EuroVis Workshop on Visual Analytics (2021)
J. Bernard and K. Vrotsou (Editors)

Multi-resolution analysis for vector plots of time series data

Bao Nguyen[1], Rattikorn Hewett[1], and Tommy Dang[1]
[1]Department of Computer Science, Texas Tech University

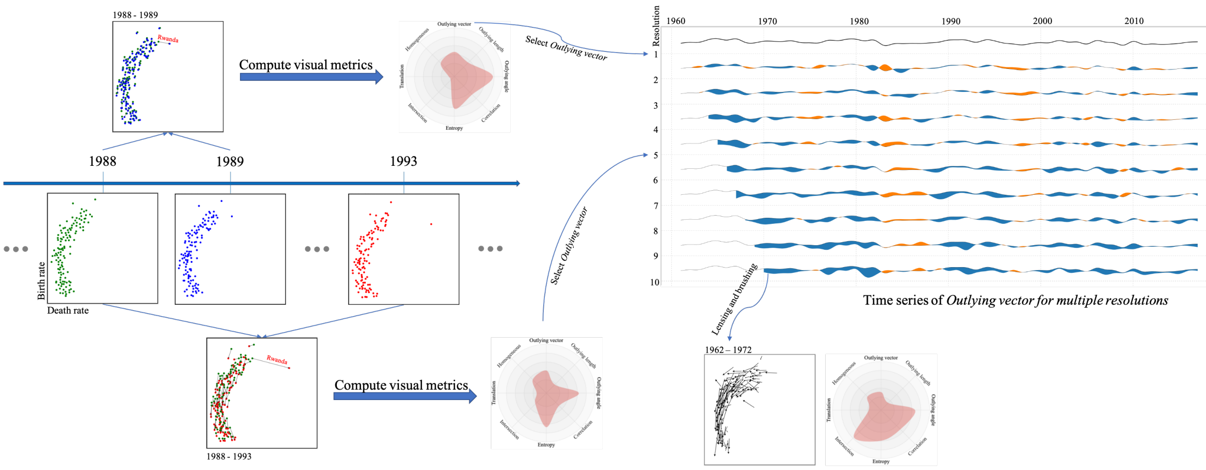

Figure 1: *Schematic overview: We build vector plots from the data, before computing their visual metrics. Next, we build time series of the selected metric and visualize the metric series for multiple resolutions by bivariate area charts. By interacting with these charts, users can investigate vector plots of interest using lensing and brushing.*

Abstract

Vector plots can directly visualize both temporal variation and spatial distribution, so it is interesting to use this type of plot for displaying multivariate time series. However, vector plots cannot reveal global temporal information. This paper introduces an interactive visualization that allows comparisons between different resolutions for easing this limit. The proposed approach is applied to two real data to demonstrate its benefits and potential.

1. Introduction

Visualization of multivariate time series has attracted much attention from researchers in recent years due to the rise of this type of data in practical applications. There are various approaches from dimensionality reduction to the extensions of regular plots for non-temporal multivariate data. While the former can provide an abstract visualization in a single view, the latter offers insights into data without missing any details. However, the latter usually generates a huge number of plots, so it is impossible to inspect all of them manually. Instead, some approaches help visualize them effectively. For example, scatterplots can visualize pairwise relationships in the data, but most data requires many scatterplots to display [MTW*20, DW14]. One straightforward extension of scatterplots to multivariate temporal data is the animation of these plots [NNDH20]. This method uses a scatterplot matrix (SPLOM) to display all pairwise relations at a time step and animate them to capture temporal behaviors. The animation requires users to memorize data point distributions in scatterplots to gain their evolutions. This paper aims to reduce user burden and make the analysis easier for users by using vector plots [BL19]. A vector plot combines two scatterplots at two separate time steps, called t_1 and t_2, and replaces data points with vectors. A vector connects a data point from its position at t_1 to its place at t_2; thus, the vector plots can present the evolution of distribution directly [WSSR20]. However, this plot only focuses on the variation over two separate time steps rather than temporal evolution over the whole period [NHD21]. Figure 1

DOI: 10.2312/eurova.20211099

shows an example for illustrating the use of vector plots in time series visualization. There is one scatterplot of *Death rate* vs. *Birth rate* at each year, and each data point represents a country. There are two vector plots in the Figure, one with $t_1 = 1988$ and $t_2 = 1989$, and another with $t_1 = 1988$ and $t_2 = 1993$. We call $\Delta t = t_2 - t_1$ a resolution because each Δt only provides a particular level of details of the evolution. The lack of global evolution limits the usability of this type of plot in time series visualization. Therefore, we propose a visual analytics approach to overcome this problem.

Our contribution is two-fold:

- We propose an interactive visualization that allows comparisons between different resolutions, so users can choose a resolution of interest for analyzing the data.
- We demonstrate the effectiveness of the proposed visualization by applying it to two distinct datasets.

The following section reviews the literature and clarifies the contribution of this paper. Section 3 describes the visualization design, while section 4 applies the proposed approach to two data sets in different domains. Finally, section 5 discusses the advantages and disadvantages of the method, and section 6 summarizes the paper.

2. Related work

There are several approaches to visualize multivariate time series. One popular method is the application of dimensionality reduction. Temporal Multidimensional Scaling (TMDS) [JFSK15] groups entries (each of them holds multiple dimensions) into sliding windows of appropriate size and overlap accordingly over time. Then, each sliding window is projected to a one-dimensional axis to display along with time. This method can depict distinct visual patterns for many network attacks. Another work, namely TimeCluster, also uses dimensionality reduction techniques for sliding windows [AJXW19]. However, the projection is two dimensions, and points are connected chronologically by line. It is similar to a connected scatterplot. This method can highlight visual patterns of motif and anomalies in long time series due to the clustering ability of dimensionality reduction techniques. Besides, numerical metrics for visual features can also allow applications of dimensionality reduction in time series analysis [NHD20b].

Although dimensionality reduction can visualize data's structure, it provides an abstraction view and may lose some details. Another approach is to integrate time into popular plots representing multivariate non-temporal data, such as parallel-coordinates or scatterplots. A straightforward method is to animate them to illustrate how they evolve [TK07, BS04]. However, the animation requires many efforts in memorizing visual patterns in the plots to capture valuable information. We can ease this inconvenience by some other methods, such as using Scagnostics as a signal to quickly focus on time steps of interest [NNDH20] in the case of SPLOM animation. Besides animation, some works try to extend the third axis for temporal direction [WLG97, GRPF16], while others use numerical metrics to capture behaviors of the plots to reduce the dimensions and complexity [DAW12, DKG12]. Beyond popular displays, some works also investigate other presentations to visualize the multivariate time series. For example, Congnostics [NHD20a] studies the use of connected scatterplots to visualize the dynamic pairwise relationship. Or, TimeWheel and MultiComb [TAS04] modify parallel-coordinates to include temporal information into the visualization.

This paper investigates vector plots, in which vectors indicate the temporal changes of data points. Vector plots are not novel in the scientific literature, as many documents in different domains use them to visualize dynamic data [BL19, WSSR20]. NetScatter [NHD21] suggests utilizing this type of plot in time series analysis and proposes a list of visual metrics for some visual features of the plots. Besides, they also introduce binned aggregation for easing clutter issues or some visualization methods for gaining information of interest from all vector plots. Because we use their visual metrics in this work, we will briefly discuss each metric in one aspect: corresponding visual features that the metric scores. How the metrics are computed and the formulas can be found in their paper [NHD21].

- *Outlying vector, Outlying length, Outlying angle*: Outliers can contain crucial information in the data, so it is essential to detect their existence. In vector plots, a vector may be abnormal due to its ridiculous position, extremely long length, or distinct direction. The metric *Outlying vector* takes into account all components of a vector, such as its location, its length, or direction. In contrast, the metric *Outlying length* only focuses on the vectors' length, and *Outlying angle* only considers vectors' direction. A high score of these metrics implies the existence of outliers on the plot, while a low score implies the opposite.
- *Correlation*: scores the local dependency between two variables over two time steps. A high score of this metric indicates most vectors have similar directions or trends, while a low score means there is no dominant trend for the vectors.
- *Entropy*: This metric applies the permutation entropy [BP02] to score the complexity and association in directions of vectors on vector plots. A high score means each vector may have a distinct direction, while a low score implies the association between the vectors.
- *Intersection*: scores how many crossings between vectors on the vector plots. The more intersections between vectors, the more jumbled distribution they form.
- *Translation*: scores amplitude of the collective evolution. In other words, this metric detects whether most data points have similar changes from t_1 to t_2 or not. The difference between this metric and the *Correlation* is that *Translation* tends to capture the amplitude of the changes. Or a high score implies significant increases/decreases in most data points for the pair of variables.
- *Homogeneous*: measures how similar vectors are in all components: location, length, and direction.

Because the vector plot cannot display the global temporal evolution, this paper introduces a visualization approach that allows comparisons between different resolutions, and then, users have hints about the resolution with interesting information. The idea is similar to the heterogeneity-based guidance [LMS*12]. However, instead of stacking and visualizing heterogeneity bands with the data, we use visual metrics to directly show the difference in other resolutions. Although we utilize the visual metrics mentioned above, the proposed approach is applicable for any collections of metrics.

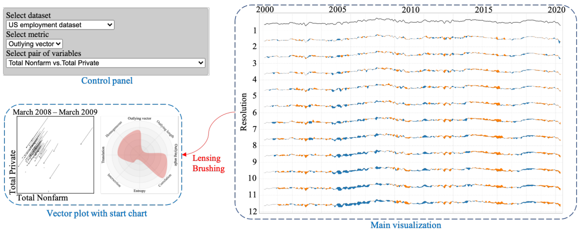

Figure 2: *Main interface of our visualization with three components.*

3. Visualization design

Main interface: The main interface of the design has three main parts: the control panel, the main visualization, and the brushing plot. The control panel locates at the top-left corner of the interface. It contains three options: dataset selection, visual metric selection, and variable pair selection. Most of the interface is the main visualization with some bivariate area charts that are on the right side of the control panel. Below the control panel and on the left side of the main visualization, there is a space for the brushing vector plot. Below is descriptions and justifications for visualization design.

Time series of metric: We begin with a multivariate time series, in which there are three dimensions: data instances (data points), variables, and time. After preprocessing data with the min-max normalization, we select a variable pair and build a sequence of vector plots under a particular resolution, Δt. In detail, if t_1 is the first time step of the pair time series, the first vector plot depicts the evolution from t_1 to $t_1 + \Delta t$, the second plot illustrate the change from $t_1 + 1$ to $t_1 + 1 + \Delta t$, and so on. This sequence fills the whole temporal period. Because the series may be too long for us to analyze if we naively display all plots, we use a numerical visual metric as a representative of each plot. In other words, we compute the selected visual metric for every vector plot and display the time series of this metric. Visual metrics are statistics that digitalize visual features so that these metrics can help users quickly gain the patterns of interest [NNDH20]. In this paper, although we choose the list of visual metrics introduced in NetScatter [NHD21], the idea is applicable for any visual metric beyond the list.

Bivariate area charts: The purpose of the visualization is to provide a multi-resolution view so that users can compare and choose the appropriate resolution for revealing interesting information. If we display multiple time series of the selected visual metric with one series for one resolution, it is difficult for users to compare them and gain valuable information. Instead, we utilize the bivariate area charts with colors to strengthen the comparisons. Each area chart depicts the difference between metric series for a particular resolution with the baseline (the metric series on the top). The baseline is the time series of the selected metric for a resolution. In this work, the default resolution of the baseline is $\Delta t = 1$, for it is the finest level of evolution. However, it is possible to choose other resolutions for the baseline without any restriction. The red color indicates regions when the metric for current resolution is higher than that for the baseline, while the blue color implies lower values. The bivariate chart has an advantage in comparisons between two series without any modification of them, so we can also get their values along with comparisons. Besides, the number of bivariate area charts is flexible and depends on many factors, such as visualization space, temporal range, or data. For the last factor, because time series usually becomes less dependent on values too far in the past, high Δt is not necessary.

Interactive methods: One crucial mission of the visualization is to show the vector plots, for the metric series only guides users to interesting plots. It is the reason why we integrate interactive methods into the visualization of bivariate area charts. We implement a "lensing effect" that displays a few vector plots at their corresponding metric scores. Although we render the metric series as sequences of points connecting by lines, we assume each point similar to a vector plot whose size is extremely small. The assumption explains the name of this method. If the range of the lens is too long, we need to display many plots. It wastes rendering time and is not necessary because users can change the regions. Due to this reason, the range of the lens should be small for fast interactions. In addition, the lensing technique should be a linear function, such as table lensing function [RC94]. If we utilize a nonlinear function, e.g. the fisheye [SB92], length of two vectors of the same data point (or data instance) in two consecutive vector plots may be wrongly estimated.

Besides, users may want to investigate a particular vector plot, so we also implement the brushing. If we brush a vector plot in the lensing region, it will appear on the left side of bivariate area charts along with a start plot depicting all of its metric values. The vector plot has interactions with its vectors so give vectors' names. The start plot shows all visual metrics of the brushing plot, and thus, we can quickly notice visual patterns in the vector plot. There are other methods to represent multiple metrics in a single plot, such as a bar chart. Although the start plot may not be the optimal choice for this purpose, we choose it due to its aesthetic and natural-looking shape [BVB*13, BW14, KKG*20]. Also, radial representations are usually utilized for illustrating relationships among disparate entities (visual metrics in our case) [DLR09].

4. Use cases

4.1. Use case 1: US employment data

The first use case is the US employment data, which can be accessed via the website of the Bureau of Labor Statistics (`https://www.bls.gov/data/`). This data is the collection of monthly employees in every US state and region for several economic sectors from January 2000 to May 2020. We can utilize vector plots to display pairwise relations between economic sectors. In this case, each state and region is a vector in the plots.

Figure 3 gives the visualization of *Translation* series for the variable pair: Total Nonfarm vs. Total Private. We aim to compare 12 different resolutions from $\Delta t = 1$ to $\Delta t = 12$. There are two noticeable observations after comparisons between resolutions.

- In most period, the *Translation* scores increase as Δt increases.
- There is one period (from March to April 2020) when the *Translation* score for $\Delta t = 1$ is ridiculously high. Its peak is comparable to the highest peak for $\Delta t = 12$ (around 2008 and 2009) in height.

The first observation is straightforward, for *Translation* scores the amplitude of collection changes. If the US economy maintains its trend over a long period and most states have similar economic situations, the evolution over higher Δt certainly has a larger amplitude. In contrast, the second observation is abnormal, and we should investigate the resolution $\Delta t = 1$ to find out valuable information from the data. With the help of interactive methods introduced in Section 3, we can claim that the US experienced a serious drop in the number of employees due to Covid-19 from March to April 2020. Also, the level of crisis is comparable to the annual decrease of The Great Recession of 2008, but this drop happens much more faster (in only one month).

4.2. Use case 2: High-Performance Computing Center data

The second use case is the health metric readings of compute nodes in a High-Performance computing center [TTU20]. We collect CPU1 and CPU2 temperature of 467 compute nodes every 5 minutes. The relations between two CPUs temperature can be visualized by vector plots, with vectors represents computing nodes. Because 467 vectors raise the clutter issue [NHD21], we applied binned aggregation with leader algorithm to reduce the number of vectors to between 50 and 150.

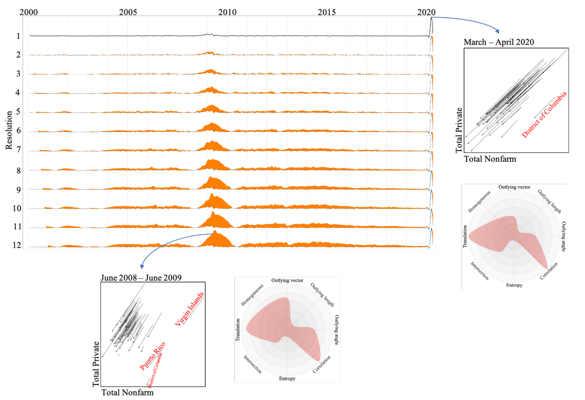

Figure 3: *Visualization of Translation series for the Total Nonfarm vs. Total Private.*

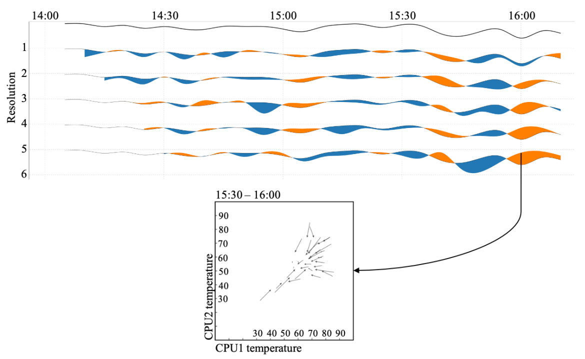

Figure 4: *Visualization of Entropy series for CPU1 vs. CPU2 temperature.*

Figure 4 shows visualization of *Entropy* series with 6 different resolutions from $\Delta t = 1$ to $\Delta t = 6$. There are also two noticeable observations in the Figure.

- Blue color dominates most regions. It implies *Entropy* score of the baseline ($\Delta t = 1$) is higher than that of other resolutions in most period.
- At around 16:00, red regions are visible, especially for $\Delta t = 4$, $\Delta t = 5$, and $\Delta t = 6$.

The *Entropy* scores complexity in the direction of vectors, so if we understand the behaviors of compute nodes, the first observation is reasonable. The temperature of compute nodes depends on many factors, so vectors with $\Delta t = 1$ tend to have various directions. However, if there is a collective event, such as the chill water issue that happened during the time we collected this data, most compute nodes have a common trend. The trend is only visible over a long enough period (high Δt) due to the noise of the time series. The second observation guides us to interesting information if we investigate those high resolutions. The system is shut down at 16:00, so all vectors point to the region around 60^0F (or around 15^0C), which is the room temperature. Besides the main purpose of the visualization, we can also see that vector plots can depict both temporal variation and spatial evolution of data points when the shutdown happens.

5. Discussions

Strength of the proposed method: Our interactive visualization supports the use of vector plots, which can directly display both temporal variation and spatial evolution (as can be seen in vector plot in Figure 4 for the shutdown of compute nodes). The visualization allows comparisons between different resolutions, and thus, it can guide users to a resolution that they can find out interesting information, e.g., the impact of Covid-19 on the US. Visual metrics can be considered a numerical representation of visual patterns in vector plots, so comparisons between the time series of the selected metric are also the comparisons between visual patterns of interest. In addition, interactive methods, such as lensing and brushing, help users to quickly investigate into vector plots corresponding to visual patterns that they want to observe.

Weakness of the proposed method: There are two main drawbacks of the visualization: it supports only one visual metric and one pair of variables. A vector plot may have several visual patterns, so it is more advanced to visualize many metrics in a single view. Also, the visualization focuses on only one variable pair, so it costs users' efforts in analyzing multiple variable pairs in the data.

6. Conclusions

We have proposed an interactive visualization that can ease the lack of global temporal information if we use vector plots for time series analysis. The visualization allows comparisons between different resolutions, so it can guide users to a resolution with interesting information. We have also demonstrated the method's benefits to two real data sets. Although the results show the potential of this approach, there are still weaknesses (as in Section 5) that we need to improve in future works.

References

[AJXW19] ALI M., JONES M. W., XIE X., WILLIAMS M.: Timecluster: dimension reduction applied to temporal data for visual analytics. *The Visual Computer 35*, 6 (2019), 1013–1026. 2

[BL19] BAQUI M., LÖHNER R.: Pedpiv: Pedestrian velocity extraction from particle image velocimetry. *IEEE Transactions on Intelligent Transportation Systems 21*, 2 (2019), 580–589. 1, 2

[BP02] BANDT C., POMPE B.: Permutation entropy: a natural complexity measure for time series. *Physical review letters 88*, 17 (2002), 174102. 2

[BS04] BARLOW N., STUART L. J.: Animator: A tool for the animation of parallel coordinates. In *Proceedings. Eighth International Conference on Information Visualisation, 2004. IV 2004.* (2004), IEEE, pp. 725–730. 2

[BVB*13] BORKIN M. A., VO A. A., BYLINSKII Z., ISOLA P., SUNKAVALLI S., OLIVA A., PFISTER H.: What makes a visualization memorable? *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2306–2315. 3

[BW14] BURCH M., WEISKOPF D.: On the benefits and drawbacks of radial diagrams. In *Handbook of human centric visualization*. Springer, 2014, pp. 429–451. 3

[DAW12] DANG T. N., ANAND A., WILKINSON L.: Timeseer: Scagnostics for high-dimensional time series. *IEEE Transactions on Visualization and Computer Graphics 19*, 3 (2012), 470–483. 2

[DKG12] DASGUPTA A., KOSARA R., GOSINK L.: Meta parallel coordinates for visualizing features in large, high-dimensional, time-varying data. In *IEEE Symposium on Large Data Analysis and Visualization (LDAV)* (2012), IEEE, pp. 85–89. 2

[DLR09] DRAPER G. M., LIVNAT Y., RIESENFELD R. F.: A survey of radial methods for information visualization. *IEEE transactions on visualization and computer graphics 15*, 5 (2009), 759–776. 3

[DW14] DANG T. N., WILKINSON L.: Scagexplorer: Exploring scatterplots by their scagnostics. In *2014 IEEE Pacific visualization symposium* (2014), IEEE, pp. 73–80. 1

[GRPF16] GRUENDL H., RIEHMANN P., PAUSCH Y., FROEHLICH B.: Time-series plots integrated in parallel-coordinates displays. In *Computer Graphics Forum* (2016), vol. 35, Wiley Online Library, pp. 321–330. 2

[JFSK15] JÄCKLE D., FISCHER F., SCHRECK T., KEIM D. A.: Temporal mds plots for analysis of multivariate data. *IEEE transactions on visualization and computer graphics 22*, 1 (2015), 141–150. 2

[KKG*20] KAMMER D., KECK M., GRÜNDER T., MAASCH A., THOM T., KLEINSTEUBER M., GROH R.: Glyphboard: Visual exploration of high-dimensional data combining glyphs with dimensionality reduction. *IEEE Transactions on Visualization and Computer Graphics 26*, 4 (2020), 1661–1671. 3

[LMS*12] LUBOSCHIK M., MAUS C., SCHULZ H.-J., SCHUMANN H., UHRMACHER A.: Heterogeneity-based guidance for exploring multiscale data in systems biology. In *2012 IEEE Symposium on Biological Data Visualization (BioVis)* (2012), IEEE, pp. 33–40. 2

[MTW*20] MA Y., TUNG A. K. H., WANG W., GAO X., PAN Z., CHEN W.: Scatternet: A deep subjective similarity model for visual analysis of scatterplots. *IEEE Transactions on Visualization and Computer Graphics 26*, 3 (March 2020), 1562–1576. 1

[NHD20a] NGUYEN B. D. Q., HEWETT R., DANG T.: Congnostics: Visual Features for Doubly Time Series Plots. In *EuroVis Workshop on Visual Analytics (EuroVA)* (2020), The Eurographics Association. 2

[NHD20b] NGUYEN B. D. Q., HEWETT R., DANG T.: Visual features for multivariate time series. In *Proceedings of the 11th International Conference on Advances in Information Technology* (2020), pp. 1–8. 2

[NHD21] NGUYEN B. D. Q., HEWETT R., DANG T.: Netscatter: Visual analytics of multivariate time series with a hybrid of dynamic and static variable relationships. In *Proceedings of the 14th IEEE Pacific Visualization Symposium* (2021). 1, 2, 3, 4

[NNDH20] NGUYEN N. V. T., NGUYEN B. D. Q., DANG T., HASS J.: Scagnosticsviewer: tracking time series patterns via scagnostics measures. In *Proceedings of the 13th International Symposium on Visual Information Communication and Interaction* (2020), pp. 1–4. 1, 2, 3

[RC94] RAO R., CARD S. K.: The table lens: merging graphical and symbolic representations in an interactive focus+ context visualization for tabular information. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (1994), pp. 318–322. 3

[SB92] SARKAR M., BROWN M. H.: Graphical fisheye views of graphs. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (1992), pp. 83–91. 3

[TAS04] TOMINSKI C., ABELLO J., SCHUMANN H.: Axes-based visualizations with radial layouts. In *Proceedings of the 2004 ACM symposium on Applied computing* (2004), pp. 1242–1247. 2

[TK07] TEKUSOVA T., KOHLHAMMER J.: Applying animation to the visual analysis of financial time-dependent data. In *2007 11th International Conference Information Visualization (IV'07)* (2007), IEEE, pp. 101–108. 2

[TTU20] TTU: High performance computing center (hpcc) at texas tech university. website, January 2020. Retrieved February 21, 2021 from `http://www.depts.ttu.edu/hpcc/`. 4

[WLG97] WEGENKITTL R., LOFFELMANN H., GROLLER E.: Visualizing the behaviour of higher dimensional dynamical systems. In *Proceedings. Visualization'97 (Cat. No. 97CB36155)* (1997), IEEE, pp. 119–125. 2

[WSSR20] WILLMAN C., SCOTT B., STONE R., RICHARDSON D.: Quantitative metrics for comparison of in-cylinder velocity fields using particle image velocimetry. *Experiments in Fluids 61*, 2 (2020), 1–16. 1, 2