Otto-von-Guericke-University Magdeburg
Faculty of Computer Science
ISG - Visual Computing

Bachelor Thesis

Tracking of mesoscale eddies in the Agulhas current in different spatial and temporal resolutions

Johannes Jendersie

18. February 2013

Verantwortlicher Hochschullehrer: Prof. Dr. Holger Theisel
Betreuer: Dr. Dirk J. Lehmann
Co-Betreuer: Dipl.-Ing. Maik Schulze

Abstract

The computational effort of earth system simulations is still a dominant problem in current climate research. Different spatial resolutions have been used as grid configurations in the models. In ocean simulations different flow characteristics occur due to these discretization. The grids with resolutions of 0.4° and 0.1° both permit mesoscale ocean eddies (a vortex-like structure), e.g. at the retroflection of the Agulhas current. There the main cause for eddy creation is the shear interaction with the Antarctic Circumpolar Current.

This thesis proposes a method to uniquely identify these eddies over time in different temporal resolution data sets. For the purpose of tracking these structures Feature Flow Fields and their stable extension are applied. Since the results have shown that both FFF methods have a huge problem with the temporal undersampling a third algorithmic approach is introduced. This improves the tracking results, but unfortunately it does not achieve the desired output. Consequently some suggestions for future improvements are given in accordance with the evaluation of the three methods. All three methods use the water velocity as input so that a data set with this properties is required to apply them.

Contents

1. Introduction

In-depth knowledge of the climate system becomes more and more important to understand the process of global warming. A part of this system is the ocean which stores and conveys huge amounts of salt and energy. Therefore the understanding of ocean currents with the help of simulations is an essential task of present climate research.

The massive computational effort and the subsequent data amount on high resolution models is a problem – even for modern super computers. The reduction of the model resolution is one possible method to decrease the data volume. However, this involves a compromise between simulation detail and the possible simulated period.

This thesis deals with the problem of comparing the quality of different model resolutions with respect to a certain property. In three different grid configurations of the Max Planck Institute Ocean Model (MPIOM) different flow characteristics occur. The low resolution model shows just laminar ocean currents which only represents large scale flows. Medium resolution, on the contrary, is permitting eddies, which are large persistent vortices that arise in the ocean currents. They convey huge amounts of warm saline or cold water. This behavior is similar to the observations. Finally, the highest resolution is eddy resolving. In this case a greater number of smaller eddies occur, which approximates reality. Whereby the role of eddies in climate simulation is still under debate. To improve the model the following question is most important: How substantial is the difference between the two eddy permitting model setups? Therefore this thesis depicts a way to track these structures and to obtain more information on model behavior.

In this context, the ocean region of the Agulhas current around south Africa has been chosen as case study example. Many eddies form in this region and huge amounts of salt and heat are transported north-westerly from the Indian Ocean towards the Atlantic.

1.1. Research Objectives

The aim of the research is to individually identify each eddy in a time series of simulation data. This is required to make reliable statements about the number and the characteristics of the eddies. Therefore it is necessary to reproduce the entire path of each eddy in its lifetime (so called tracking).

Such a tracking cannot be easily achieved in general. For this, exspecially the correspondence issue between related eddies at different time slices has to be solved. This is difficult because new eddies can emerge and already existing ones might vanish in bifurcations or through a loss of rotation speed. Moreover the simulation results are saved in monthly intervals. This wide span of time makes it much more difficult to correctly allocate corresponding features. Thus it is vital to overcome the problem of undersampled data in order to achieve the research objective.

1.2. Structure of the Thesis

To introduce the topic, various well-established techniques are presented in Chapter 2 and reference is given for further algorithms that have been used.

Afterwards Chapter 3 offers some knowledge for the understanding of the underlying model and the terminology. Before the algorithms can be applied the data has to be transformed in a way that the time axis can be accessed. Two different possibilities are introduced in Chapter 3.4.

In Chapter 4 different methods to solve the tracking problem are explained in detail. The first step is the extraction of the eddies (4.1) in one particular time slice. Then Feature Flow Fields [TS03] and Stable Feature Flow Fields [WTGP11] are applied (4.2). While these techniques solve the problem of moving, emerging and disappearing features in general, they might entail flaws at resolutions that are too low in time, which is clarified in the evaluation in Section 4.2.1. Subsequently a second approach for the context sensitive problem of tracking the ocean eddies is developed in Chapter 4.3. Thereby obtaining additional information on the flow characteristics supports the identification of corresponding eddy locations over time. The evaluation in 4.3.3 compares the algorithm with the two Feature Flow Fields and points out that the problem of tracking in undersampled data is still not completly solved yet. For this reason, some suggestions for improvements are derived from this evaluation.

The last Chapter 5 sums up the results and suggests approaches for future work.

2. Related Work

The tracking of features in 2D unsteady flow data is a common problem in flow visualizations and other scientific fields. Many different general techniques were developed in the flow visualization community. In climate analyses this is also an often appearing problem.

2.1. Tracking Approaches in Meteorology

In meteorology the most-well known area of application is the tracking of cyclones as done in [Hod94, BFL97]. Hodges used an image segmentation strategy on the relative vorticity map and tracked cyclones in 6-hourly data set. [BFL97] focuses on the eigenvalue based clustering of cyclone paths. In [JSSJ89] four mesoscale eddies in the Norwegian Coastal Current were tracked and characterized over a ten day period. For that experiment the satellite infrared data was examined. In meteorology observed real world data forms the basis for most applications. Thereby the main difference to simulation data is that a simulation provides much more values, such as the flow velocity. Due to this the velocity can then be evaluated as additional information to track the eddies.

2.2. Feature Flow Fields

The eddies can be followed by tracking the first order critical points of a 2D vector field. First order critical points have first been extracted and classified in 2D by [HH89]. There a classification of the points by the eigenvalues of their (2x2) Jacobian matrix was introduced which is used in this thesis to filter the critical points. In addition [HH91] raise the issue of classify the points in 3D. Again the eigenvalues of the (3x3) Jacobian matrix are recognized. The Bisection method described in [Gre92] is yet another approach for the localization of these points in a 3D flow. For that purpose the volume is bisected by an octree where a cell is subdivided if the topology degree of the node postulates a critical point inside the cell.

The critical points are general features, which can be tracked using Feature Flow Fields (FFF) [TS03]. These are vector fields derived from other scalar, vector or tensor fields. In such a field the integration of a stream line traces the feature location in time. This technique allows the handling of evolving and disappearing structures and was also applied to a sea simulation data set in the original paper. However, FFF tends to be unstable if the feature locations are numerically erroneous. The extension to Stable Feature Flow Field (SFFF) established by [WTGP11] introduces an additional component such that integration automatically converts to the features. In Chapter 4.2 the SFFF is applied to the MPIOM data set.

2.3. Vortex Extraction

Convenient surveys of topology-based vortex detection and visualization is given by [PPF$^+$10, Sah09]. [Wei08] is a more general approach for the extraction and visualization of vector field topology and [Sad10, chapter 2] also provides a good introduction into vortex topology and extraction. Moreover [Sah09] introduces a novel FFF for the tracking of vortex core center lines in 3D unsteady data sets. This could be an alternative solution to the given problem, but the potential simplification to 2D makes it unnecessary. For scenarios with many bifurcations, the creation and extinction of feature pairs, occur [KHNH12] suggest a merge graph as visualization technique. These tree like structure plots the bifurcations on one axis to the time.

The paper by [WS01] introduces a method to determine closed streamlines of a 2D vector field. The main idea is to find a region that is never left from a streamline. The same authors have also discussed the problem of tracking closed streamlines in [WSH01]. In contrast to this paper I have worked the other way around: I have used closed streamlines to gain further information on the tracking of critical points. [TWHS04] depicts another method, which guarantees to find all closed streamlines. This is achieved by a transformation to 3D where the cut of a well seeded streamsurface is a closed streamline. A modified version of this method has been used to extract the area of the eddies with a closed streamline. Namely the modification consists of a minimal spanning tree which was computed as seeding structure for streamlines. Therefore Prim's algorithm [Pri57] based on a Fibonacci heap [FT87] was implemented.

2.4. Further Related Work

There are completely different approaches to analyse 2D time dependent flows as well. One is the Finite-Time Lyapunov Exponent (FTLE) described in [SLM05]. The FTLE is based on a local analysis which maps the rate of separation of two points located infinitesimally close to each grid point. This visualizes the topology since the exponent increases between areas of different flow behaviors. For my research objective it is not applicable as FTLE does not solve the correspondence problem, but it could be used together with an image segmentation method to extract the eddies in one time segment.

For the vector field visualization the Line Integral Convolution (LIC) technique by [CL93] has been used and for some renderings the Illuminated Streamline method by [SZH97] is applied. Both methods are implemented in standard modules of Avizo which is a scientific visualization software. The algorithms in this thesis are implemented as Avizo modules as well and most of the pictures are created with this software.

3. Background

This Chapter describes the ocean model and the data underlying the analysis in this thesis. First of all some of the terminology is introduced.

3.1. Definitions

Critical Point This thesis only considers first order critical points. These are local features where the vector field vanishes which means that the magnitude of the vectors is zero. Depending on the characteristics of the local neighborhood these points can be classified [HH89]. Picture 3.1 illustrates the characteristics of different classes of first order critical points.

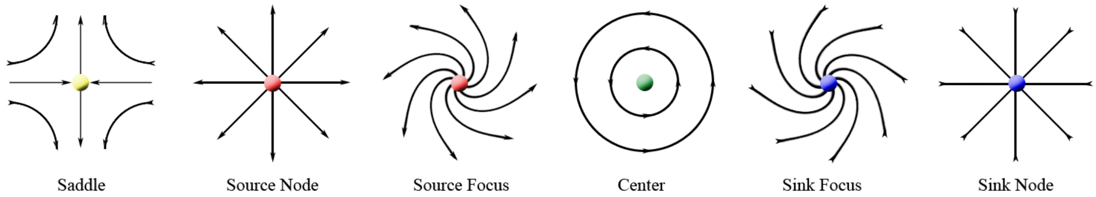

Figure 3.1. Classes of critical points, colors as in [Wei08]

Streamline Streamlines are curves whose tangent vectors are equal to the vector field at the curve's point. A streamline starts/ends in a critical point, at the domain boundary or is closed. Streamlines do not intersect each other except at critical points.

Vortex There is no standard definition of a vortex. I refer to the following one if speaking of vortices.

> A vortex is the rotating motion of a multitude of material particles around a common center [Lug79].

Further definitions can be found in [Rob91] and [Por97].

In 2D the common center is a source, sink or center critical point, whereas in 3D this center is a curve. Each point on this curve is again a source, sink or center in the plane perpendicular to the curve.

Mesoscale Ocean Eddies An *eddy* is a swirling structure in fluid dynamics. Eddies in the ocean with a size of 10 to 500km, which persist over months, are commonly called mesoscale eddies. In the following mesoscale eddies will be referred to as eddy. Eddies are special forms of vortices, because they fulfill the definition of a vortex (see above).

Rotation Sense - An eddy rotates cyclonic if it rotates counterclockwise on the northern hemisphere and clockwise on the southern hemisphere. Anticyclonic rotation is the other way around respectively.

3.2. Ocean Model

The examined model (MPIOM) is part of the Earth System Model (MPI-ESM) developed by the Max-Planck-Institute (MPI) for meteorology [GJR+12]. The MPI-ESM is a new model used in the Coupled Models Intercomparison Project Phase 5 (CMIP5) for the next Intergovernmental Panel on Climate Change (IPCC AR5) in 2013/2014.

The model is based on the components ECHAM6 for the atmosphere, MPIOM for flow simulations of the oceans, HAMOCC as ocean-biogeochemistry model and JSBACH for the land biosphere. OASIS3 is a separate coupler component for the ocean-atmosphere information exchange. Most advancements related to former models have been achieved in atmosphere simulations [AB12]. Besides a further improvement is the option to examine different resolutions in the model components.

Within the MPIOM there are several grid configurations with different resolutions:

Configuration	approx. Resolution	levels in depth	time resolution during simulation
MPI-ESM-LR	$1.5° - 167km$	40 levels	4320 s
MPI-ESM-MR	$0.4° - 45km$	40 levels	3600 s
MPI-ESM-STORM	$0.1° - 11km$	40/80 levels	600 s

Table 3.1. MPIOM grid configurations [JFH+12, vSEF+12]

[JFH+12] describes the simulation results of MPIOM configurations LR and MR and [vSEF+12] handles the STORM simulation. The MPIOM includes primitive equations, a free surface model, a dynamic/thermodynamic sea ice simulation and a bottom boundary layer scheme for the flow across steep topography. It uses an orthogonal curvilinear C-Grid with unevenly distributed level depth in the Z-coordinate [MHJ+03, JFH+12, vSEF+12]. A C-Grid is a grid where different values are located at different locations like the cell center/borders (see Figure 3.2). The resolution in Z direction is much higher at the sea surface which means the deeper the ocean the wider the layers.

The model setup of LR uses a bipolar grid with the poles located over Antarctica and Greenland. The higher resolutions MR and STORM have a tripolar grid with poles over North America, Central Asia and Antarctica. For curvilinear grids the supporting points lie on a sphere and have longitudinal and latitudinal curve coordinates. Inside

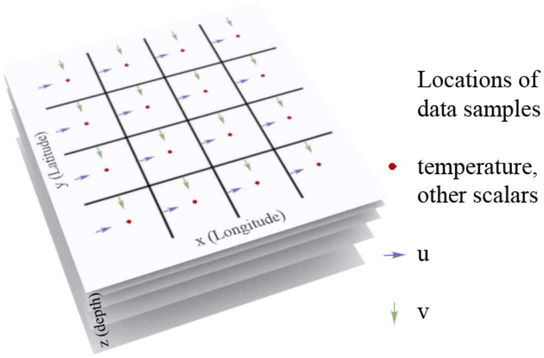

Figure 3.2. C-Grid with unevenly distributed depth

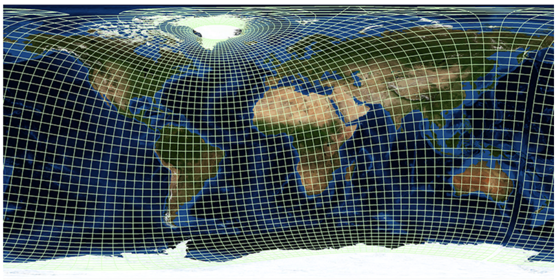

Figure 3.3. Curvilinear dipole grid (down sampled resolution; equidistant cylindrical projection)

the cells the coordinates are interpolated linearly. In the region of interest this is nearly regular as figure 3.3 illustrates.

From these simulations different data sets and their evaluations were saved. Due to the huge amount of variables, the high spatial resolution and the long simulation period of multiple hundred years the amount of data gets excessively large. More than 6 petabyte of data would be created per simulated day, if the full results of the STORM configuration have been saved. As a consequence not every simulation step is transferred into storage. Therefore the required variables of the velocity in U and V direction are merely provided in monthly intervals. This makes the tracking more complicated. In this paper different approaches are tested and evaluated in terms of their applicability to handle this problem.

3.3. Model Properties

In addition to the spatial and temporal resolution some more physical properties are of interest for the tracking process. In contrast to other 3D vector field data the mesoscale eddies are vertically aligned structures (see 3.4). The upward and downward velocity of water is very small in most cases, because the water generally forms stable stratified layers apart from locations where the ocean currents turn around. Another assumption of the model is that water is incompressible. This leads to a divergence free vector field. What follows from the small vertical water exchange and the incompressibility is that eddies must have center critical points and that there are closed streamlines around the eddies in theory. In praxis the discretization in the simulation and the reconstruction through interpolation adds noise to the data and corrupts these structures.

3.4. Preparation of Simulation Data

The size of volumetric data over a large period of time accounts for multiple terabyte. This Chapter shows how to reprocess the data to make an in-core approach working. Furthermore the preparation simplifies the tracking problem from 3D unsteady to 2D unsteady.

The first step is to reduce the amount of data. Therefore the area of interest is extracted and unused variables are discarded. The stages of this process are shown in the appendices A.1 and A.2.

3.4.1. Handling of Initial 3D Data

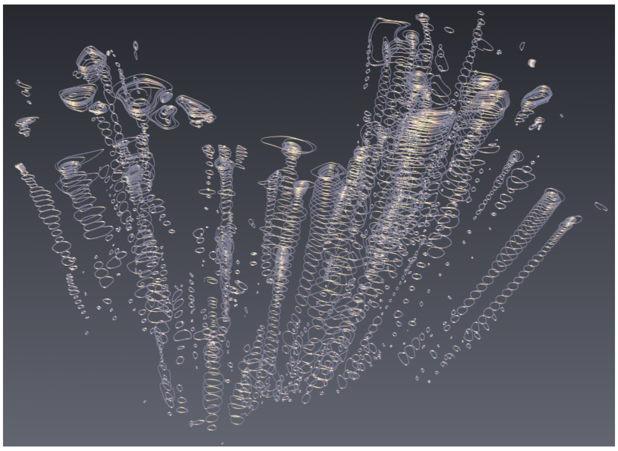

Figure 3.4. Extracted eddy boundaries at all depths (6m - 5720m) of MPI-ESM-MR data

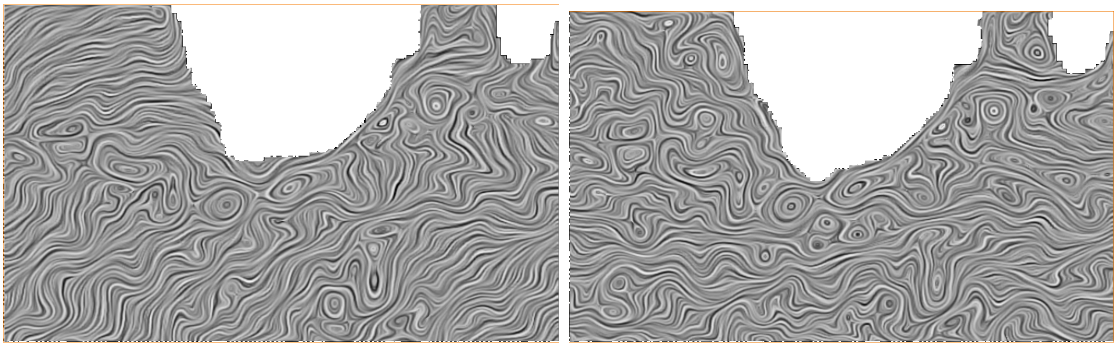

Figure 3.5. Direction of flow at 3m in MPI-ESM-MR

Figure 3.6. Direction of flow at 100m in MPI-ESM-MR

In the initial unsteady 3D data sets the eddies are contained as 3D vortices. It is necessary to extract the vortex cores prior to applying the tracking. Figure 3.4 shows – especially at the right side – that the eddies spread across a depth of more than 1000 meters. At the surface they are strongly influenced by the wind which is most observable in the upper left corner. The Line Integral Convolutions in 3.5 and 3.6 show the differences between the flow direction in a depth of $3m$ and $100m$. In 3.4 it can be seen that the top layers have incorrect eddy boundaries. A tracking of the structures in the upper $100m$ is therefore unstable. Deeper in the ocean the ground changes the flow behavior as well. This does not account for as much disturbance as the wind, because at the bottom the layers of the grid are much wider than within the upper layers. In regions where the ocean depth is shallow the eddies do not spread as deep as in other regions.

Averaging over Depth

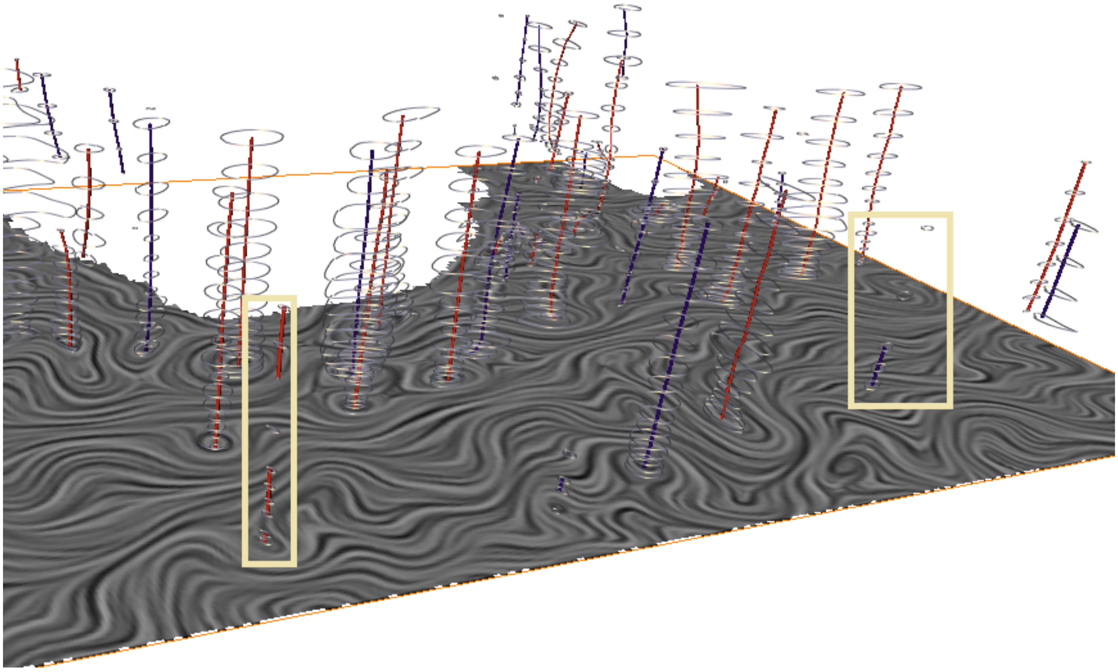

Figure 3.7. Extracted eddies in layers 8-23 of MPI-ESM-MR; color coded sense of rotation: blue = cyclonic, red = anticyclonic

The fact that the eddies are so stable over a large range in depth allows us to break them down to 2D. For the monthly data sets the vertical average, weighted by the layer depths, of each data value is computed. In this context, only the upper $1000m$ are considered (number of layers: 23 in MR, 42 in STORM). The averaging removes physical noise and preserves the basic structures well. Unfortunately, the first layers influenced by the wind, add noise to the average approximation. Since these upper layers are comparatively thin the impact on the result from averaged data is negligible. Removing the upper hundred meters leads to a more stable initial situation for the algorithms. To this end, the first eight layers in MR and nine layers in STORM were deleted. Figure 3.7 shows the alignment and structure stability in the layers 8-23 of the MR resolution.

The figure is the result of the algorithmic tracking solution applied to the steady 3D field. In the marked boxes errors from physical disturbance are observable. The eddies are at this points too weak to be identified for each depth.

The cdo-commands for averaging over depth are listed in appendix A.3.

Single layer

Another option to reduce the vector field to 2D is to select a representative single layer. This layer should be in a depth between $100m$ and $1000m$. Above $100m$ the wind adds too much disturbance and below $1800m$ the ground covers to many areas. Using a deep layer below $1000m$ also discards smaller eddies and could be influenced too much by the bottom topography depending on the proximity to the coasts. Choosing only one layer is numerically less stable, but nonetheless the algorithms run stable. This approach has to be used in the case of the high-time-frequency test data, because in this case data is only provided for the layer at a depth of $75m$. Although I would not recommend to use a depth of less than $100m$, it still works in this particular case. In a depth between $10m$ and $100m$ the structures are mainly the same. Only their location changes slightly and the extraction of areas tends to find very close borders which is not always representative.

The cdo-command to select a layer can be found in appendix A.4.

Consequences of Preparations

- Both methods reduce the data to a 2D unsteady case. In that domain one can apply well-known tracking approaches as (S)FFF. For the given problem it is unnecessary to track the 3D vortex cores.

- Working with a chosen layer or an average of several chosen layers reduces numerical and semantical (wind, ground) noise.

  Noise is normal distributed and has a mean of zero in general. Averaging removes the noise because the more values are used the nearer is the mean of the noise to zero. The single representative layer reduces the disturbance in respect to many of the other layers but it does not remove the noise anyway.

- The massive data amount is reduced by a factor of 40 to 80, depending on the model resolution.

- The information about the eddy depth is lost. If available this characteristics could be very helpful for the eddy identification process.

3.4.2. Restructuring of netCFD Files

Up to now the data was of the format netCFD with the CF-conventions 1.0. A file consists of a 3D field including velocity and other values over several time steps. After the previous preparation step the third dimension is reduced to a size of one. The time dimension is the record dimension (see Unidata netcfd format specification) for streaming. The potential unlimited time steps are loaded separately from each other in Avizo. To track the eddies in time the algorithm needs to access more than one point in time at ones, which is technically not possible in Avizo. The tool cdftts (short for 'netcdf time to steady') included in this work restructures the file and swaps the record dimension with the last non-record dimension. The data is loaded as a 3D steady field afterwards, whereas the time steps are the layers in the 3D field and the former slices are time steps now. This files do not have the CF-convention 1.0 anymore and can therefore cause minor errors in Avizo.

4. Approaches

In this chapter the Feature Flow Fields and the algorithmic approach are explained in greater detail. It is assumed that the precomputation of the data is finished. At first common steps of both approaches are explained.

4.1. Identifying Critical Points

The first step applied for each method is the feature extraction. The center critical points – representing the eddies – have to be found. For the (S)FFF these points are used as seed points for a streamline integration. The biased algorithmic approach requires them to extract the entire eddies and uses the extrema as their path points.

In the region around south Africa the curvilinear grid is almost regular. Hence the search can be done directly in the grid space. A simple linear search for local minima is used as method. Thereafter every point found is classified. At each point the classification is done with the Jacobian matrix. The first derivative is calculated as the value-difference of adjacent samples divided by their distance. This yields the four values for the 2x2 Jacobian matrix (4.1).

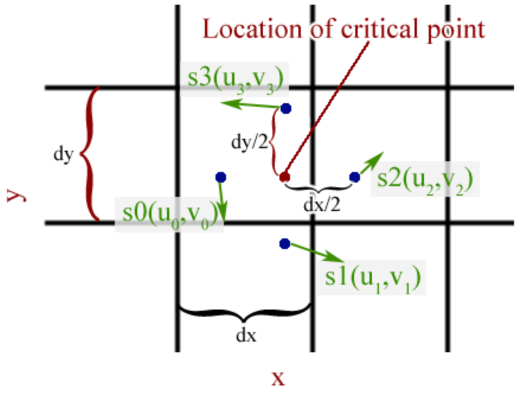

Figure 4.1. Sampling of velocity vectors on a regular grid

$$J = \begin{pmatrix} \delta u/\delta x & \delta u/\delta y \\ \delta v/\delta x & \delta v/\delta y \end{pmatrix} = \begin{pmatrix} \dfrac{(u_2 - u_0)}{dx} & \dfrac{(u_3 - u_1)}{dy} \\ \dfrac{(v_2 - v_0)}{dx} & \dfrac{(v_3 - v_1)}{dy} \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \tag{4.1}$$

The eigenvalues of J for the classification can be calculated straight forward by computing the roots of the characteristic polynomial of degree 2.

$$0 = det(J - \lambda I) \tag{4.2}$$

$$0 = \lambda^2 - \lambda * (a + d) + (a * d - b * c) \tag{4.3}$$

The complex eigenvalues classify the critical points as follows: Let λ_1 always be the smaller of the two eigenvalues in respect of the considered component (Re or Im).

Class	Eigenvalues					Poincaré-index
Source	0	$<$	$Re(\lambda_1)$	$=<$	$Re(\lambda_2)$	$+1$
Sink	$Re(\lambda_1)$	$=<$	$Re(\lambda_2)$	$<$	0	$+1$
Center	$Re(\lambda_1)$	$=$	0	$=$	$Re(\lambda_2)$	$+1$
Saddle	$Re(\lambda_1)$	$<$	0	$<$	$Re(\lambda_2)$	-1

Table 4.1. Classification of critical points

Derived from the model (Section 3.3) only the center critical points represent eddy centers. These critical points are structurally unstable. Using $Re(\lambda_1) = Re(\lambda_2) = 0$ will fail due to numeric noise in the model. Usually, sources and sinks only occur if considerable amounts of water are moving down- or upwards. The incompressibility of water assures that sources and sinks cannot be caused differently. If the averaging of layers reaches deep enough this effect disappears since the vertical stream has a source on one end and an equally strong sink at the other. Every critical point with an index of $+1$ is noted as a potential eddy center. Some of the noted points are generated by noise or vertical streams in the ocean. Therefore the resulting set includes all existing eddy centers and some supernumerary points.

4.2. Stable Feature Flow Fields

A Feature Flow Field is a vector field derived from other scalar, vector or tensor field(s) such that stream objects in the new field follow the desired features of the initial field(s) [TS03]. After the computation of the FFF standard integration schemes can be used to track the features. There are numerous ways to derive this field from different data sets. I will briefly explain the case for a series of 2D velocity vector fields, where the desired features are the critical lines over time.

First of all, one considers the 2D unsteady case to be a 3D steady vector field. Let u and v be the components of the velocity vector field. In the new 3D field the gradient for u and v consists of three components. They point into the direction of the highest spatial and temporal change. The cross product of the two 3D vectors is perpendicular to both components and therefore points into the direction where both values do *not* change.

$$f(x, y, t) = \nabla u \times \nabla v \tag{4.4}$$

These vectors already form the FFF as for each point in the field they point into the direction where the velocity remains constant. If one seeds a streamline at a critical point with velocity 0, this line will stick to the zero point in each time period, unless the streamline vanishes in a bifurcation or reaches the domain boundary.

The raw FFF has been applied to the MPIOM data and was rejected for being numerically too unstable. The extension to Stable Feature Flow Field adds a second vector field in which streamlines converge to the critical lines.

$$g(x, y, t) = \frac{f}{\|f\|} \times (u \cdot \nabla v - v \cdot \nabla u) \tag{4.5}$$

Critical lines of the correction field g are feature lines of f. This is a direct consequence of the Equation 4.5, because if $u = v = 0$ applies, g resolves to 0. So g is zero at critical lines of the velocity field. Furthermore g is perpendicular to f and points into the direction of the feature lines in f.

The SFFF is defined as the weighted sum of f and g.

$$h(x, y, t) = f + \alpha g \tag{4.6}$$

In [WTGP11] α has been chosen according to the eigenvalues of the vector field. For the application to the MPIOM data α is chosen experimentally between 5 and 8. A higher weight tends to produce a fluctuating behavior of the streamlines around the feature lines. A smaller value converges slower and some faulty streamlines occur.

Note that a backward integration of streamlines in h diverges from the feature lines. The backward integration can be achieved by using a modified version of the field $h' = -f + \alpha g$. Since this requires two different fields only the forward integration is used for the experiments.

As the streamlines are tracking the critical points, the seeding at each point of time results in a high number of overlapping streamlines. Clustering can reduce all redundant streamlines to one per feature.

4.2.1. Evaluation

The aim of the thesis was to achieve a valid tracking of eddies based on a monthly data set. The result at a temporally high resolution with SFFF or the biased approach is assumed to be the ground truth. This seems reasonable since the results match the expectations derived from expert knowledge. Moreover both approaches show similar behaviors.

Experiment Setup

Only one data set where the velocity was given on a high temporal resolution was available. It is an hourly snapshot of a simulation with the STORM grid-configuration of the velocity (u, v) in a depth of $75m$. The influence of the atmosphere still exists at this depth, but is negligible.

The data set is down sampled to daily, 5-day, 10-day, 20-day and 30-day resolution to show the development of the results depending on the temporal resolution. The related pictures for the tracking results in the Feature Flow Field and the Stable Feature Flow Field are listed in appendix B. To reduce the visual clutter only streamlines seeded at time step 0 are shown. So there are much more eddies in one year which are not represented. As integration scheme an adaptive Runge Kutta 4 integration has been used.

Discussion of FFF and SFFF Results

At one point in time about 300 critical points with a Poincaré index of one have been detected. More than a third of these points comes from noise and do not represent an eddy. This input could be improved by a better method for the critical point detection. Since an eddy persists over months it is represented by a long streamline. Most small streamlines at the bottom originate from noise within the seeding of critical points. Their behavior can be ignored. It would be useful to automatically remove them.

As expected there are problems in the FFF if the streamline does not lie directly on the feature. Then it diverges and results in a tangle. This happens even under the daily resolution. Comparing SFFF and FFF, the FFF contains almost all streamlines from the SFFF. Therefore an analysis of the lines in the FFF could be used to detect the locations where the streamline begins to diverge and cut it there.

The development of both methods over time is very similar. Up to the 10-day resolution both contain the true eddy paths. In the FFF the number of lines from noise decreases, because the short time non-eddy critical points of adjacent time steps match less and their tracking gets lost at first. In all time sections from the 5-day to a 30-day resolution the SFFF shows some very straight lines. These are always located at the coast and appear to be a result of noise.

In the 20-day and lower resolutions more straight upward lines appear. This is a severe problem. The period in which eddies pass an observed point seems to be nearly a month (In Section 4.3.3 this is shown in an experiment). If the temporal resolution is close to that period, critical points from different eddy centers are recognized as one feature.

This is due to the fact that neighboring eddies move towards the position of the previous ones. Surprisingly FFF have a higher accuracy than the SFFF at a 20-day resolution. The additional component in SFFF which assures the converging behavior towards the critical points supports this wrong matching effect. Derived from the velocity the feature flow direction of the critical point is more accurate than the property to converge to the *nearest* feature. In this context *nearest* means that in the vector field topology no separatrice is located between the streamline and the critical point, so that they are in the same region.

Another experiment on a dataset of MR-monthly-resolution has been carried out. In the medium resolution fewer, but larger eddies occur. Due to more overlapping of the eddy areas the results are expected to be better than in the STORM configuration. The Figures B.16 and B.17 show that this is not the case. Here no picture of the ground truth is available, but towards the west of Africa the eddies should move in a north-westerly direction again. The FFF illustrates this characteristics, but produces diverging errors after some time. The stable variant does not track the features at all.

Both FFF and SFFF are not able to track the eddies correctly in a monthly resolution, because of the undersampling of the eddy frequency. It does not matter which spatial resolution is probed. In both eddy containing resolutions the (Stable) Feature Flow Fields produce no useful results. In the following a biased approach is developed, which tries to overcome this problem by using additional semantic information.

4.3. Biased algorithmic approach

If there are too few time steps the correspondence of eddy center points cannot be detected correctly with SFFF. However, the ocean model offers more information to solve this problem.

The critical points from different eddy centers are not distinguishable, but the eddies can be specified. About half of them are rotating cyclonicly and half of them anticyclonicly. Furthermore they can have a different size, rotation speed or temperature. If two or more points of the path are connected the movement direction is known. It is unlikely that an eddy suddenly changes its direction completely. This knowledge could help to design a more stable algorithm for the tracking problem. In the following an eddy extraction and voting algorithm is introduced, which uses this additional information to resolve the critical point correspondence. With the extra information faulty links can be dismissed and better ones should be found.

4.3.1. Eddy Extraction

The set of potential eddy center critical points (Section 4.1) are used as input again.

The next step is to choose a representative area for the eddy. The first idea was to use the largest closed streamline which encloses one critical point. This streamlines should exists due to the model properties (see Section 3.3). Closed streamlines can guaranteed be found with the approach introduced by [TWHS04]. As seeding structure I have used the minimal spanning tree (MST) of all source, sink and center critical points. The original paper suggests a randomly ordered path of these points. This would enlarge the size of the seeding structure and slow down the algorithm. A MST also covers the required properties and therefore can ensure that all streamlines are found, too. Both the random path and the MST have to be connected with the domain boundary to fulfill all necessary properties. This is achieved by adding a dummy point at the boundary.

The Figure 4.2 shows that many eddies cannot be found with this method. Two reasons are possible: On the one hand, the vertical water velocity could be nonzero and the critical center constitutes a source or sink without a closed streamline. On the other hand, closed streamlines are structurally unstable and can be lost through numerical errors and noise in the field. Both cases cause that the streamlines have a small gap at their seeding point, if they are integrated 360° around the center. They have a spiraling characteristics.

The solution is to choose-pseudo closed streamlines (Figure 4.3). In this context, a pseudo-closed streamline is a streamline in 2D which starts at an arbitrary non critical point in the vector field and ends on the line defined by the enclosed critical point and the starting point. It has no gap if projected to a circle around its critical point. In other words the critical point inside this line cannot *see* a gap. The distance between the two end points is the error in respect of the real closed streamline.

To find the pseudo-closed streamlines the same seeding structure – the MST – is used again. On each edge, that is a straight line between two adjacent points, a set of streamlines is seeded. Figure 4.4 shows all streamlines seeded in this way in a part

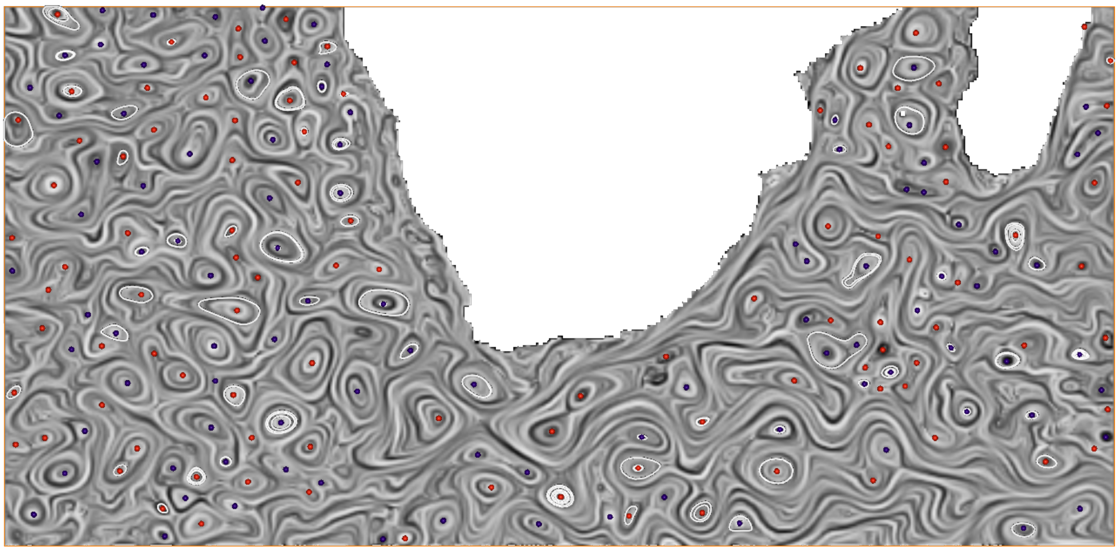

Figure 4.2. Detected closed streamlines and their center critical points

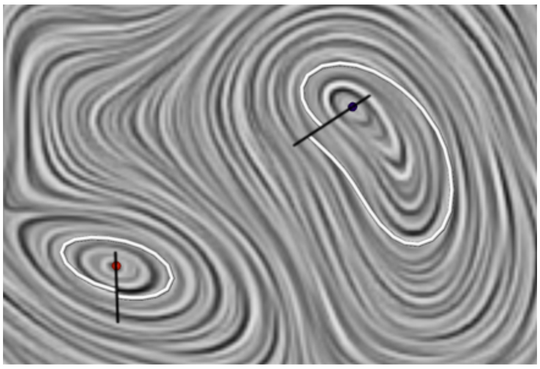

Figure 4.3. Pseudo closed streamlines (representatives only)

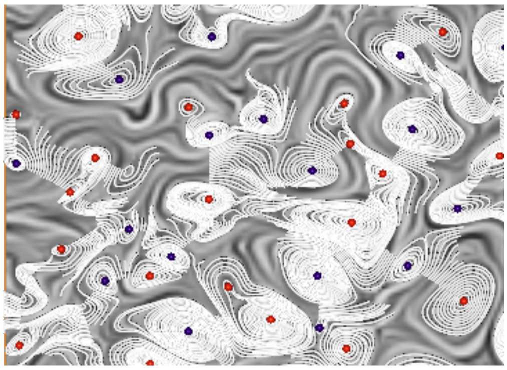

Figure 4.4. All seeded streamlines

of the region. Not all of these lines are used, because they are not pseudo-closed with regard to any critical point. The pseudo-closed lines are assigned to the critical points respectively.

For each point with at least one line, a representative line is chosen and the point is accepted as eddy center. There are points without any line. These were local extrema in disturbed regions, which can be discarded now. It is possible that even a noise point can have a pseudo-closed streamline. In that case it can have detrimental influences on the tracking results.

If more than one line is presented the median of the error is computed. The median multiplied by a constant β gives the relative error threshold of all streamlines from that point. From all remaining lines with an error less or equal to the threshold, the longest line is chosen. This yields one representative pseudo-closed streamline for each eddy center. In the Figures 3.4 and 3.7 this algorithm was used to show the alignment. The pictures also show that the extraction of the representative line is quite stable.

Afterwards different parameters are extracted from the derived area. To compute the radius the Haversine formula by [Sin84] is used. The rotation direction and the rotation speed are calculated by the average velocity sampled on the pseudo-closed streamline and the radius. Temperature, salinity and other parameters could be derived from area integrals. This has not been done, because these values are relatively similar for all eddies and would not add enough new criteria to distinguish them.

4.3.2. Building Paths by a Voting Strategy

The extraction process gives us a set of points with a number of different parameters. To find vortex cores or the path over time one has to connect these points in a meaningful way. This is done by finding the most probable next point with a coherence criterion.

For each point $p \in P$ all other points $q \in P, q \neq p$ are considered as the potential next point. Every point q votes with its coherence criterion $c(p,q) = c(q,p)$. It is possible that the point q has voted for a third point $r \in P$ and was chosen as r's successor. It only votes for p if the coherence is higher than for r $(c(p,q) > c(r,q))$. r loses its next point in that case. It can obtain a new one if a second vote pass is done for all remaining points from the first pass. It is not possible that assignments toggle between two points, because the criterion c remains constant for a pair of points and the assignment is only changed if c is greater than for the prior vote. This algorithm maximizes the coherence locally inside each path. It does not maximize the global $\sum c(p,q), \forall p,q \in P, p \neq q$. This is considered to be useful, because the mesoscale eddies are stable structures over a long period of time and optimizing single paths avoids the influence of less stable structures and noise.

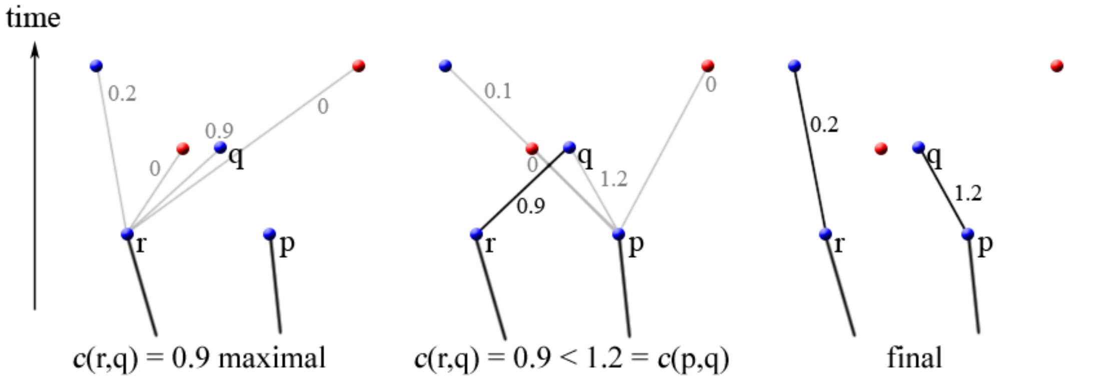

Figure 4.5. Voting steps for a successor for p and r; Colors denote the rotation direction

Choosing the Coherence Criterion c

The quality of the paths depends on the function c. Two different types of criteria have been chosen: total and relative. The total criteria have to be fulfilled, otherwise the vote is 0. The relative criteria are proportional functions depending on one value from the eddy parameters.

The first total criterion is the rotation direction. It is not possible that a structure entirely changes its direction without vanishing in between. The benefit of this criterion is very high, because cyclonic and anticyclonic eddies are distributed uniformly. Removing half of them to search paths in the remaining half enlarges the spatial distance in average by a significant amount. So the nearest point in the next time step is now more likely to be a good successor.

The second total criterion assures temporal causality. Votes are only carried out towards one temporal direction. Points in the same time segment or in the past are not recognized. This builds paths only in one direction.

As last total criterion I have introduced a threshold maximum distance γ. The threshold is an input parameter of the algorithm. This excludes that structures can move too far in one segment of time. That is important because otherwise a vanishing and an evolving eddy within the same time period could be interpreted as one eddy, even if they are more than a thousand kilometers away.

As only relative criterion the inverse distance is used. This means, the nearer the points the more weighted their vote.

4.3.3. Evaluation

The experiment is done with the same data set and the same conditions as for the (Stable) Feature Flow Field before. Now it will be discussed if the biased approach can improve the results.

The results of applying the algorithm to the data set are presented in the Figures B.3, B.6, B.9, B.12 and B.15. The coloring shows the rotation direction as additional information.

The algorithm output for the high resolutions up to 10-day is very similar to that of the SFFF. The lines of the SFFF are a sub-set of the ones in the algorithm. The lower the temporal resolution the more additional paths are found. These new paths are not visible in the ground truth and are therefore wrong. Again, the reason is that different eddies are being interpreted as one due to sampling problems. Looking at the 20-day or 30-day results the algorithm shows some improvements. It still finds some of the correct paths (see the left-hand side). Furthermore the wrong traces have a zigzag pattern which they did not show in the FFF methods. It is possible to exploit that knowledge for further improvements.

The figures reveal that cyclonic and anticyclonic eddies are mixed. Looking only at one color the spatial difference between individual eddies grows. There are more false lines and a less intense zigzag pattern, if the rotation criterion is not applied.

Until now the algorithm shows slightly better results in the high spatial resolution than the SFFF, because it still contains the true results in monthly resolution. However it does not solve the problem and produces overly deficient results to be useful.

The result of the equivalent experiment in MR-monthly-resolution is shown in Figures B.18 and B.19. The first can be compared with the other experiments with monthly resolution. The second picture shows all paths found during the whole year which is possible now without visual clutter, because fewer, larger eddies occur in the (spatial) medium resolution. The result seems very reasonable compared to the expectation that eddies move in the north-westerly direction. At some points the zigzag pattern is visible again, which is assumed to be wrong. Nonetheless, the algorithm output on medium resolution is an enhancement compared to the other resolution or the (S)FFF approach for the same data. With some minor improvements the algorithm can at least succeed in the medium resolution and perhaps on the STORM resolution as well.

Horizontal Eddy Velocity

Until now it has been said that the problems occur because of the periodicity of eddies. To prove this right some values have been derived from the experiments. For the STORM configuration the results of the daily resolution are well-conditioned. The average velocity is computed from the extracted paths. To this end the horizontal distance between two path points is calculated with the Haversine function. The time between two points is also known (one day) so that the average velocity of an eddy equals the average of all those samples. The same was done for the MR grid. In this case the ground truth is not known and the results of the monthly resolution has to be used. The numbers are not correct and can be used for discussion only.

Configuration	Average velocity	Maximum velocity
MPI-ESM-MR	$0.027908ms^{-1}$	$0.086264ms^{-1}$
MPI-ESM-STORM	$0.060173ms^{-1}$	$0.228716ms^{-1}$

Table 4.2. Horizontal velocity of extracted eddies

At a velocity of $0.03ms^{-1}$ a distance of $77.76km$ is covered in a month. For a velocity of $0.09ms^{-1}$ more than $233km$ are mastered. With the absolute maximum velocity of $0.22ms^{-1}$ this distance increases to $570km$. The size of the eddies is between $10km$ and $500km$ in reality. Although in the experiment the small ones are not resolved for the prevailing number of eddies their size and their velocity in $km/month$ is in the same range. This proves that the observed problems originate from the undersampling of the eddy frequency.

Possible Improvements

The experiment has shown that the undersampling could be evened out with an algorithm which uses semantic information. The current version effectively benefits from the rotation criterion. There is more information that could be involved – of which movement prediction is the most promising.

Movement Prediction There are two possible options to predict the next path point and involve this knowledge in the voting function c. The first approach is to *penalize* the zigzag-like patterns. Assuming that two or more path points are already found properly a regression line could be computed for the prediction of the next location. The problem here is that the assumption that two or more points belong together is not necessarily true.

The second possibility would be to compute the FFF at a path point and use the resulting direction vector for the prediction. This is derived from the fact that the FFF often finds the correct directions in a 20-day resolution. The problem is with a lower sampling the gain of that direction is reduced. The advantage is that this can be done for one single point.

Both can be combined to achieve a higher level of certainty. Therefore compute the cross product of the velocity's gradient (FFF) at two points of interest and the direction vector between the two points. If all those vectors point in the same direction the voting weight is increased. If two or more points are already connected compare this newly derived direction with the regression line. If the average error in respect to the regression line is too high, the path should be highlighted as wrong or be discarded.

Parameter Distance Size (area), rotation speed, shape, position, temperature, salinity and more parameters are available for each eddy or could be computed. With these parameters the distance metric can be redefined to more than three dimensions. This helps, if the eddies observable at one location are distinguishable in at least one of the parameters. The problem will be to find appropriate scales for the different values.

Clustering and Machine Learning The entire problem can be seen in another way. Let all critical points of one eddy belong to one and only one class. Then the problem of finding all classes is a standard clustering or learning problem. Those learning methods optimize the clustering locally or globally which all could be tested to solve the tracking problem.

5. Conclusion

In this thesis Feature Flow Fields and their stable variant has been applied to the MPIOM simulation data in order to track mesoscale ocean eddies. As revealed by the evaluation, the temporal undersampled data changes the Feature Flow Fields so that they do not track features anymore. In the non-stable FFF the problem of diverging integration is observable. On a low temporal resolution the evolving streamlines connect critical points of different features in both methods. The artifacts caused by undesampling are less stronger for the case of the FFF. The stabilizing component of the vector field seems more likely to have this problem.

Afterwards an algorithm was been introduced which should overcome the undersampling by additional semantic data. To this end some eddy properties have been computed: center point, rotation direction and speed, size and other parameters are available to support the tracking. With these criteria a voting strategy has been done to assign succeeding points to a path. Effectively only the rotation direction and the center point distance are used for the voting. The other criteria offer possibilities to extend the algorithm in the future. So far the results could be improved compared with that of the two FFF methods. The errors of the algorithm due to the temporal under sampling are that some paths are too short and unconnected. Additionally, zigzag patterns occur if paths of different eddies get mixed up. In contrast to the Feature Flow Fields the spatial resolution plays a vital role. For the MPIOM-MR configuration the unique identification of eddies has nearly been achieved.

5.1. Performance

In general, the SFFF performs better than the indroduced approach. Regarding this the computational time of the SFFF themself is the bottleneck, even though it has been calculated at interactive speed for the high resolution experiments. Subsequently computing the path line of a feature is a simple streamline integration, which is very fast. The algorithmic approach took several minutes for each step. The introduction of parallelism can give a significant performance boost and interactive speed could possibly be reached. Moreover, the whole process is very intense in use of memory. If the limiting factor is performance or large data sets need to be handled the SFFF method is preferable.

5.2. Future Work

To overcome the problem of temporal undersampled data the algortithm needs to be improved. In Section 4.3.3 some alternatives are listed to do that. A completely else approach would be necessary if only values such as *sea surface temperature* are available. This applies to observed data whereby simulation offers this value as well. Using the *sea surface temperature* the reality and simulation could be compared in respect to their eddy characteristics. One could think of another prospective approach that handle issues which are caused by temporal undersampling artifacts.

To achieve a higher level of information by the visual output a clustering could be applied on the extracted eddies. In this thesis the main aim was to identify unique eddies. If a clustered variant is of interest it could be possible to use all eddy center points and their parameters without any tracking and without regarding the time axis. The cluster representative could be derived from a regression of all points of a special parameter class.

It would be beneficial to run the algorithm during the simulation. That would add a control component for climate model simulation experiments. Thereby runs with a faulty parametrization could be identified by the criterion of proper eddy characteristics. Therefore the performance should be improved. The computation time can be reduced by parallel computing and some additional optimizations. The memory consumption could be reduced by an out-of-core approach for the time periods. For each visible time segment the extraction has to be done in-core. The voting strategy could be repeated for the last n time stages based on the extracted parameters after a new segment has been analyzed.

The developed algorithm should facilitate the understanding of how ocean simulation models could be improved. Therefore an interpretation of the eddy characteristics is necessary in the future.

Acknowledgments

I would like to thank my supervisors Dirk J. Lehmann and Maik Schulze for constantly sparing their time for me and for the advices on how to improve my work. They have also provided a lot of material and references, which I would never had found by myself. I thank Helmuth Haak, Niklas Röber and Michael Böttinger from the DKRZ for providing me with task and the data and for helpful discussion. Also I would like to thank Kristina Aßmann Gramberg and Marie Lambert for helping me with the language as this was my first written work of this complexity in English.

Bibliography

[AB12] A.Kirk and M. Böttinger. Neues Erdsystemmodell des Max-Planck-Instituts für Meteorologie. http://www.mpimet.mpg.de/fileadmin/grafik/presse/Forschung_aktuell/PDFs/120828_MPI-ESM_dt.pdf, December 2012.

[BFL97] R. Blender, K. Fraedrich, and F. Lunkeit. Identification of cyclone-track regimes in the north atlantic. *Quarterly Journal of the Royal Meteorological Society*, 123(539):727–741, 1997.

[CL93] Brian Cabral and Leith Casey Leedom. Imaging vector fields using line integral convolution. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '93, pages 263–270, New York, NY, USA, 1993. ACM.

[FT87] Michael L. Fredman and Robert E. Tarjan. Fibonacci heaps and their uses in improved network optimization algorithms. *J. ACM*, 34(3):596–615, July 1987.

[GJR$^+$12] M.A. Giorgetta, J. Jungclaus, C.H. Reick, S. Legutke, V. Brovkin, T. Crueger, M. Esch, K. Fieg, K. Glushak, V. Gayler, H. Haak, H.-D. Hollweg, T. Ilyina, S. Kinne, L. Kornblueh, D. Matei, T. Mauritsen, U. Mikolajewicz, W. Mueller, D. Notz, T. Raddatz, S. Rast, R. Redler, E. Roeckner, H. Schmidt, R. Schnur, J. Segschneider, K. D. Six, M. Stockhause, J. Wegner, H. Widmann, K.-H. Wieners, M. Claussen, J. Marotzke, and B. Stevens. Climate change from 1850 to 2100 in MPI-ESM simulations for the Coupled Model Intercomparison Project 5. submitted 2012.

[Gre92] John M. Greene. Locating three-dimensional roots by a bisection method. *Journal of Computational Physics*, 98(2):194 – 198, 1992.

[HH89] J. Helman and L. Hesselink. Representation and display of vector field topology in fluid flow data sets. *Computer*, 22(8):27 –36, aug. 1989.

[HH91] J.L. Helman and L. Hesselink. Visualizing vector field topology in fluid flows. *Computer Graphics and Applications, IEEE*, 11(3):36 –46, may 1991.

[Hod94] K. I. Hodges. A General Method for Tracking Analysis and Its Application to Meteorological Data. *Monthly Weather Review*, 122, 1994.

[JFH$^+$12] J.H. Jungclaus, N. Fischer, H. Haak, K. Lohmann, J. Marotzke, D. Matei, U. Mikolajewicz, D. Notz, and J.S. von Storch. Characteristics of the ocean simulations in MPIOM, the ocean component of the MPI Earth System Model. *Journal of Advances of Modeling Earth Systems*, 2012.

[JSSJ89] Johnny A. Johannessen, E. Svendsen, Stein Sandven, and Ola M. Johannessen. Three dimensional structure of mesoscale eddies in the norwegian coastal current. *Journal of Physical Oceanography*, 19(1):3–19, 1989.

[KHNH12] Jens Kasten, Ingrid Hotz, Bernd Noack, and Hans-Christian Hege. Vortex Merge Graphs in Two-dimensional Unsteady Flow Fields. pages 1–5, 2012.

[Lug79] H. J. Lugt. The dilemma of defining a vortex. In *Recent Development in Theoretical and Experimental Fluid Mechanics*, pages 309–321. Springer, 1979.

[MHJ$^+$03] S.J. Marsland, H. Haak, J.H. Jungclaus, M. Latif, and F. Röske. The Max-Planck-Institute global ocean/sea ice model with orthogonal curvilinear coordinates. *Ocean Modelling*, 5(2):91 – 127, 2003.

[Por97] Luis M. Portela. *Identification and Characterization of Vortices in the Turbulent Boundary Layer. Volume I.* PhD thesis, Stanford University, November 1997.

[PPF$^+$10] Armin Pobitzer, Ronald Peikert, Raphael Fuchs, Benjamin Schindler, Alexander Kuhn, Holger Theisel, Krešimir Matković, and Helwig Hauser1. On the Way Towards Topology-Based Visualization of Unsteady Flow – the State of the Art. In Helwig Hauser and Erik Reinhard, editors, *State of the Art Reports*, pages 137–154. Eurographics Association, May 2010.

[Pri57] R. C. Prim. Shortest connection networks and some generalizations. *Bell System Technology Journal*, 36:1389–1401, 1957.

[Rob91] S K Robinson. Coherent motions in the turbulent boundary layer. *Annual Review of Fluid Mechanics*, 23(1):601–639, 1991.

[Sad10] Filip Sadlo. Computational visualization of physics and topology in unsteady flow. 2010.

[Sah09] Jan Sahner. *Extraction of Vortex Structures in 3D Flow Fields*. PhD thesis, 2009.

[Sin84] R. W. Sinnott. Virtues of the Haversine. *Sky and Telescope*, 68(2):159+, 1984.

[SLM05] S. C. Shadden, F. Lekien, and J. E. Marsden. Definition and properties of Lagrangian coherent structures from finite-time Lyapunov exponents in two-dimensional aperiodic flows. *Physica D*, 212(3-4):271–304, 2005.

[SZH97] D. Stalling, M. Zockler, and H.-C. Hege. Fast display of illuminated field lines. *Visualization and Computer Graphics, IEEE Transactions on*, 3(2):118 –128, apr-jun 1997.

[TS03] H. Theisel and H.-P. Seidel. Feature flow fields. In *Proceedings of the symposium on Data visualisation 2003*, VISSYM '03, pages 141–148, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.

[TWHS04] H. Theisel, T. Weinkauf, H.-C. Hege, and H.-P. Seidel. Grid-independent detection of closed stream lines in 2d vector fields. In *Proc. Vision, Modeling and Visualization (VMV) 2004*, pages 421–428, Stanford, USA, 15.-18. November 2004.

[vSEF$^+$12] J.-S. von Storch, C. Eden, I. Fast, H. Haak, D. Hérnandez-Deckers, E. Maier-Reimer, J. Marotzke, and D. Stammer. An Estimate of Lorenz Energy Cycle for the World Ocean Based on the STORM/NCEP Simulation. *Journal of Physical Oceanography*, 2012.

[Wei08] Tino Weinkauf. *Extraction of topological structures in 2D and 3D vector fields*. PhD thesis, University Magdeburg, 2008.

[WS01] T. Wischgoll and G. Scheuermann. Detection and visualization of closed streamlines in planar flows. *Visualization and Computer Graphics, IEEE Transactions on*, 7(2):165 –172, apr-jun 2001.

[WSH01] Thomas Wischgoll, Gerik Scheuermann, and Hans Hagen. Tracking closed streamlines in time-dependent planar flows. In *In Proceedings of the Vision Modeling and Visualization Conference 2001 (VMV 01*, pages 447–454, 2001.

[WTGP11] Tino Weinkauf, Holger Theisel, Allen Van Gelder, and Alex Pang. Stable feature flow fields. *IEEE Transactions on Visualization and Computer Graphics*, 17(6):770–780, 2011.

List of Figures

List of Tables

A. Commands for Precomputation

CDO (climate data operators) is a freely available collection of operators to modify climate model data. Some computation steps were done with the following operators.

A.1. Choosing the Area of Interest

```
cdo selindexbox,lon1,lon2,lat1,lat2 infile outfile
```

The longitudes and latitudes are chosen with ncview.

Configuration	lon1	lon2	lat1	lat2
MR	610	725	256	336
STORM	2711	3283	1090	1464

Table A.1. Coordinates of chosen areas

A.2. Variables

Lots of variables are available from the data of MPIOM simulations. There are even more than listed below (most of the non mentioned can be calculated from that in the list):

- lon, lat, depth - the coordinates of the grid cells
- time - hours, relative to the start date of each data set
- zo - sea surface height above sea level
- tho - sea water potential temperature
- uko - sea water x velocity
- vke - sea water y velocity
- sao - sea water salinity
- po - sea water pressure
- rhoo - sea water density
- sst - sea surface temperature
- sss - sea surface salinity
- sictho - sea ice thickness
- sicomo - sea ice area fraction
- bottom_pressure - sea water pressure at sea floor
- rhopoto - sea water potential density

- wmo - upward ocean mass transport
- umo - ocean mass x transport
- vmo - ocean mass y transport
- psitro - ocean barotropic mass streamfunction
- eminpo - water flux correction
- pem - water flux into ocean
- avo - sea water vertical viscosity
- dvo - sea water vertical diffusivity
- transix - sea ice x transport
- transiy - sea ice y transport
- sicsno - surface snow thickness where sea ice
- amld - daily maxium of ocean mixed layer thickness
- zmld - ocean mixed layer thickness defined by sigma t
- thkcello - cell thickness
- strairx - zonal wind stress on ice
- strairy - meridional wind stress on ice
- strocx - zonal ice stress on ocean
- strocy - meridional ice stress on ocean
- ww0 - square of upward sea water velocity

Only coordinates and velocity (in x and y direction) are used in this work. As mentioned the salinity and the temperature could also add some new information and improve the quality. So every data set with these four values is compatible with the algorithms.

```
cdo selvar,uko,vke,sao,tho infile outfile
```

A.3. Computing Velocity Average over Depth

Computing the weighted average needs some more steps. First a netcdf file with weights for the whole grid has to be computed. Afterwards it is multiplied with the data and summed up vertically.

```
cdo sellevidx,8/23 -selindexbox,150,150,150,150 -selvar,ddpo
infile_weight weights.nc
```

ddpo is the layer thickness which can be found in description files of the models. The `selindexbox` command selects an arbitrary grid point. The result is the same as for all other points, because the layer thickness is constant. `sellevidx` selects which layers are used for the averaging. The resulting file is a list with the layer widths.

```
cdo enlarge,infile -div weights.nc -vertsum weights.nc
weights2.nc
```

The list of weights is normalized to one by the division through the sum. The result is repeated over the whole grid size (longitude, latitude directions) of the data file (with velocity, salinity, ...).

```
cdo vertsum -mul -sellevidx,8/23 infile weight2.nc outfile
```

This last step selects the same layers from the data file as selected from the weight file (with ddpo). These layers are weighted and summed up with the weight2.nc file from the previous step.

A.4. Selecting a single layer

Instead of averaging only a single layer (level) can be used.

```
cdo sellevidx,10 infile outfile
```

B. Time Series Evaluation

The test data set is the horizontal velocity (u,v) at a depth of $75m$. The spatial resolution is that of the MPI-ESM-STORM (0.1°). To reduce visual clutter only the extreme points of the first time slice are used for seeding the streamlines and for the extraction of eddy paths.

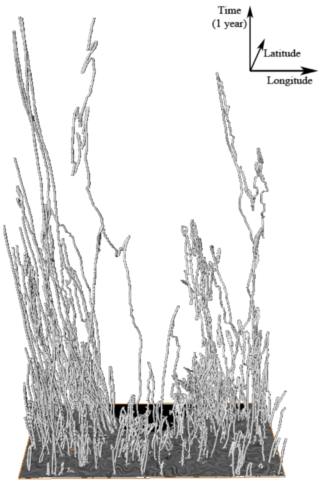

Figure B.1. FFF - daily resolution

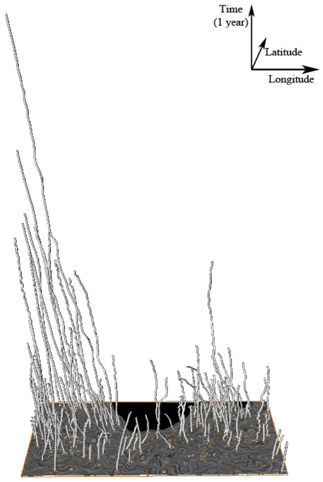

Figure B.2. SFFF - daily resolution

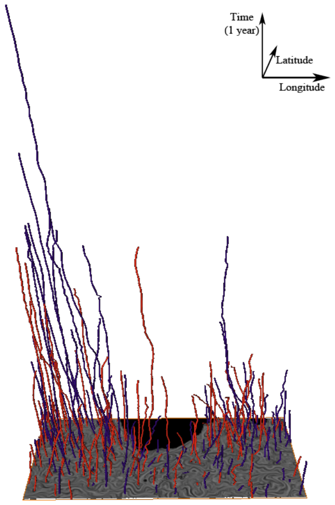

Figure B.3. Biased Approach - daily resolution

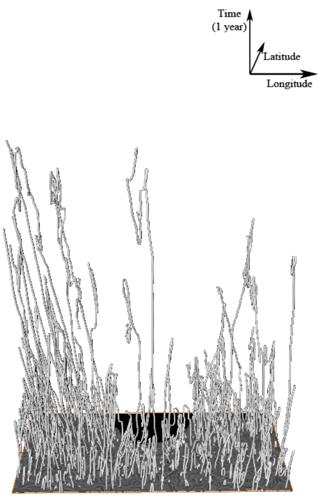

Figure B.4. FFF - 5-day resolution

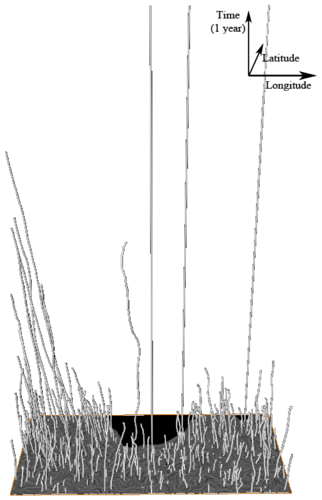

Figure B.5. SFFF - 5-day resolution

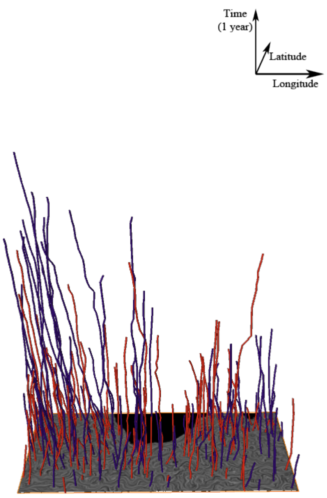

Figure B.6. Biased Approach - 5-day resolution

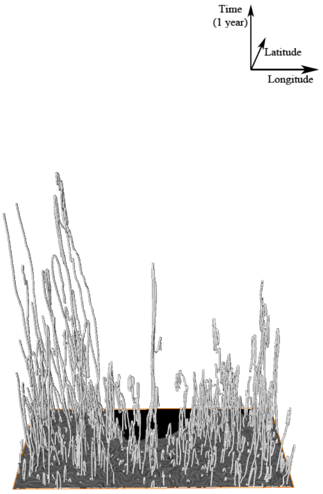

Figure B.7. FFF - 10-day resolution

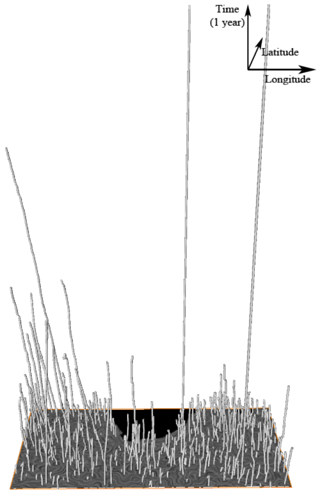

Figure B.8. SFFF - 10-day resolution

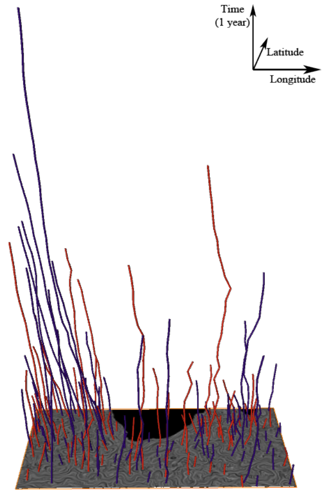

Figure B.9. Biased Approach - 10-day resolution

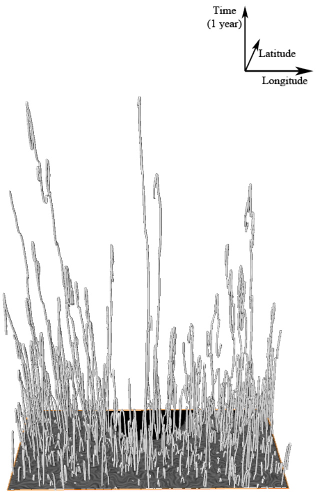

Figure B.10. FFF - 20-day resolution

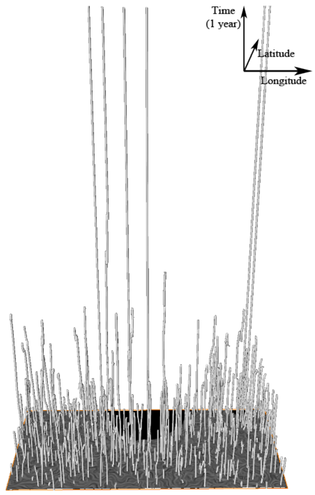

Figure B.11. SFFF - 20-day resolution

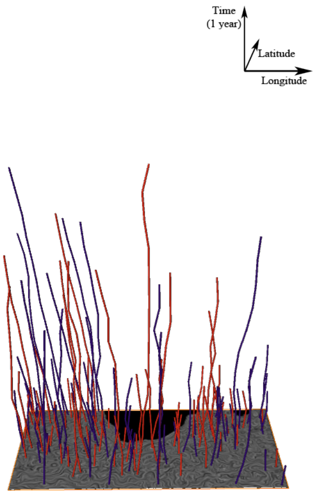

Figure B.12. Biased Approach - 20-day resolution

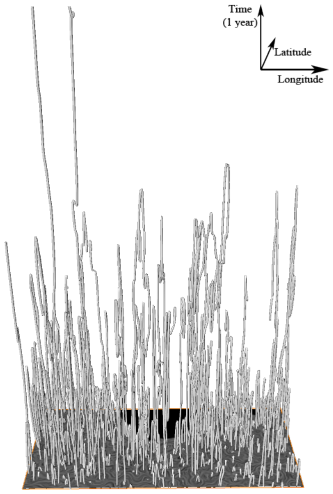

Figure B.13. FFF - 30-day resolution

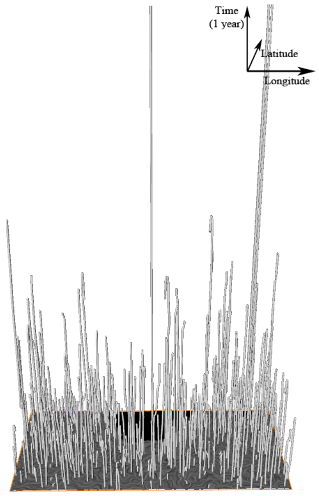

Figure B.14. SFFF - 30-day resolution

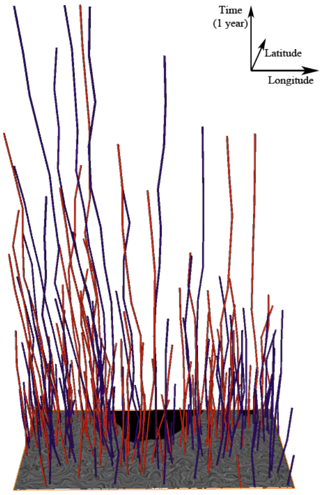

Figure B.15. Biased Approach - 30-day resolution

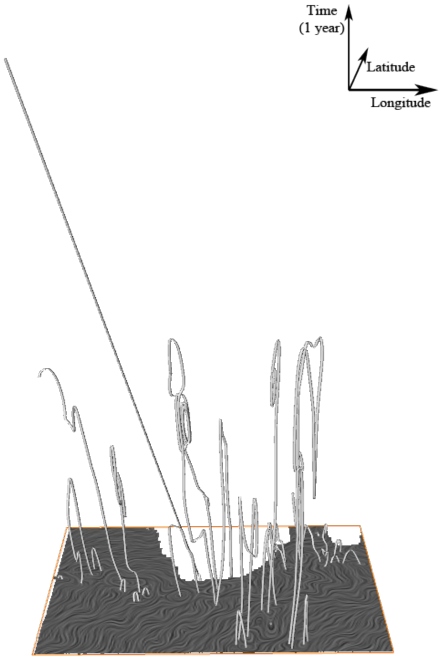

Figure B.16. FFF - 30-day, MR

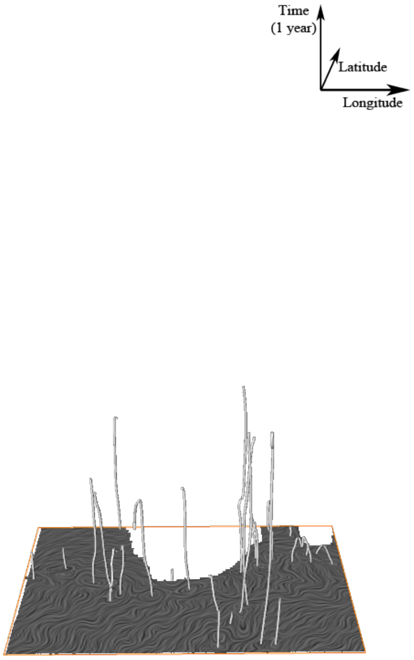

Figure B.17. SFFF - 30-day, MR

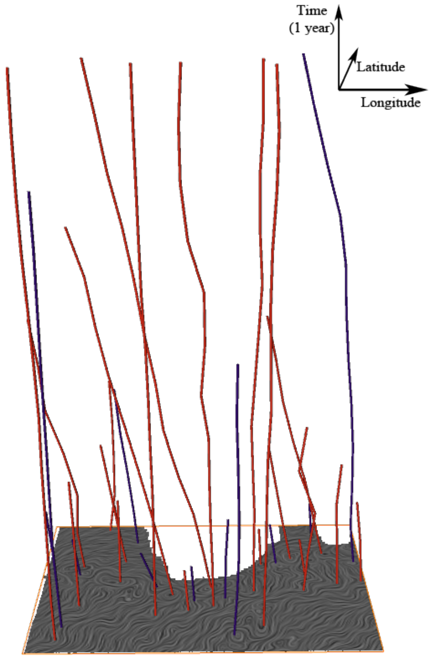

Figure B.18. Biased Approach - 30-day, MR

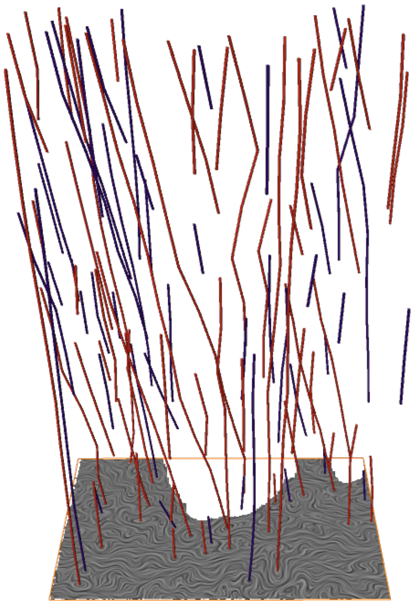

Figure B.19. Biased Approach - full path set - 30-day, MR

Selbstständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Bachelorarbeit selbstständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe. Die aus fremden Quellen direkt oder indirekt übernommenen Stellen sind als solche kenntlich gemacht.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

Magdeburg, 11.02.2013 ....
Johannes Jendersie