pyParaOcean: A System for Visual Analysis of Ocean Data

Toshit Jain[1], Varun Singh[1], Vijay Kumar Boda[1], Upkar Singh[1], Ingrid Hotz[2,1], P. N. Vinayachandran[1], Vijay Natarajan[1]

[1]Indian Institute of Science Bangalore, India
[2]Department of Science and Technology (ITN), Linköping University, Norrköping, Sweden

Abstract

Visual analysis is well adopted within the field of oceanography for the analysis of model simulations, detection of different phenomena and events, and tracking of dynamic processes. With increasing data sizes and the availability of multivariate dynamic data, there is a growing need for scalable and extensible tools for visualization and interactive exploration. We describe pyParaOcean, a visualization system that supports several tasks routinely used in the visual analysis of ocean data. The system is available as a plugin to Paraview and is hence able to leverage its distributed computing capabilities and its rich set of generic analysis and visualization functionalities. pyParaOcean provides modules to support different visual analysis tasks specific to ocean data, such as eddy identification and salinity movement tracking. These modules are available as Paraview filters and this seamless integration results in a system that is easy to install and use. A case study on the Bay of Bengal illustrates the utility of the system for the study of ocean phenomena and processes.

1. Introduction

Understanding ocean data is paramount to predicting extreme weather events such as hurricanes and tsunamis, better understanding of planetary scale processes like global warming, and sustainably managing and preserving ocean resources and its marine life. Visualizing ocean data is challenging due to the presence of multiple fields and parameters that change with time. Ocean currents are undeniably the biggest factor that maintain the heat balance of the ocean-atmosphere system and affect the movement of minerals and salt. Mesoscale eddies, that span from tens to hundreds of kilometers in diameter and have a lifespan that can range from days to months [RR10], are ubiquitous in the oceans. They play a big role in transporting heat and mass within the oceans [McW08]. They also have a big impact on the ecology of the ocean and on the biogeochemical processes [MJD*99, BNBD*07].

With the strides in collection and generation of ocean data [FD06, Ros89], there is a need for tools that support effective visualization of the data, and are scalable to keep pace with the ever expanding resolutions and sizes of ocean datasets.

1.1. Related work

Visualization in oceanography is a challenging area of research due to the rapidly increasing size of data, the heterogeneity and multivariate nature of the data, and the inherent complexity of the ocean phenomena. The use of general purpose analysis and visualization software such as Matlab, Tecplot, AVS, and Paraview is prevalent in the community. However, oceanographers often use tools developed specifically for ocean data, such as Ferret [Fer23], pyFerret [pyF23], and Copernicus MyOcean [myO23]. These specialized tools provide multiple functionalities and produce 2D views of the data.

A few software frameworks developed within the visualization community provide 2D and 3D data visualization capabilities. COVE [GSK*08] is a collaborative ocean visualization environment that supports interactive analysis of ocean models over the web. RedSeaAtlas [AGT*19] supports the selection of regions in a 2D map and provides exploratory views of winds, waves, tides, chlorophyll, etc. over the Red Sea. OceanPaths [NL15] is a multivariate data visualization tool that computes pathways tracing ocean currents and supports the plotting of different high-dimensional data along the pathways. This enables the study of correlations between different oceanographic features.

An oceanographer's analysis workflow includes a few common tasks [GSK*08] such as inspection of temperature and salinity distributions and vertical cross sections, compare recently measured salinity data against model data, inspect and analyze current vorticities and circulation based on flow data, and analyze extreme events. Isosurfaces and volume rendering are natural choices for visualization of 3D temperature and salinity distributions [DAN12, PBI04]. However, visualization of the dynamically changing distributions is a challenge. VAPOR [LJP*19] is one of the few tools that provides efficient 3D visualization for oceanography and atmospheric science applications. The VAPOR data collection (VDC) data model supports interactive visual analysis of large data sizes on modern GPUs and commodity hardware.

Xie et al. [XLWD19] and Afzal et al. [AHG*19] present surveys of visual analysis methods and tools developed for ocean data. Xie et al. classify the visual analysis tasks into four categories, namely

study of different environmental variables, ocean phenomena identification and tracking, discovery of patterns and correlations, and visualization of ensembles and uncertainty. Further, they identify different opportunities and unexplored areas for future work including efficient and scalable methods for data processing and management, identification of features at multiple scales, and immersive platforms. While we recognize the availability of several methods for oceanography visualization, we note that they are often standalone solutions. We aim to leverage the extensive advancement in visualization technology as implemented in Paraview while providing functionalities and views that are specific to ocean data.

1.2. Contributions

We present pyParaOcean, a system for interactive exploration and visual analysis of ocean data. The system leverages the power of Paraview [AGL05] to enable scalable visualizations of data available from ocean models while supporting a multitude of tasks and functionalities that are specialized for oceanography. The visualization capabilities of pyParaOcean are available via a seamless integration into Paraview using plugins. Key features of the system include

- 3D field visualization to study salinity and temperature distribution with support to display and explore dynamically evolving isovolumes.
- Ocean current visualization with a support for different seeding strategies for streamlines and pathlines.
- Vertical section views and parallel coordinates plot that support interactive selection and slicing of the data.
- Identification and tracking of salinity movement via extraction of surface fronts.
- Visualization and tracking of eddy features.
- An extensible design that supports incorporation of new functionalities as *filters* in Paraview.

We present the results of an exploration of the Bay of Bengal, performed in collaboration with an oceanographer, as a case study to demonstrate the utility of the system.

2. Ocean data

Oceanographers typically deal with large multivariate spatio-temporal datasets – time-varying scalar or vector fields over a three dimensional region. The data is generated using simulations, satellite imagery, sensors on buoys, or in-situ physical observations. With strides in high performance computing, higher resolution sampling, and the increasing number of observables, the size of such datasets is rapidly increasing. Reanalysis datasets combine a numerical simulation model with observational inputs to furnish data that is spatio-temporally consistent. Ocean data contains strong temporal and spatial processes involving complex interactions between multi-scale entities [XLWD19]. It is analyzed on a variety of scales, from small-scale features such as eddies and fronts, to large-scale features such as ocean basins and circulation patterns.

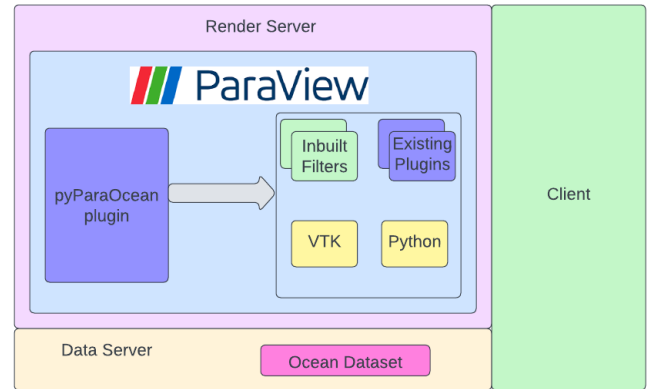All visualizations in this paper are generated using two datasets, the Red Sea and Bay of Bengal.

Figure 1: pyParaOcean system architecture. The plugin includes multiple specialized filters for visualizing ocean data that seamlessly integrate with the high performance capabilities of Paraview.

Red Sea: This dataset [TZG*17] was made available as a part of the IEEE SciVis 2020 contest. It is a 50 member ensemble of three-dimensional scalar and velocity fields. The data is regularly sampled on a $500 \times 500 \times 50$ grid over 60 time steps covering a whole month of simulation time. Ensembles are the outputs of the simulated models with different parameters and initial conditions, and they may vary significantly even with a small change in parameter values. The members are the forecasts from MITgcm setups configured for the 30°E - 50°E and 10°N - 30°N domain that spans the entire Red Sea. They are implemented in Cartesian coordinates with a horizontal resolution of $0.04° \times 0.04°$ (4 km) and 50 vertical layers, with a surface spacing of 4 m and a bottom spacing of 300 m. The dataset is available in the NetCDF format.

Bay of Bengal: This dataset is generated by a reanalysis product and available from the Nucleus for European Modelling of the Ocean (NEMO) repository [Mad08], with a daily resolution spanning the months of July-August 2020, a total of 62 time steps. The data is available in NetCDF format, with a $1/12°$ latitude-longitude resolution. Salinity measurements are available at 50 vertical levels, ranging from 1 m resolution near the surface to 450 m resolution towards the sea floor, including 22 samples in the upper 100 m. The Bay of Bengal, a geographical region confined by longitudes 75°E and 96°E and latitudes 5°S to 30°N, with depth up to 200 m, is extracted from this data.

3. pyParaOcean: Architecture

pyParaOcean is a plugin developed on top of Paraview to cater to the visualization needs of oceanographers, see Figure 1. Paraview [AGL05] is an open source visualization software. The visualization pipeline in Paraview is a data flow network of executable modules. A module is a functional unit in Paraview with zero or more input ports and zero or more output ports. A module in Paraview can perform one of three tasks: produce data (source), handle and process incoming data (filter), or render/produce image (sink). Paraview is a general purpose visualization tool and hence includes
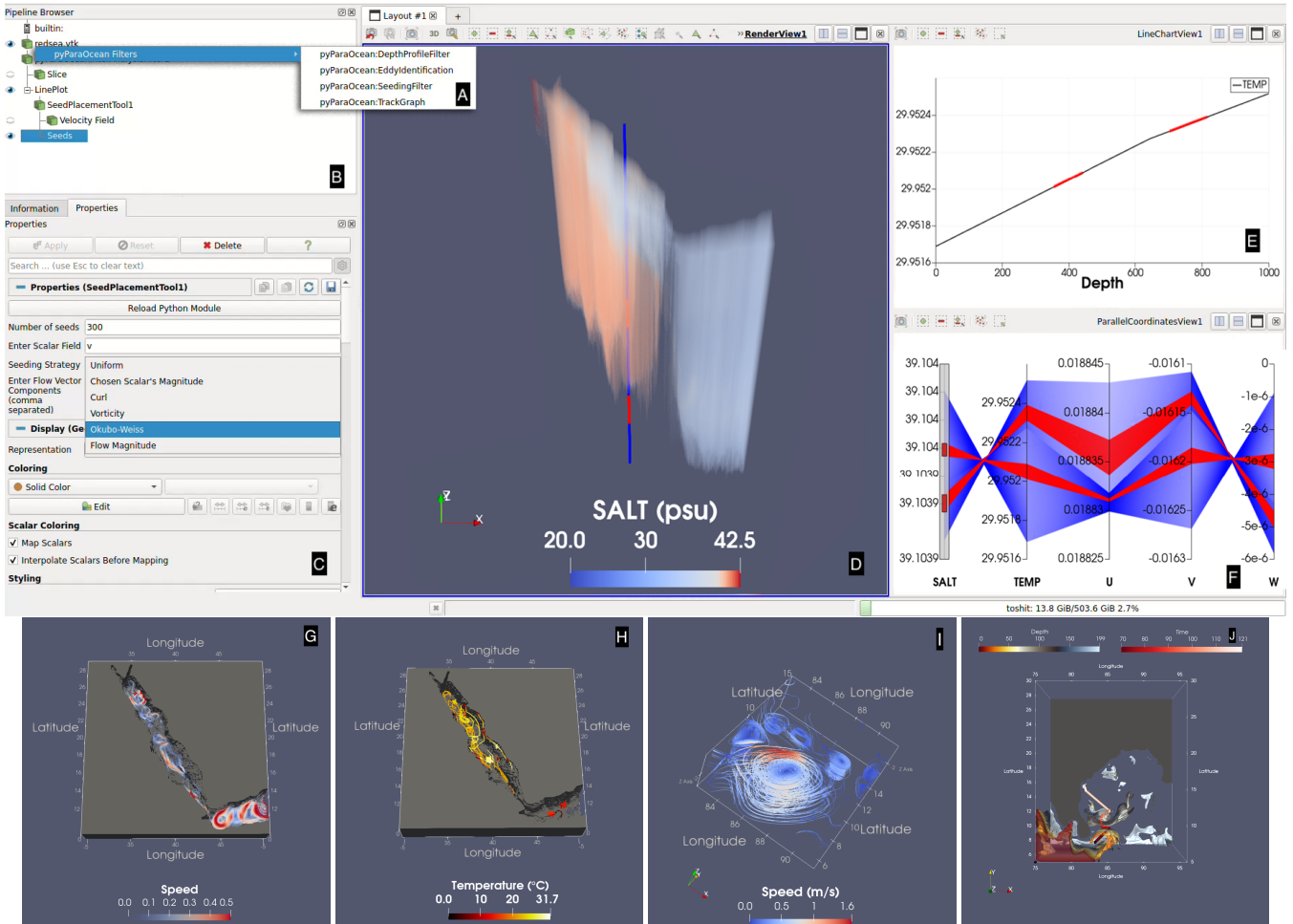
Figure 2: pyParaOcean functionality and user interface. (A) All pyParaOcean modules are implemented as Paraview filters. (B) Paraview pipeline browser shows the different datasets under study and the filters applied on them. (C) The seeding filter from pyParaOcean provides multiple options for tracing field lines. The figure illustrates the usage of various filters showcasing (D) salinity visualization in the Red Sea using volume rendering, (G) flow visualization in the Red Sea with streamlines, (H) interactively seeded pathlines in the Red Sea, (I) eddy detection and visualization in the Bay of Bengal, and (J) tracking high salinity water movement towards the east coast of India with a surface front track.

a large number of readers, data sources, and filters. The sheer number of available filters tends to get overwhelming and arduous to navigate, particularly for an application domain expert. Paraview also provides a mechanism for including new modules via plug-ins in scenarios where the available collection of modules does not meet the needs of the user.

Figure 1 shows an overview of the pyParaOcean architecture. The plugin is a collection of filters that provide visualization tools for interactive analysis of 3D ocean data and specialized features such as eddy computation and visualization, high salinity water visualization, and tracking of high salinity water masses. Ocean data is typically available as samples on a rectilinear grid in the NetCDF format. The plugin uses built-in VTK and Paraview libraries when appropriate, which are well maintained and supported, and hence

benefits from the availability of support from the user community. The plugin can be loaded in a few simple steps.

Paraview is designed as a three-tier client-server architecture, consisting of a data server, a render server, and the client. The data server is responsible for all data related tasks such as reading, writing, and filtering. The render server is responsible for rendering and the interaction and exploration is performed at the client. The client manages the creation, execution, and destruction of objects on the servers, but it does not contain the data. This architecture is useful for running applications in a parallel environment. The data and render server can be run on a headless server or a super-computer while the device with the end user acts as a client. When executed on the client without any connection to a remote server, Paraview seamlessly connects to a built-in server and provides all of its functionality. pyParaOcean is designed to leverage the paral-

lelization capabilities of Paraview, to scale with larger data sizes. pyParaOcean aims to provide a set of filters specific to the needs of an oceanographer that is extensive, accessible, and easy-to-use.

4. pyParaOcean: Functionalities

We now list and describe the various functions that are implemented in the pyParaOcean plugin and made available as Paraview filters. Figure 2 and the video in the supplementary material show the different filters in pyParaOcean and the user interface.

4.1. Isovolume visualization

Volume rendering is a natural choice for visualizing the 3D scalar fields in the ocean data because it provides a quick overview of the distribution (Figure 2(D)). Animation with a fixed transfer function allows visualization of the scalar field over time. The volume rendering filter within Paraview can be tuned to visualize subvolumes of interest by choosing a interval within the range of the scalar field. Specifically, an isovolume containing the mean salinity/temperature value within the spatial region of interest or an isovolume that captures high salinity water provide a good overview of the 3D field.

4.2. Seed placement and field lines

Field lines, including streamlines and pathlines, provide a good overview of a 3D vector field. pyParaOcean provides a filter that implements multiple seeding strategies for initiating the computation of streamlines and pathlines, and allows the user to choose one. Seeds generated using this filter are fed as input to the custom source streamline integrator or particle tracer in Paraview (Figure 2(G,H)).

Streamlines are a set of integral curves tangent to the velocity at every point in space. They portray instantaneous lines of flow that characterize important oceanographic phenomena such as eddies, currents, and filaments. Pathlines are tangential to the velocity as it evolves over time. A pathline describes the path a massless virtual particle would take beginning from the seed positioned at a particular timestep. Pathlines are useful for understanding transport, such as salinity advection and debris collection. They are more compute-intensive than streamlines.

The seeding filter controls the number of seeds and how the domain is sampled for seed placement, see Figure 2 (C)). Sampling can be (a) uniform, (b) weighted by flow speed, curl, vorticity, or the Okubo-Weiss criterion [Oku70], or (c) weighted by user-defined scalar fields that are computed earlier in the pipeline. A user may tune the line integration parameters and the sampling options to reduce visual clutter, focus the computation to regions of interest, maximize coverage of the domain, and highlight interesting flow features. For example, rendering wispy streamlines in regions of high vorticity produces closed loops around eddies with some temporal coherence across frames (Figure 7).

Additionally, each component of the vector field to which the seeding filter is applied can be specified as a separate scalar field. This makes it easier to perform other downstream operations like
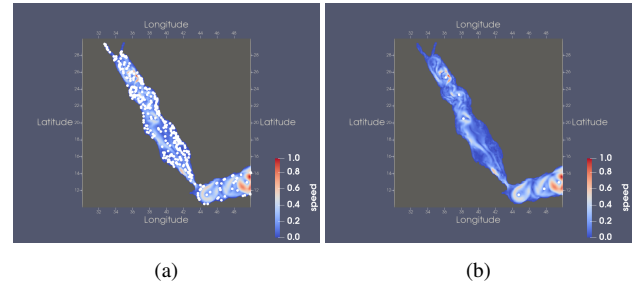

(a)   (b)

Figure 3: *Efficacy of the method used for eddy centre detection. (a) Local minima (white spheres) of the velocity magnitude are potential eddy centres. (b) Topological simplification removes noise and insignificant minima, and identifies vortex cores of mesoscale eddies in the Red Sea and Gulf of Aden.*

ignoring the vertical velocity component or adjusting the scale along each axis.

4.3. Interactive particle paths

This filter facilitates heat and mass transport queries on ocean data with an interactive seeding extension to the particle tracer from Paraview. It displays a linked parallel coordinates plot, where the user can brush to select ranges of scalars like temperature and salinity and hence restrict the seeding to isovolumes. Points sampled from these subvolumes serve as seeds for pathline computation (Figure 2(H)).

4.4. Depth profile view

This filter enables the user to inspect a vertical column of the ocean, specified by a longitude and latitude pair. It drops a "needle" into the ocean and samples points along this line at different values of depth (Figure 2(D)). It displays a linked parallel coordinates plot that gives a depth profile of all scalars sampled along the vertical column. A line plot view of the chosen scalar against depth (Figure 2(E)) is displayed. Optionally, the scalar field mapped to a vertical slice at the chosen longitude is shown in the volume render window. The user can select and highlight a subset of the points in the vertical column from the parallel coordinates plot (Figure 2(F)), and track them across time in all views. This is useful for studying vertical mass transport, especially upwelling or downwelling via Ekman transport [Sar13] in eddy centres, and to study the depression of isotherms indicating redistribution of heat [KNR*07]. Studying these changes is important to understand marine life as they drive phytoplankton bloom and nutrient transport.

4.5. Eddy identification and visualization

Several algorithms have been devised for reliable and automatic identification of eddies [AHG*19]. McWilliam [McW90] developed a 2D method using vorticity ω as a physical parameter, whose local minima and maxima locate the centres of potential eddies and the values of vorticity in the neighborhood relative to the centre helps determine the eddy boundary. Okubo [Oku70] uses a special
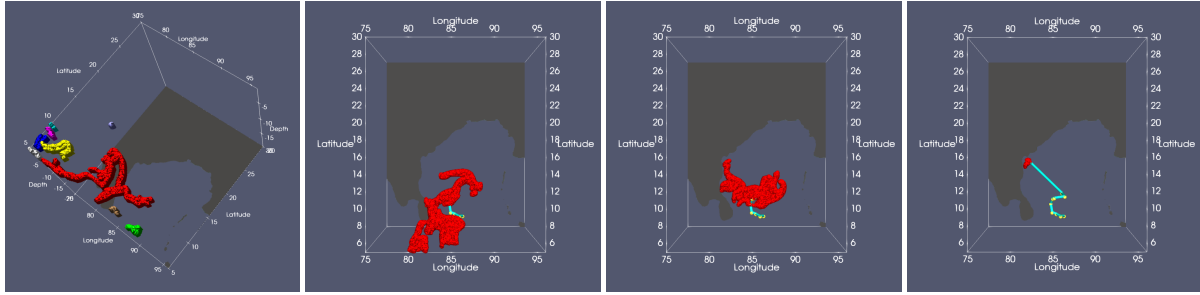
Figure 4: Visualizing movement of high salinity water via computation and tracking of surface fronts of high salinity isovolumes. (left) Surface fronts computed at one time step. (middle, right) One of the components of the surface front moves towards the east coast of India, near Visakhapatnam. The evolution of this surface front component is computed and visualized as a track.

Okubo-Weiss parameter based on shear and strain deformation and the vertical component of vorticity to measure rotation and hence identify potential eddies. A circularity criterion may be applied after the Okubo-Weiss criterion to improve the results [WHP*11]. Sea surface height and velocity profile have also been used for eddy detection [MAIS16]. The winding angle criterion, together with a streamline clustering, helps identify eddies in 3D [FFH21].

The eddy identification filter in pyParaOcean focuses on mesoscale eddies [AMM17] . It uses only the velocity field in individual time steps and does not compute any derived fields. This 3D detection scheme can be applied in parallel across timesteps and across depth slices since the vertical velocity is not used.

Flow speed of swirling fluid decreases radially inwards towards the centre of rotation. The filter inspects local minima of the flow speed to identify potential eddy centres. Vertical velocity is ignored to discount the motion of upwelling or downwelling in vortex cores, thus enhancing the corresponding flow minima. Noise and less significant minima are removed by applying topological simplification directed by the notion of persistence [TFL*17]. Next, the method employs an approximation of the winding angle criterion [FFH21] by checking if the streamline crosses into all four quadrants of an XY plane centred at the minimum [GEP04]. This method is more effective in regions with relatively stationary eddy centres like the Red Sea and Gulf of Aden. Figure 3 shows the set of potential eddy centres identified in the Red Sea using this filter.

Streamlines seeded near the core of an eddy form spirals or closed loops. The boundary of an eddy is determined using a binary search along the radial axes. The search helps locate the seed that is furthest from the eddy centre but results in a spiral or nearly closed loop streamline. The filter displays all streamlines originating near the detected vortex core and hence presents a 3D profile of the eddy (Figure 2(I)). It may be extended to support other eddy detection methods [MAIS16, FFH21] that may be selected via the interface.

4.6. Surface front tracking and salinity visualization

Oceanographers are often interested in water parcels that transport mass or heat. These are moving volumes of water with distinctive temperature or salinity characteristics. The surface front tracking filter computes connected components of the boundary of a scalar

Figure 5: Currents and eddies in the Bay of Bengal during the monsoon season, including the Summer Monsoon Current (SMC), the Sri Lanka Dome (SLD), and an anticyclonic eddy (AE).

field isovolume (called the surface front), tracks them over time, and generates a track graph that summarizes movement of all surface fronts. A subset of tracks extracted from this graph are rendered for visual analysis. Surface fronts have been shown to be a good representative of high salinity water masses [SDVN22]. It has been used to trace the path of the high salinity core (HSC) entering into the Bay of Bengal from the Arabian Sea (Figure 2(J)).

5. Case study: Bay of Bengal

The Summer Monsoon Current (SMC) is a prominent feature of Indian ocean circulation and the SMC flows around Sri Lanka to flow into the Bay of Bengal. We use pyParaOcean to study different phenomena in the Bay of Bengal, particularly during the monsoon.

Eddies. Figure 5 is a rough schematic of the major currents and eddies in the Bay during the monsoon season. A large anticyclonic eddy (AE) located to the right of the SMC and a cyclonic eddy known as the the Sri Lanka Dome (SLD) to its left [VY98] are regular features in this region during summer. The AE has a diameter
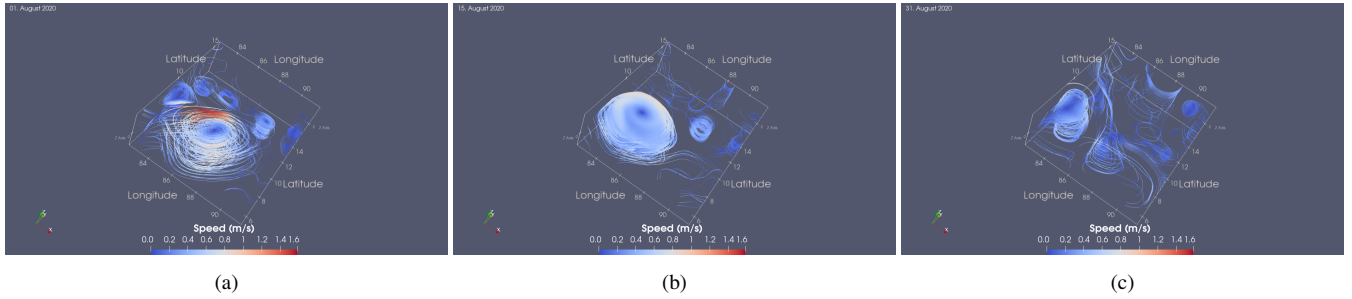
Figure 6: *Dissipation of a large anticylonic eddy in the Bay of Bengal through August 2020. Streamlines are seeded near detected vortex cores to show evolution of eddy profiles in 3D.*
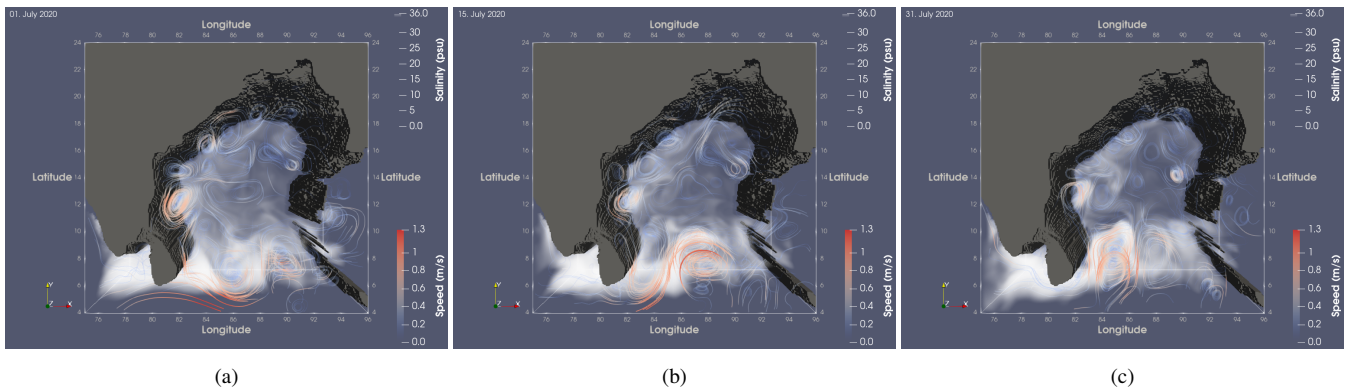

Figure 7: *The Bay of Bengal between July 1, 2020 and July 31, 2020. Visualization of the flow using streamlines with uniform seeding and (≥ 35 psu) salinity isovolume rendering. (a) July 1, 2020: The AE can be seen forming around 8°N and 90°E with the SMC streamlines visible from 78°E to about 86°E. (b) July 15, 2020: The AE, 8°N and 87°E, has matured into a circular shape and moves westward towards Sri Lanka. The (≥ 35 psu) isovolume shows recirculation of high salinity waters into they Bay by AE. (c) July 31, 2020: The AE, 7°N and 84°E, reaches the eastern coast of Sri Lanka where it will start to dissipate.*

of about 500 km, located to the southeast off the coast of Sri Lanka, and is characterized by intense downwelling inside owing to its anticyclonic circulation. [VY98] proposed that the AE is formed by the interaction of the SMC and the incoming Rossby waves from Sumatra. The timeline of appearance and disappearance of the AE was documented in later work [VCMN04]. The AE starts forming in June, develops into its circular shape in July, and weakens in August, as shown in Figure 6 and the accompanying video.

Salinity transport. pyParaOcean serves as an efficient tool to analyze the effects of AE on the Bay of Bengal. Streamlines and pathlines offer visualization of circulation associated with the AE and its movement in the ocean. The field lines may be overlaid on a volume rendering of a scalar to visualize the transport caused by the eddy. Figure 7 and the accompanying video show the streamlines overlaid on a salinity volume rendering at different time steps to show the role of the AE in transport of salt. The movement of high salinity water from the Arabian sea by the SMC into the Bay of Bengal and its recirculation by the AE is well captured in this representation. Tracking surface fronts of high salinity water and highlighting the long-lived tracks helps capture an overview of sig-

nificant salinity movement in the region. We observe a track that moves towards the coast of India, see Figure 4.

Downwelling. Figure 8 and the accompanying video show the use of the depth profile filter to visualize the depression of the 27° isotherm by the AE. The anticyclonic nature of the eddy causes a downwelling inside the eddy and pushes the relatively warmer water downward. The parallel coordinates view shows changes in temperature, salinity, and speed in the water column caused by the arrival of the eddy at the point of interest.

Experience and performance. This case study was conducted in collaboration with an oceanographer coauthor. Several observations on phenomena such as the SLD and the movement of high salinity water could be made using pyParaOcean. While our oceanographer collaborators typically use tools such as pyFerret for 2D analysis, they found the capability of pyParaOcean to be very useful. After this initial satisfying experience, we plan to work together on the study of higher resolution model output using pyParaOcean. The surface front tracking and eddy detection filters take a few minutes, while all other filters take 1-2 seconds or less. All the experiments were run on a workstation with an 8 core AMD
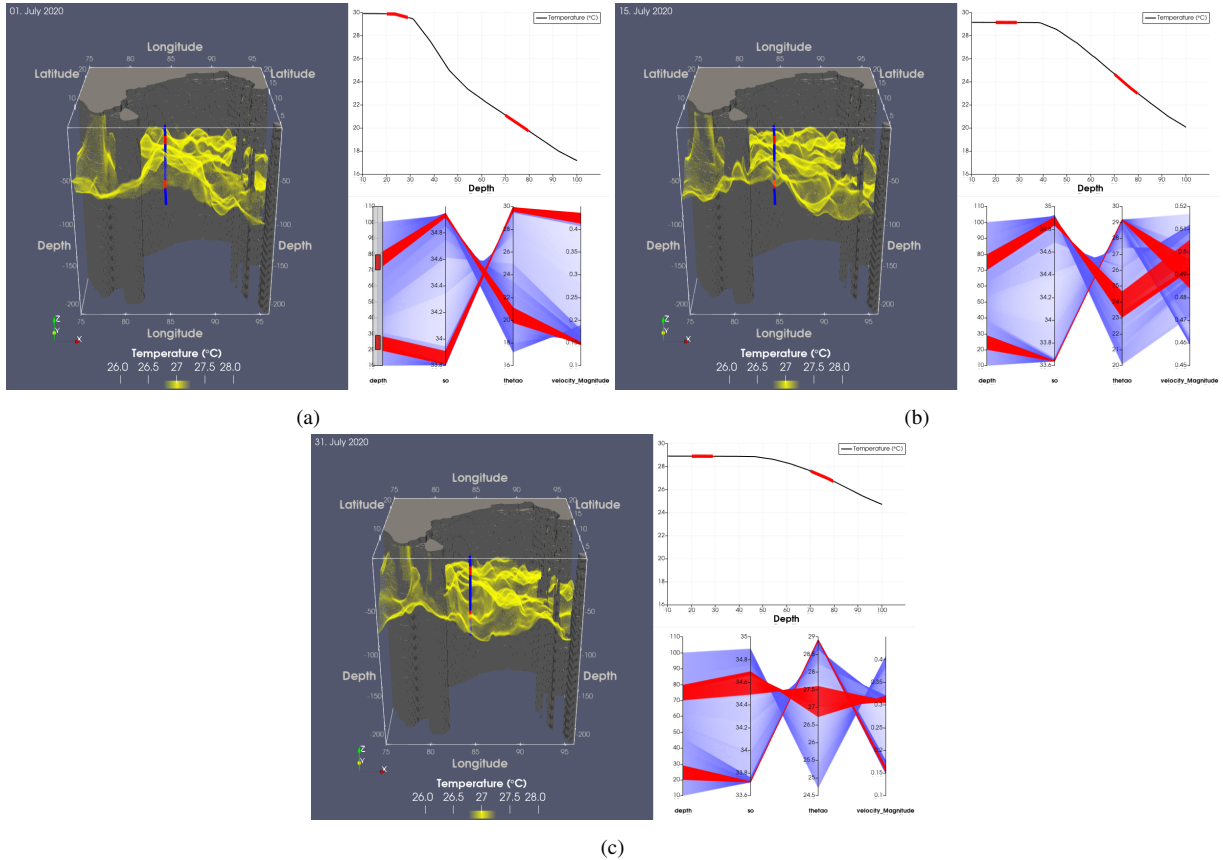
(a)

(b)

(c)

Figure 8: The depression of the 27° isotherm (yellow) by the anticylonic eddy in the Bay of Bengal. A needle is dropped at 7°N, 84°E and the depth profile shows the temperature drop. The interactive parallel coordinates plot is used to brush-select 10 m intervals at depths of 25 m and 85 m. (a) July 1, 2020: The downwelling of the AE can be seen around 8°N and 90°E, at the depth of 100 m. As it forms, the AE pushes the 27° isotherm down. (b) July 15, 2020: The AE, 8°N and 87°E, can be seen moving east with the depression of the isotherm and the depth profile of temperature begins to flatten near 29°C as the eddy moves closer to the needle. (c) July 31, 2020: The AE centre, 7°N and 85°E, is very close to the needle and the depression in the isotherm has moved all the way to near the east coast of Sri Lanka.

EPYC 7262 @ 3.2 GHz CPU with 512 GB main memory and NVIDIA RTX A4000 (16 GB) GPU. The surface front computation is parallelized using the python multiprocessing library but there is scope for further improvement in runtime. The eddy detection and visualization filter can also be optimized by parallelizing some of the computation. We plan to take this up in the future.

6. Conclusions

This paper described pyParaOcean, a system for visual exploration and analysis of 3D time-varying ocean data. pyParaOcean is designed as a plugin for Paraview and provides visualization and analysis functionalities that are specific to oceanography. Its integration into Paraview enables efficient computation and optional support for a larger set of general purpose filters for visual analysis. A case study on the Bay of Bengal, conducted together with an oceanographer collaborator, illustrates the use of pyParaOcean to explore eddies, ocean currents, and salinity movement in the Bay of Bengal. In future work, we will include support for additional filters for

computing statistics, temporal data analysis, and correlation studies. We also plan to improve the runtime performance and conduct scalability tests for higher resolution ocean models before releasing the plugin as an open source software.

References

[AGL05] AHRENS J., GEVECI B., LAW C.: Paraview: An end-user tool for large data visualization. *The visualization handbook 717* (2005). 2

[AGT*19] AFZAL S., GHANI S., TISSINGTON G., LANGODAN S.,

DASARI H. P., RAITSOS D. E., GITTINGS J. A., JAMIL T., SRINI-VASAN M., HOTEIT I.: RedSeaAtlas: A visual analytics tool for spatio-temporal multivariate data of the red sea. In *EnvirVis: Workshop on Visualization in Environmental Sciences (EnvirVis2019)* (2019), pp. 25–32. 1

[AHG*19] AFZAL S., HITTAWE M. M., GHANI S., JAMIL T., KNIO O., HADWIGER M., HOTEIT I.: The state of the art in visual analysis approaches for ocean and atmospheric datasets. *Computer Graphics Forum 38*, 3 (2019), 881–907. 1, 4

[AMM17] AMORES A., MELNICHENKO O., MAXIMENKO N.: Coherent mesoscale eddies in the North Atlantic subtropical gyre: 3-D structure and transport with application to the salinity maximum. *Journal of Geophysical Research: Oceans 122*, 1 (2017), 23–41. 5

[BNBD*07] BENITEZ-NELSON C. R., BIDIGARE R. R., DICKEY T. D., LANDRY M. R., LEONARD C. L., BROWN S. L., NENCIOLI F., RII Y. M., MAITI K., BECKER J. W., ET AL.: Mesoscale eddies drive increased silica export in the subtropical pacific ocean. *Science 316*, 5827 (2007), 1017–1021. 1

[DAN12] DINESHA V., ADABALA N., NATARAJAN V.: Uncertainty visualization using HDR volume rendering. *The Visual Computer 28* (2012), 265–278. 1

[FD06] FRASER S., DICKSON B.: Data mining geoscientific data sets using self organizing maps. *Mastering the Data Explosion in the Earth and Environmental Sciences, Extended Abstracts* (2006), 5–7. 1

[Fer23] Ferret. https://ferret.pmel.noaa.gov/Ferret/, 2023. [Online; accessed 28-April-2023]. 1

[FFH21] FRIEDERICI A., FALK M., HOTZ I.: A winding angle framework for tracking and exploring eddy transport in oceanic ensemble simulations. In *EnvirVis: Workshop on Visualization in Environmental Sciences (EnvirVis2021)* (2021). 5

[GEP04] GUO D., EVANGELINOS C., PATRIKALAKIS N.: Flow feature extraction in oceanographic visualization. In *Proceedings of Computer Graphics International Conference* (07 2004), pp. 162–173. doi:10.1109/CGI.2004.1309207. 5

[GSK*08] GROCHOW K., STOERMER M., KELLEY D., DELANEY J., LAZOWSKA E.: COVE: A visual environment for ocean observatory design. *Journal of Physics: Conference Series 125*, 1 (2008), 012092. 1

[KNR*07] KUMAR S. P., NUNCIO M., RAMAIAH N., SARDESAI S., NARVEKAR J., FERNANDES V., PAUL J. T.: Eddy-mediated biological productivity in the Bay of Bengal during fall and spring intermonsoons. *Deep Sea Research Part I: Oceanographic Research Papers 54*, 9 (2007), 1619–1640. 4

[LJP*19] LI S., JAROSZYNSKI S., PEARSE S., ORF L., CLYNE J.: Vapor: A visualization package tailored to analyze simulation data in earth system science. *Atmosphere 10*, 9 (2019), 488. 1

[Mad08] MADEC G.: *NEMO ocean engine*. Note du Pôle de modélisation, Institut Pierre-Simon Laplace (IPSL), France, No 27, ISSN No 1288-1619, 2008. 2

[MAIS16] MATSUOKA D., ARAKI F., INOUE Y., SASAKI H.: A new approach to ocean eddy detection, tracking, and event visualization–application to the northwest pacific ocean. *Procedia Computer Science 80* (2016), 1601–1611. 5

[McW90] MCWILLIAMS J. C.: The vortices of two-dimensional turbulence. *Journal of Fluid mechanics 219* (1990), 361–385. 4

[McW08] MCWILLIAMS J. C.: The nature and consequences of oceanic eddies. *Ocean modeling in an eddying regime 177* (2008), 5–15. 1

[MJD*99] MCNEIL J., JANNASCH H., DICKEY T., MCGILLICUDDY D., BRZEZINSKI M., SAKAMOTO C.: New chemical, bio-optical and physical observations of upper ocean response to the passage of a mesoscale eddy off bermuda. *Journal of Geophysical Research: Oceans 104*, C7 (1999), 15537–15548. 1

[myO23] Copernicus myOcean. https://marine.copernicus.eu/access-data/ocean-visualisation-tools, 2023. [Online; accessed 28-April-2023]. 1

[NL15] NOBRE C., LEX A.: OceanPaths: Visualizing multivariate oceanography data. In *Proceedings of the Eurographics Conference on Visualization (EuroVis 2015) - Short Papers* (2015), The Eurographics Association. doi:10.2312/eurovisshort.20151124. 1

[Oku70] OKUBO A.: Horizontal dispersion of floatable particles in the vicinity of velocity singularities such as convergences. *Deep sea research and oceanographic abstracts 17*, 3 (1970), 445–454. 4

[PBI04] PARK S., BAJAJ C., IHM I.: Visualization of very large oceanography time-varying volume datasets. In *Computational Science-ICCS 2004: 4th International Conference, Kraków, Poland, June 6-9, 2004, Proceedings, Part II 4* (2004), Springer, pp. 419–426. 1

[pyF23] pyferret. https://ferret.pmel.noaa.gov/Ferret/, 2023. [Online; accessed 28-April-2023]. 1

[Ros89] ROSENBLUM L. J.: Visualizing oceanographic data. *IEEE computer graphics and applications 9*, 3 (1989), 14–19. 1

[RR10] ROBINSON I. S., ROBINSON I. S.: Mesoscale ocean features: Eddies. *Discovering the Ocean from Space: The unique applications of satellite oceanography* (2010), 69–114. 1

[Sar13] SARMIENTO J. L.: Ocean biogeochemical dynamics. In *Ocean Biogeochemical Dynamics*. Princeton university press, 2013. 4

[SDVN22] SINGH U., DHIPU T. M., VINAYACHANDRAN P. N., NATARAJAN V.: Front and skeleton features based methods for tracking salinity propagation in the ocean. *Computers & Geosciences 159* (2022), 104993. doi:https://doi.org/10.1016/j.cageo.2021.104993. 5

[TFL*17] TIERNY J., FAVELIER G., LEVINE J. A., GUEUNET C., MICHAUX M.: The Topology Toolkit. *IEEE Transactions on Visualization and Computer Graphics 24*, 1 (2017), 832–842. 5

[TZG*17] TOYE H., ZHAN P., GOPALAKRISHNAN G., KARTADIKARIA A. R., HUANG H., KNIO O., HOTEIT I.: Ensemble data assimilation in the Red Sea: sensitivity to ensemble selection and atmospheric forcing. *Ocean Dynamics 67*, 7 (Jul 2017), 915–933. doi:10.1007/s10236-017-1064-1. 2

[VCMN04] VINAYACHANDRAN P. N., CHAUHAN P., MOHAN M., NAYAK S.: Biological response of the sea around Sri Lanka to summer monsoon. *Geophysical Research Letters 31*, 1 (2004). 6

[VY98] VINAYACHANDRAN P. N., YAMAGATA T.: Monsoon response of the sea around Sri Lanka: generation of thermal domes and anticyclonic vortices. *Journal of Physical Oceanography 28*, 10 (1998), 1946–1960. 5, 6

[WHP*11] WILLIAMS S., HECHT M., PETERSEN M., STRELITZ R., MALTRUD M., AHRENS J., HLAWITSCHKA M., HAMANN B.: Visualization and analysis of eddies in a global ocean simulation. *Computer graphics forum 30*, 3 (2011), 991–1000. 5

[XLWD19] XIE C., LI M., WANG H., DONG J.: A survey on visual analysis of ocean data. *Visual Informatics 3*, 3 (2019), 113–128. 1, 2