Centennial Olympic Games, in: Proceedings of the 13th IIPS Confer-
ence, 2–7 February 1997, Long Beach, CA, 2–8, 1997. 2103, 2108

Untch, A. and Hortal, M.: A finite-element scheme for the vertical discretization of the semi-
Lagrangian version of the ECMWF forecast model, Q. J. Roy. Meteor. Soc., 130, 1505–1530,
2004. 2119

Voigt, C., Schumann, U., Jurkat, T., Schäuble, D., Schlager, H., Petzold, A., Gayet, J.-F.,
Krämer, M., Schneider, J., Borrmann, S., Schmale, J., Jessberger, P., Hamburger, T., Licht-
enstern, M., Scheibe, M., Gourbeyre, C., Meyer, J., Kübbeler, M., Frey, W., Kalesse, H.,
Butler, T., Lawrence, M. G., Holzäpfel, F., Arnold, F., Wendisch, M., Döpelheuer, A.,
Gottschaldt, K., Baumann, R., Zöger, M., Sölch, I., Rautenhaus, M., and Dörnbrack, A.:
In-situ observations of young contrails – overview and selected results from the CON-
CERT campaign, Atmos. Chem. Phys., 10, 9039–9056, doi:10.5194/acp-10-9039-2010,
2010. 2116

Wanger, L. R., Ferwerda, J. A., and Greenberg, D. P.: Perceiving spatial relationships in
computer-generated images, IEEE Comput. Graph., 12, 44–58, 1992. 2109

Weigle, C. and Banks, D. C.: A comparison of the perceptual benefits of linear perspective and
physically-based illumination for display of dense 3-D streamtubes, IEEE T. Vis. Comput. Gr.,
14, 1723–1730, 2008. 2109

Whitaker, R. T., Mirzargar, M., and Kirby, R. M.: Contour boxplots: a method for characterizing
uncertainty in feature sets from simulation ensembles, IEEE T. Vis. Comput. Gr., 19, 2713–
2722, 2013. 2110

Wilks, D. S.: Statistical Methods in the Atmospheric Sciences, 3rd Edn., Academic Press, Am-
sterdam, 2011. 2110, 2126

Wittenbrink, C. M., Pang, A. T., and Lodha, S. K.: Glyphs for visualizing uncertainty in vector
fields, IEEE T. Vis. Comput. Gr., 2, 266–279, 1996. 2110

Yalda, S., Zoppetti, G., Clark, R., and Mackin, K.: Interactive immersion learning: flying through
weather data onboard the GEOpod, B. Am. Meteorol. Soc., 93, 1811–1813, 2012. 2109

Yessad, K.: FULL-POS in the Cycle 40T1 of ARPEGE/IFS, Tech. rep., Meteo-France, 2014.
2122, 2127

Zehner, B., Watanabe, N., and Kolditz, O.: Visualization of gridded scalar data with uncertainty in geosciences, Comput. Geosci., 36, 1268–1275, 2010. 2111

Zeileis, A., Hornik, K., and Murrell, P.: Escaping RGBland: selecting colors for statistical graphics, Comput. Stat. Data An., 53, 3259–3270, 2009. 2115

3-D visualization of ensemble weather forecasts – Part 1: Met.3D

M. Rautenhaus et al.

Title Page

Abstract	Introduction
Conclusions	References
Tables	Figures

◄ | ►

◄ | ►

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

**3-D visualization of ensemble weather forecasts – Part 1: Met.3D**

M. Rautenhaus et al.

Table 1. Order of magnitude of response and rendering times of Met.3D. Timings are measured on the test hardware described in Sect. 3. Top: time required by Met.3D to display a new image after the user has advanced the forecast time. Timings are given for the horizontal section in Fig. 3b, which uses four forecast variables (reading the ensemble from disk hence involves reading $4 \times 51 \times 1$ MB at 1° resolution and $4 \times 51 \times 16$ MB at 0.25° resolution). Bottom: time required to render selected visualizations shown in this paper when the data fields are available in GPU memory. Timings are average values of continuous rendering over 30 s. A Met.3D window of 1600 by 900 pixels is used (the size used for the video in the Supplement, corresponding to a viewport of 1192 by 864 pixels). "Animated" for cross-sections refers to vertically sliding a horizontal section or moving a waypoint of a vertical section.

single member, NetCDF		ensemble mean, NetCDF		single member, cached		ensemble mean, cached	
1°	0.25°	1°	0.25°	1°	0.25°	1°	0.25°
50 ms	365 ms	2.2 s	34 s	< 10 ms	25 ms	< 10 ms	25 ms

Fig. 3b		Fig. 4a		Figs. 6a, 7a/c, 2c	
static	animated	static	animated	step size 0.1	step size 1
2.3 ms	2.8 ms	6.2 ms	6.4 ms	417, 222/248, 273 ms	107, 62/72, 83 ms

Title Page

Abstract · Introduction

Conclusions · References

Tables · Figures

◄◄ · ►►

◄ · ►

Back · Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Table 2. Example of vertically interpolating statistical quantities. Consider an ensemble of three members and corresponding scalar quantities $s_1 .. s_3$ at the two vertical levels k and $k+1$. While the mean value $\mu(s)$, interpolated to the midlevel between k and $k+1$, equals the mean of the interpolated scalar values, this is not true for the SD $\sigma(s)$ and the probability that a scalar value exceeds 1.5, $p(s > 1.5)$. The subscript i refers to "interpolated".

level	s_1	s_2	s_3	$\mu(s_i)$	$\mu_i(s)$	$\sigma(s_i)$	$\sigma_i(s)$	$p(s_i > 1.5)$	$p_i(s > 1.5)$
k	0.8	1.7	1.8	1.433		0.45		0.66	
midlevel	1.4	1.45	1.4	1.4166	1.4166	0.24	0.44	0	0.5
$k+1$	2.0	1.2	1.0	1.40		0.43		0.33	
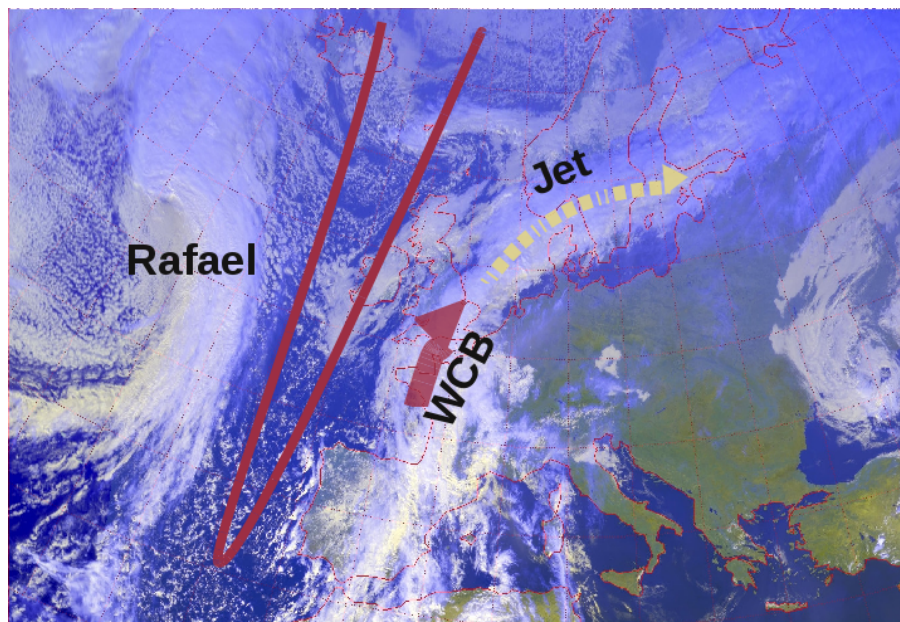
Figure 1. Real-world context for the T-NAWDEX-Falcon case used for the examples: visible Meteosat satellite image of Europe and the North Atlantic of 19 October 2012, 12:00 UTC (Meteosat operated by EUMETSAT, image processing by DLR-IPA). Important features are the narrow trough to the west of the British Isles (dark red line), the former Hurricane *Rafael* and the WCB manifest in the cloud band east of the trough.
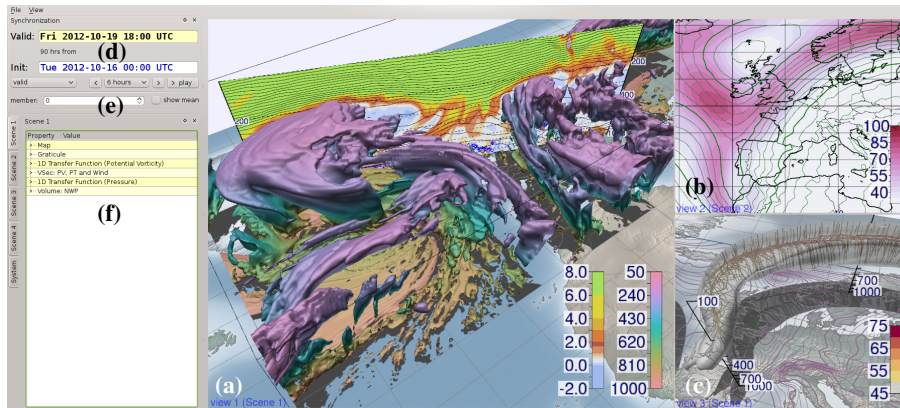
Figure 2. The main user interface of Met.3D. We apply 2-D and 3-D visualization techniques to explore ensemble weather forecasts. **(a)** Isosurfaces of cloud cover fraction of 0.5 coloured by elevation (hPa), and a vertical section of potential vorticity (PVU). **(b)** Horizontal section with contour lines of the mean geopotential height field (m) and and filled contours of its SD (m). **(c)** Normal curves applied to the wind field to visualize the jet core. The white isosurface shows $45\,\mathrm{m\,s^{-1}}$. Colour coding in $\mathrm{m\,s^{-1}}$. **(d–f)** See text for details.

Figure 3. Bridge from 2-D to 3-D visualization. **(a)** Horizontal section of geopotential height (contour lines) and horizontal wind speed (colour) at 250 hPa, as obtained from the DLR Mission Support System. ECMWF deterministic forecast from 00:00 UTC on 17 October 2012, valid at 18:00 UTC on 19 October 2012. **(b)** The same data, rendered by Met.3D and mapped into the 3-D context. The section can be interactively moved by the user. **(c)** Vertical section of horizontal wind speed (colour) and potential temperature (contour lines) in Met.3D, amended by a $50\,\mathrm{m\,s^{-1}}$ isosurface of wind speed, coloured by pressure (hPa). Note how spatial perception of the 3-D isosurface is aided by rendering shadows and labelled vertical poles (animated version of this figure in the Supplement at 00:05 min).

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

|◄ | ►|

◄ | ►

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion
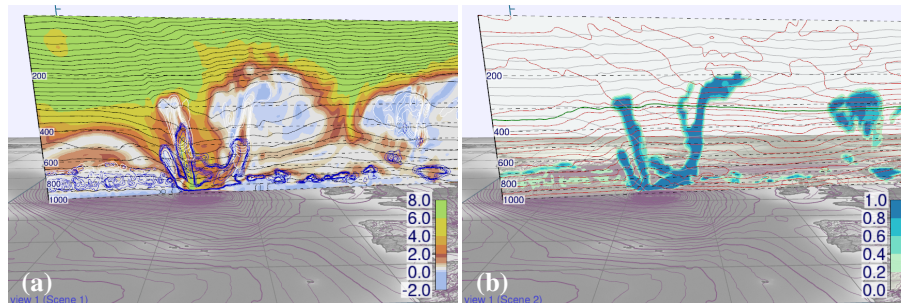
Figure 4. Vertical sections can be moved interactively in Met.3D to explore the vertical structure of the atmosphere, for example along potential flight track segments. **(a)** Potential vorticity (colour coding in PVU), **(b)** cloud cover fraction. Red colours in **(a)** mark the 2-PVU surface and thus the dynamic tropopause. Note the low tropopause along the trough. Same forecast as in Fig. 3 (animated version of this figure in the Supplement at 01:24 min).
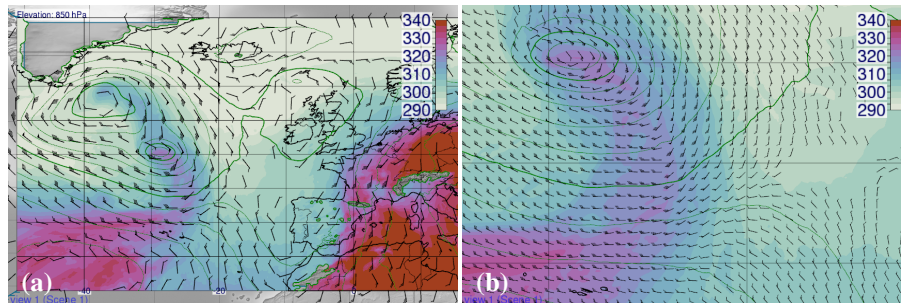
Figure 5. Met.3D automatically scales size and density of wind barbs overlain on horizontal sections. (**a** and **b**) Equivalent potential temperature (colour coded in K) at 850 hPa, overlain with contour lines of geopotential height. Same forecast as in Fig. 3 (animated version of this figure in the Supplement at 01:54 min).
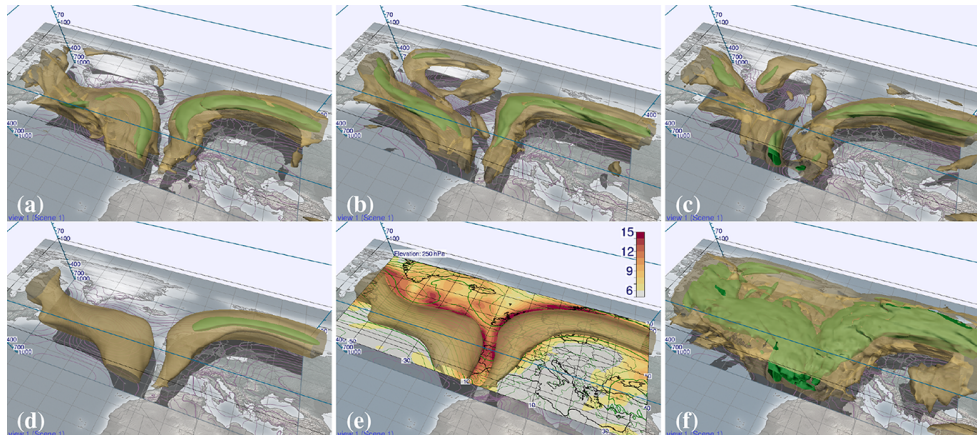
Figure 6. Navigation through the ensemble. Visualized are the $50\,\mathrm{m\,s}^{-1}$ (green opaque) and $30\,\mathrm{m\,s}^{-1}$ (yellow transparent) isosurfaces of horizontal wind speed (forecast from 00:00 UTC on 15 October valid at 18:00 UTC on 19 October 2012). **(a)** Control run, members **(b)** 27 and **(c)** 33, **(d)** ensemble mean, **(e)** ensemble mean augmented by a horizontal section of SD ($\mathrm{m\,s}^{-1}$), **(f)** ensemble maximum (animated version of this figure in the Supplement at 02:26 min).
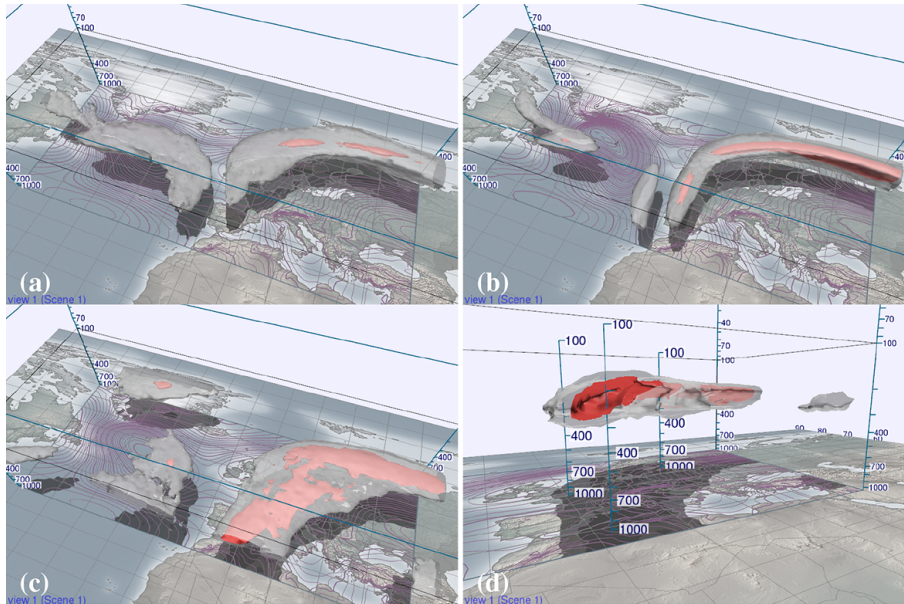
Figure 7. Probability fields computed from the ensemble, valid on 19 October 2012, 18:00 UTC. (**a** and **b**) Probability of horizontal wind speed exceeding $50\,\mathrm{m\,s^{-1}}$, as computed from the forecast initialized (**a**) at 00:00 UTC on 15 October and (**b**) at 00:00 UTC on 17 October. Shown are the 70 % (red opaque) and 10 % (white transparent) isosurfaces. Note how the ensemble converges. (**c** and **d**) Probability of contrail occurrence (Schmidt–Appleman criterion fulfilled and relative humidity greater than 80 %), as viewed from difference camera positions (80 % red opaque and 50 % white transparent) (animated version of this figure in the Supplement at 03:23 min).
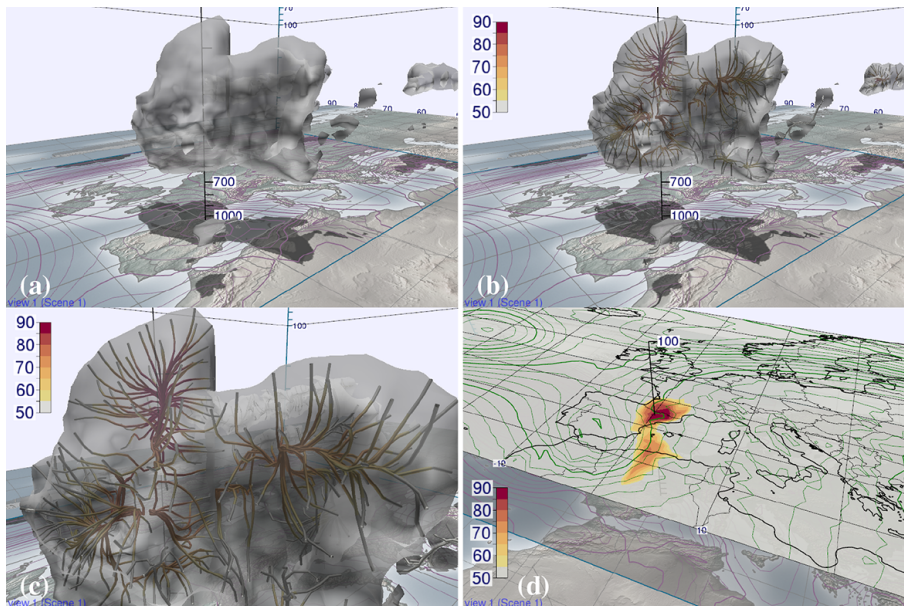
Figure 8. Normal curves help to analyse the topology of 3-D scalar fields. They reveal the distribution of data values in a subdomain enclosed by a 3-D isosurface and enable fast identification and tracking of local extrema. **(a–c)** Probability of cloud ice water content exceeding $0.01\,\mathrm{g\,kg^{-1}}$. The white transparent isosurface shows 40 % probability. Colour coding in %. **(d)** Details of the identified maximum are inspected with a horizontal section at 250 hPa. Forecast from 00:00 UTC on 17 October valid at 12:00 UTC on 20 October 2012 (animated version of this figure in the Supplement at 04:28 min).
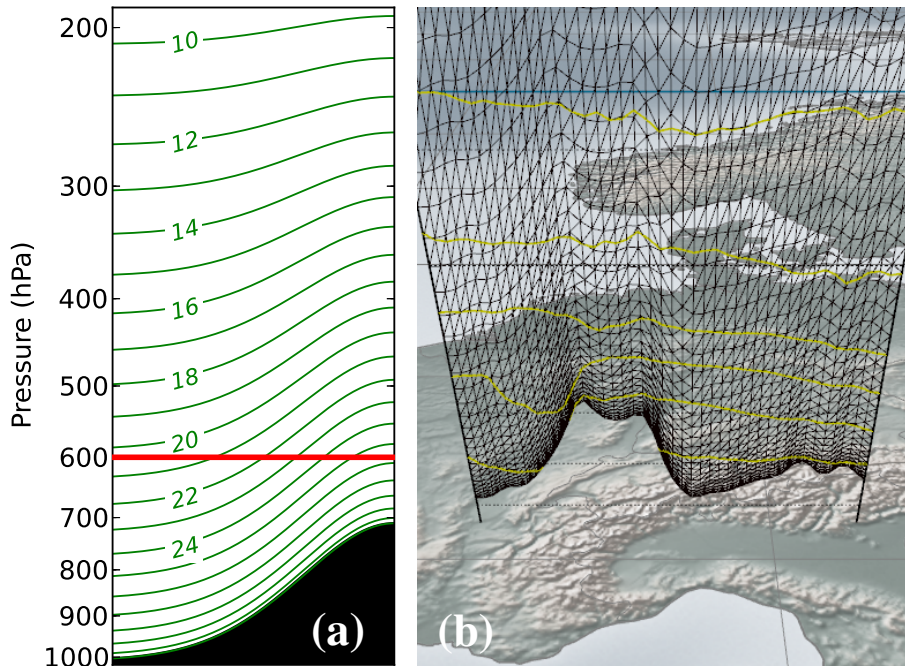
Figure 9. Hybrid sigma-pressure levels used by the ECMWF model. **(a)** The elevation of the model levels (green lines, the example shows levels from the 31 level model, level indices k in green) changes with surface pressure (black curve at the bottom). The data value for a given pressure value p can be located at different levels in the grid (the red line marks the location of $p = 600$ hPa). **(b)** Example of how the surface orography affects the vertical displacement of the grid points in a vertical section.

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

◀ | ▶

◀ | ▶

Back | Close

Full Screen / Esc

3-D visualization of ensemble weather forecasts – Part 1: Met.3D

M. Rautenhaus et al.
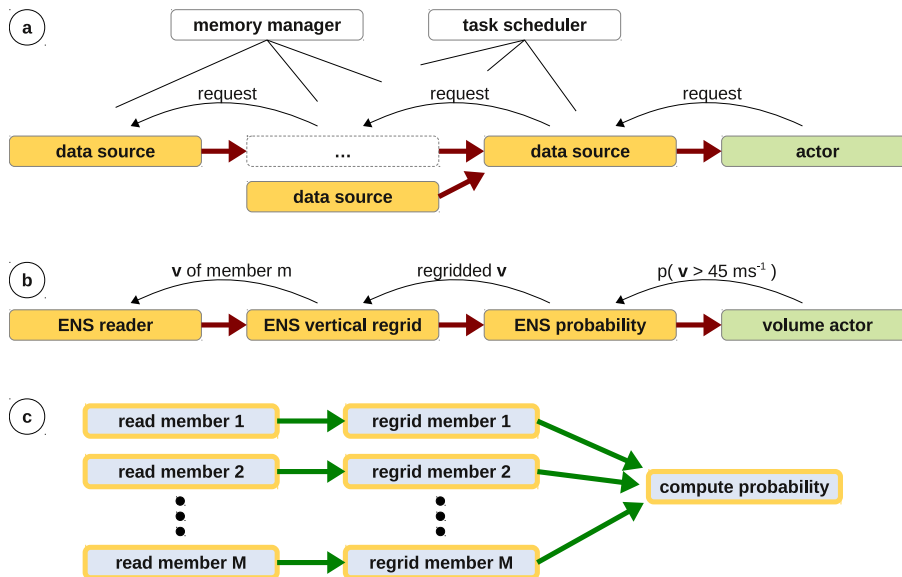
Figure 10. Pipeline concept of Met.3D: **(a)** Data sources are connected to form a pipeline, into which a visualization *actor* sends data requests. **(b)** Sample pipeline to visualize the probability of horizontal wind speed exceeding $45\,\mathrm{m\,s^{-1}}$. A request for the probability triggers further requests up the pipeline. **(c)** Task graph generated by the pipeline in **(b)**.

3-D visualization of ensemble weather forecasts – Part 1: Met.3D

M. Rautenhaus et al.
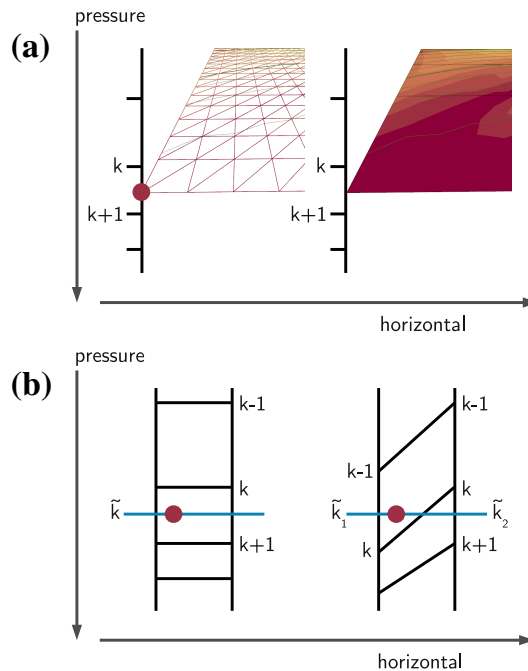
Figure 11. Sampling data fields in GPU shaders. **(a)** For each vertex of a horizontal section, model levels k and $k+1$ are found by binary search. The scalar value is linearly interpolated in $\ln(p)$ between these two levels. **(b)** PL grids are rectilinear (left), allowing the usage of trilinear hardware interpolation between the grid points surrounding a sample position (red dot). For ML grids (right), the sample position can be located between different model levels k for two adjacent grid columns, thus prohibiting hardware interpolation.
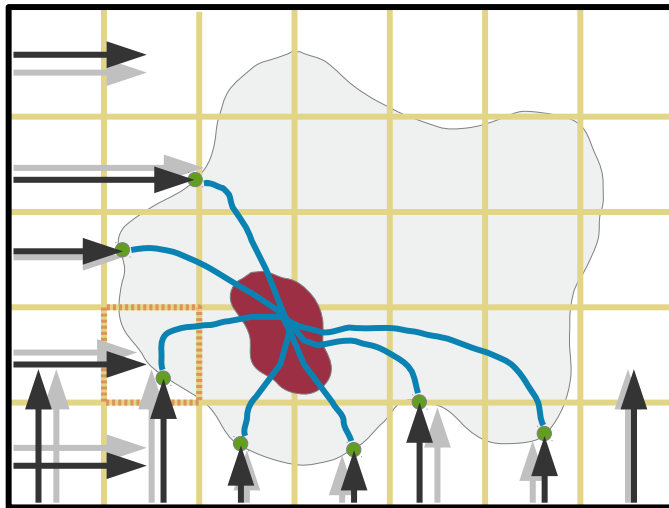
Figure 12. Computation of normal curves. Seeding points for the curves (green dots) are placed at the intersections between axis aligned rays (black arrows) and the outer isosurface (only rays from two directions are shown for illustration). Only a single seed is allowed in each grid box of the yellow volume.

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

|◄ | ►|

◄ | ►

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion
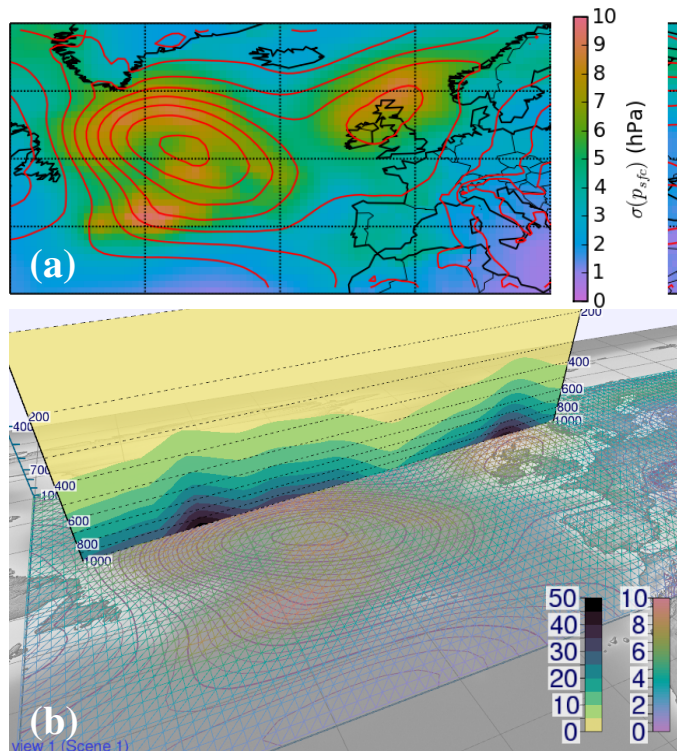
Figure 13. (a) SD of surface pressure, $\sigma(p_{\text{sfc}})$. Forecast of 15 October 2012, 00:00 UTC, valid on 19 October 2012, 18:00 UTC. Red contour lines show mean sea level pressure. **(b)** Vertical section of the pressure difference (yellow-blue-black colour bar in hPa) between highest and lowest ensemble member, rendered on top of a wireframe map of $\sigma(p_{\text{sfc}})$.
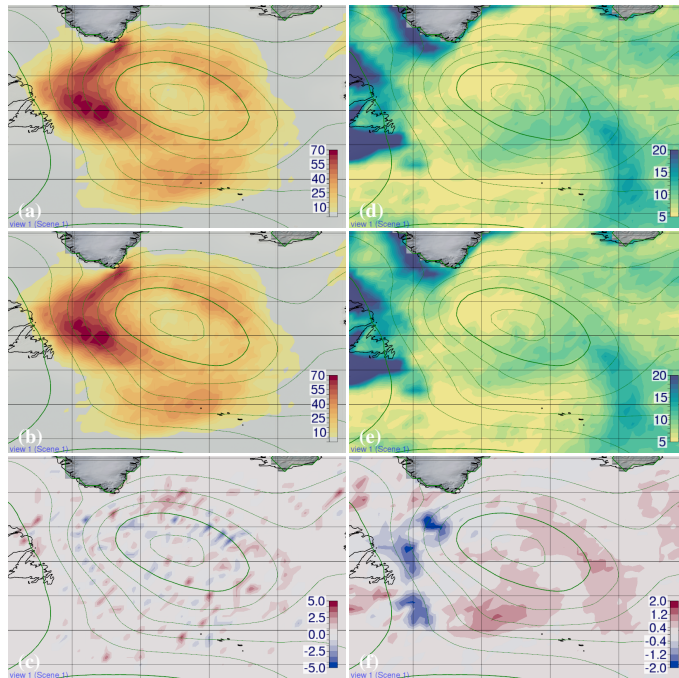
Figure 14. Visual differences between statistical quantities computed from a vertically regridded ensemble to those computed from the original ensemble. Horizontal section at 950 hPa (approx. model levels 51–55 in Figs. 15 and 16) of **(a–c)** $p(|\mathbf{v}| > 20\,\mathrm{m\,s^{-1}})$ (%) and **(d–f)** $\sigma(\mathrm{RH})$. Same forecast as in Fig. 13. Shown is **(a)** the probability and **(d)** SD computed from the original model grid, **(b** and **e)** computed from members regridded to the grid defined by the mean p_{sfc}, and **(c** and **f)** the difference between both fields.
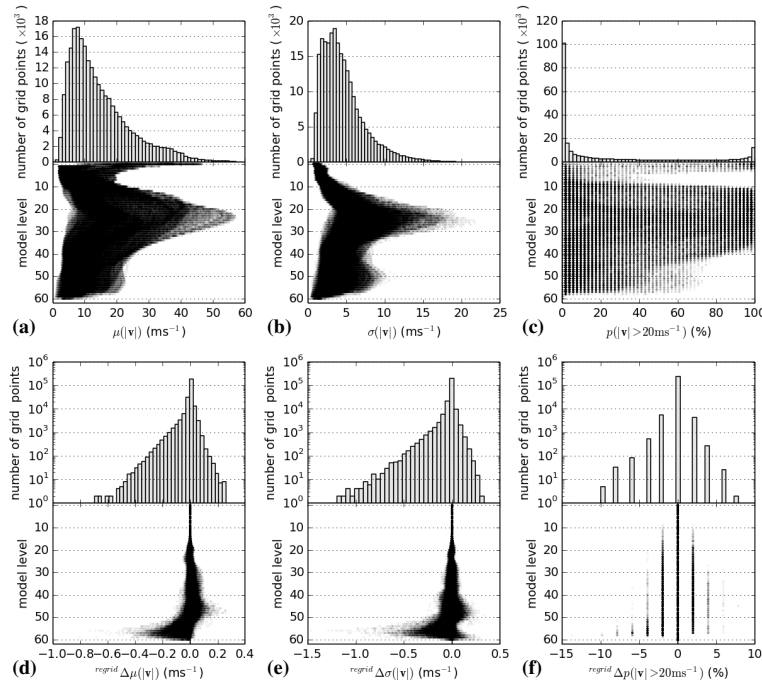
Figure 15. Distribution of differences between statistical quantities computed from a vertically regridded ensemble to those computed from the original ensemble. Plots are generated from all 256 742 grid points of the data field. Same forecast as in Fig. 13. Shown are (**a** and **d**) $\mu(|\mathbf{v}|)$, (**b** and **e**) $\sigma(|\mathbf{v}|)$, and (**c** and **f**) $p(|\mathbf{v}| > 20\,\mathrm{m\,s}^{-1})$. (**a–c**) Distribution and vertical occurrence of absolute values of the quantities. (**d–f**) Distribution and vertical occurrence of differences due to regridding (denoted by $^{regrid}\Delta$). Note the logarithmic scale of the histograms in (**d–f**). Probability values are discrete due to the size of the ensemble (51 members).
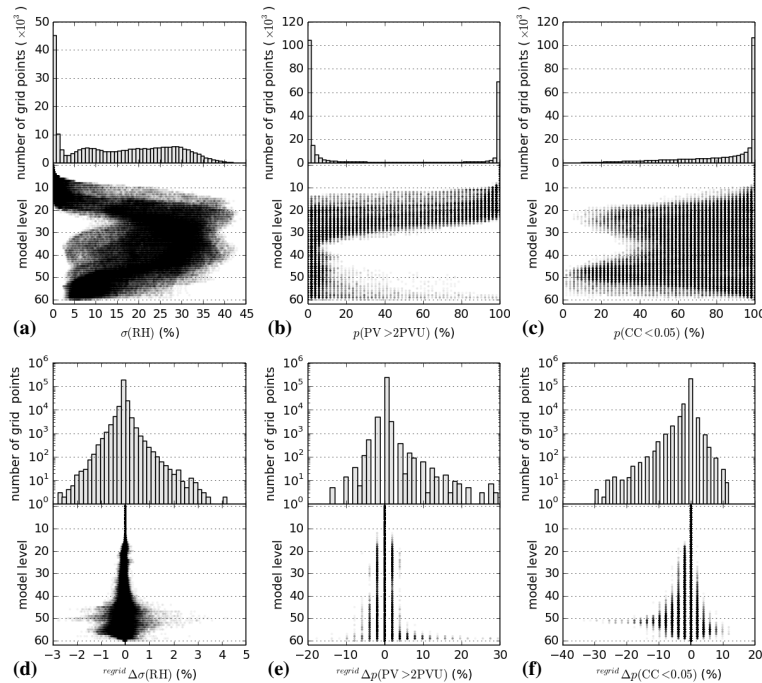
Figure 16. The same as Fig. 15 but for variables depending on moisture. (**a** and **d**) SD of relative humidity. (**b** and **e**) Probability of potential vorticity exceeding 2 PVU. (**c** and **f**) Probability of grid box cloud cover fraction falling below 0.05.

3-D visualization of ensemble weather forecasts – Part 1: Met.3D

M. Rautenhaus et al.
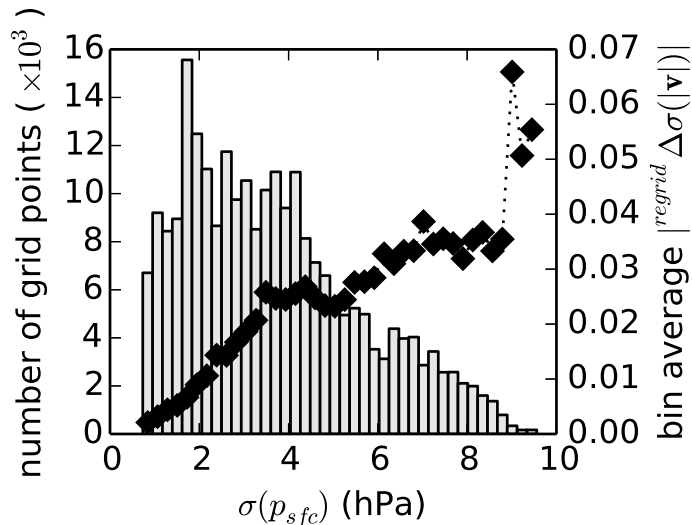
Figure 17. Histogram of $\sigma(p_{sfc})$, overlain with the bin-average difference of $\sigma(p_{sfc})$ against the differences between $\sigma(|\mathbf{v}|)$ computed from a vertically regridded ensemble and computed from the original member grids. Same forecast as in Fig. 13.
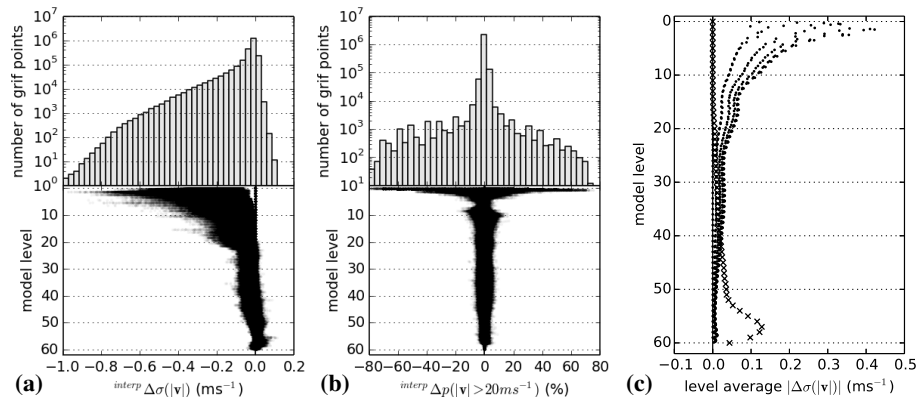
Full Screen / Esc

Figure 18. Distribution of errors due to vertical linear interpolation (denoted by $^{interp}\Delta$) of statistical quantities. **(a)** Distribution of errors of $\sigma(|\mathbf{v}|)$ (top), and vertical occurrence of the errors (bottom). **(b)** The same for $p(|\mathbf{v}| > 20\,\mathrm{m\,s^{-1}})$. **(c)** Vertical profile of level average differences due to regridding (crosses) and interpolation (dots). Same forecast as in Fig. 13.