Supplement

In this supplement, we first provide details on the datasets in Appendix A. We then discuss standard and topology-based evaluation metrics in Appendix B, followed by parameter configurations for base and augmented compressors in Appendix C. Next, we include analysis of reconstruction quality in Appendix D.

We describe algorithmic details in Appendix E, followed by additional information on run time analysis in Appendix F. We provide additional renderings for visual comparison in Appendix G. We discuss edge cases involved in cell correction in Appendix H. Finally, for readers who are mathematically inclined, we offer proofs for the lemmas and theorems in Appendix I and Appendix J; these can be skipped during the initial reading.

A An Overview of Datasets

We provide details on the datasets used in our experiments. Each dataset contains a set of 2D slices, obtained by slicing a corresponding 3D dataset along the z-axis and discarding information associated with the z direction. Any slices where the tensor field was entirely zero were removed.

The **Stress A** and **Stress B** datasets come from the public dataset associated with the work by Patel and Laidlaw [6]. The datasets are publicly accessible from the IEEE DataPort [7]; Stress A dataset corresponds to the first, whereas Stress B dataset corresponds to the third dataset in the DataPort.

The **Brain A** dataset comes from the public dataset of brain MRI scans from Tian et al. [9]. We use the data from patient 23. We then employ Diffusion Imaging in Python (DIPY) to extract the diffusion tensor field from the data.

The **Brain B** dataset is a scan of Dr. Gordan Kindlmann's brain that he released for public use. The dataset can be accessed from his personal homepage [5]. The dataset also comes with a mask indicating which areas are part of the brain. We set the tensors equal to zero in all areas outside of the brain (i.e. where the mask is not equal to 1).

The asymmetric tensor fields are all derived from 2D flow fields $v : \mathbb{R}^2 \to \mathbb{R}^2$. We compute the gradient at each point of a flow field, producing a 2D matrix. Now we describe the computation for $\frac{\partial v_x}{\partial x}$, which proceeds analogously for the other partial derivatives. Each flow field is defined on a regular grid $[n] \times [m]$ for some $m, n \in \mathbb{N}$. We describe the computation for a point $p = (p_x, p_y)$ at one of the grid points. The gradients are computed differently based on its location in the interior or at the boundary:

- $\underline{1 < p_x < n}$: $\frac{\partial v_x}{\partial x}\big|_p \leftarrow v_x(p_x+1, p_y) - v_x(p_x-1, p_y)$.
- $\underline{p_x = 1}$: $\frac{\partial v_x}{\partial x}\big|_p \leftarrow v_x(2, p_y) - v_x(1, p_y)$.
- $\underline{p_x = n}$: $\frac{\partial v_x}{\partial x}\big|_p \leftarrow v_x(n, p_y) - v_x(n-1, p_y)$.

The **Ocean** dataset comes from the Global Ocean Physics Reanalysis dataset from the E.U. Copernicus Marine Service [4]. For the flow field, we use the uo and vo fields from the daily data (file name: "cmems_mod_glo_phy_my_0.083deg_P1D-m"), which is then sliced in the range x : 100-200, y : 10-110 z : 0-26, where z corresponds to depth (all ranges are inclusive). We use the data on June 2, 2019.

The **Miranda** dataset comes from the hydrodynamics code for large turbulence simulations conducted by Lawrence Livermore National Laboratory. We use the x velocity and y velocity fields to derive the flow field. It can be accessed through the SDR Bench [2, 11].

The **Vortex Street** dataset comes from the Cylinder Flow with von Karman Vortex Street simulation [3, 8]. The **Heated Cylinder** dataset comes from the Heated Cylinder with Boussinesq Approximation simulation [3, 8]. Both datasets are accessed through the Computer Graphics Laboratory at ETH Zurich [1].

For each dataset, we eliminate any slices where all data points had the same value.

B Evaluation Metrics

We now discuss evaluation metrics. During compression, we compress each slice of each dataset separately, but report evaluation metrics aggregated across all datasets.

The *compression ratio* is the size of the ground truth data file divided by the size of the compressed file. To report the compression ratio for a dataset, we report the total size of every ground truth data slice divided by the total size of every compressed slice.

The *bit-rate* is the average number of bits used to encode each tensor. For each slice, we compute the bit-rate by dividing the compressed file size (in bits) by the number of tensors. We compute the bit-rate for a dataset by taking the average bit-rate across all slices.

Peak-Signal to Noise Ratio (PSNR) is typically defined for the compression of a single scalar field. If MSE is the mean squared error, and R is the range of the data, then PSNR for a scalar field is defined as

$$\text{PSNR} = 10 \log_{10} \left(\frac{R^2}{\text{MSE}} \right).$$

Our PSNR calculation for tensor fields is a bit more complex. Let D be a dataset containing a set of slices. If $s \in D$ is a slice, let $f_s : \mathbb{R}^2 \to \mathbb{T}$ be the ground truth tensor field and $f'_s : \mathbb{R}^2 \to \mathbb{T}$ the reconstructed tensor field. Let $v(s)$ be the set of vertices of each slice s. To compute the PSNR, we first compute two values for each slice s: a mean squared error MSE_s, and a range R_s. For asymmetric tensor fields, we define MSE_s by

$$\text{MSE}_s = \frac{1}{4|v(s)|} \sum_{x \in v(s)} \sum_{(i,j) \in [1,2]^2} (f(x) - f'(x))^2_{i,j}.$$

For a symmetric tensor field, we define MSE_s similarly, except that we ignore the T_{21} entry of each tensor T in our computation, because it is equal to the T_{12} entry. Corresponding to this omission, we divide by $3|v(s)|$ instead of $4|v(s)|$.

For a slice s, the range R_s is defined as the maximum entry across all tensors in s minus the minimum entry across all tensors of s. More formally, we define it as

$$R_s = \max_{\substack{x \in v(s) \\ (i,j) \in [1,2]^2}} \{f(x)_{i,j}\} - \min_{\substack{x \in v(s) \\ (i,j) \in [1,2]^2}} \{f(x)_{i,j}\}.$$

Let N be the number of slices. We then define the PSNR as:

$$\text{PSNR} = 10 \log_{10} \left(\frac{1}{N} \sum_{s \in D} \frac{R_s^2}{\text{MSE}_s} \right).$$

We define the PSNR in this way to account for the fact that certain slices may have different ranges.

C Parameter Configurations

We include the parameter configurations for our experiments. In Tab. 5, we include the error bounds used to generate the plots in Fig. 10 and Fig. 11. We use the same error bounds for the base and augmented compressors.

In Tab. 6, we include the error bounds used to generate Tab. 2. Each row represents a pair of trials, one using a base compressor, and the other using TFZ. For each row, we run the base compressor (listed under 'BC'), as well as TFZ augmenting that base compressor on the given dataset. For the base compressor, we use the error bound listed under 'BC-EB'. For TFZ, we use the error bound listed under 'A-EB'.

We include a similar table, Tab. 7, describing the error bounds used to generate Fig. 1, Fig. 8, Fig. 9, Fig. 17, Fig. 18, and Fig. 19.

D Reconstruction Quality

In this section, we give further analysis of reconstruction quality. We analyze the tradeoff between bit-rate and PSNR for asymmetric tensor fields in Appendix D.1. We provide error maps in Appendix D.2. We give a distribution of the errors imposed by the base compressors (and fixed during augmentation) in Appendix D.3. Finally, we analyze the causes of visual artifacts in Appendix D.4

Table 5: Parameter configurations used to generate Fig. 10 and Fig. 11. BC: Base Compressor.

Dataset	BC	Error Bounds (ξ)
Stress A	SZ3	0.006, 0.0195, 0.033, 0.0465, 0.06
Stress A	SPERR	0.006, 0.0195, 0.033, 0.0465, 0.06
Stress B	SZ3	0.006, 0.0195, 0.033, 0.0465, 0.06
Stress B	SPERR	0.006, 0.0195, 0.033, 0.0465, 0.06
Brain A	SZ3	0.006, 0.0195, 0.033, 0.0465, 0.06
Brain A	SPERR	0.006, 0.0195, 0.033, 0.0465, 0.06
Brain B	SZ3	0.006, 0.0195, 0.033, 0.0465, 0.06
Brain B	SPERR	0.006, 0.0195, 0.033, 0.0465, 0.06
Ocean	SZ3	0.003, 0.00975, 0.0165, 0.02325, 0.03
Ocean	SPERR	0.003, 0.00975, 0.0165, 0.02325, 0.03
Miranda	SZ3	0.0003, 0.000975, 0.00165, 0.002325, 0.003
Miranda	SPERR	1×10^{-5}, 3×10^{-5}, 9.75×10^{-5}, 0.000165, 0.0002325
Vortex Street	SZ3	5×10^{-5}, 0.0001625, 0.000275, 0.0003975, 0.0005
Vortex Street	SPERR	1×10^{-5}, 3.25×10^{-5}, 5.5×10^{-5}, 7.05×10^{-5}, 0.0001
Heated Cylinder	SZ3	5×10^{-5}, 0.0001625, 0.000275, 0.0003975, 0.0005
Heated Cylinder	SPERR	5×10^{-5}, 0.0001625, 0.000275, 0.0003975, 0.0005

Table 6: Parameter configurations used to generate Tab. 2. BC: Base compressor used in each trial. A-EB: Error bound for an augmented compressor. BC-EB: Error bound for the base compressor that produces a similar compression ratio to the augmented counterpart (i.e., when the augmented compressor uses the error bound listed under A-EB).

Dataset	BC	A-EB	BC-EB
Stress A	SZ3	0.01	0.00953125
Stress A	SPERR	0.01	0.009609375
Stress B	SZ3	0.01	0.0090625
Stress B	SPERR	0.01	0.009140625
Brain A	SZ3	0.01	0.00265625
Brain A	SPERR	0.01	0.00203125
Brain B	SZ3	0.01	0.00111328125
Brain B	SPERR	0.01	0.00056640625
Ocean	SZ3	0.001	0.00075
Ocean	SPERR	0.001	0.00065625
Miranda	SZ3	0.001	5.078125×10^{-5}
Miranda	SPERR	0.001	9.765625×10^{-6}
Vortex Street	SZ3	0.001	6.4453125×10^{-5}
Vortex Street	SPERR	0.001	$3.466796875 \times 10^{-5}$
Heated Cylinder	SZ3	0.001	0.00013671875
Heated Cylinder	SPERR	0.001	0.0001484375

D.1 Compression of Asymmetric Tensor Fields

In addition to the experiments highlighted in Sec. 5, we study the tradeoff between bit-rate and PSNR for TFZ when compressing asymmetric tensor fields. We preserve the topology of either the eigenvector or the eigenvalue partition, but not both, as illustrated in Fig. 11. In Fig. 11, we plot the tradeoff between bit-rate and PSNR for SZ3, SPERR, augmented SZ3 and augmented SPERR. We show the same tradeoff for augmented SZ3, which preserves the topology of either the eigenvalue or the eigenvector partition (but not both), denoted as A-SZ3 (val) or A-SZ3 (vec). We also show the corresponding curves for augmented SPERR.

In Fig. 11, we observe that preserving only one type of partition results in lower storage overhead compared to preserving both. In general, preserving the topology of the eigenvalue partition requires less storage than preserving the topology of the eigenvector partition.

Table 7: Parameter configurations used to generate Fig. 1, Fig. 8, Fig. 9, Fig. 17, Fig. 18, and Fig. 19. BC: Base compressor used in each trial. A-EB: Error bound used for an augmented compressor. BC-EB: Error bound for the base compressor that produces a similar compression ratio to the augmented counterpart (i.e., when the augmented compressor uses the error bound listed under A-EB.)

Dataset	BC	A-EB	BC-EB
Stress A	SZ3	0.06	0.028798828125
Stress A	SPERR	0.06	0.052265625
Stress B	SZ3	0.01	0.03609375
Stress B	SPERR	0.01	0.038671875
Brain A	SZ3	0.06	0.0196875
Brain A	SPERR	0.06	0.0234375
Brain B	SZ3	0.03	0.00228515625
Brain B	SPERR	0.03	0.001875
Ocean	SZ3	0.01	0.003984375
Ocean	SPERR	0.01	0.00359375
Miranda	SZ3	0.003	5.859375×10^{-5}
Miranda	SPERR	0.0002325	$1.1806640625 \times 10^{-5}$
Vortex Street	SZ3	0.0005	6.4453125×10^{-5}
Vortex Street	SPERR	0.0001	2.8515625×10^{-5}

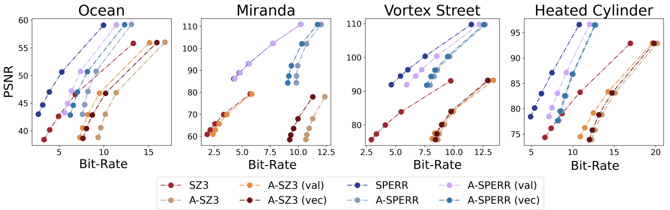


Fig. 11: Plots showing the tradeoff between bit-rate and PSNR for SZ3 and SPERR on each dataset, as well as the augmented compressors. SZ3 (val) and SPERR (val) respectively preserve the topology of the eigenvalue partition only. SZ3 (vec) and SPERR (vec) respectively preserve the topology of the eigenvector partition only.

Specifically, for the Miranda dataset, preserving the topology of the eigenvalue partition incurs almost no storage overhead, whereas significant overhead is needed to preserve the topology of its eigenvector partition.

D.2 Error Maps

In Fig. 12, We include an error map of the Vortex Street dataset compressed with the SZ3 compressor both (A) before augmentation with TFZ and (B) after augmentation. Here, we can see that the error profile of the dataset is generally the same before and after augmentation. However, there are a few areas where augmentation increases or decreases the error in order to correct the topology.

In Fig. 13, we include an error map of the Stress B dataset. In the top row, we include an error map similar to that showed in Fig. 12 compressed with SZ3 (A) before augmentation and (B) after augmentation with $\xi = 0.01$. We show a 65×65 pixel grid, where each pixel corresponds to a tensor. We can see that the error profile is nearly identical in both cases. In the bottom row, we show an error profile for the eigenvector directions of the Stress B dataset (C) before augmentation and (D) after augmentation with $\xi = 0.01$. We show a linearly interpolated view, rather than a pixel grid as before, due to the nontrivial fashion in which the tensorlines behave under interpolation. We can see that the error profile is nearly identical before and after augmentation, except that, in a few areas of where the base compressor has significantly distorted the eigenvector directions, TFZ corrects the error in those regions during augmentation.

D.3 Errors Corrected

In Fig. 14, we give distributions of the different errors achieved on each asymmetric dataset by each base compressor before augmentation with TFZ. Here, we can see that the percentage of points that are

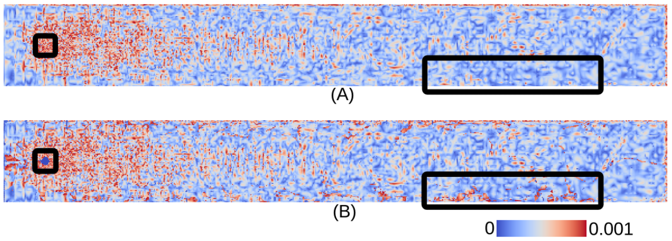


Fig. 12: Error maps of Vortex Street dataset compressed with SZ3 both (A) before augmentation and (B) after augmentation. The error value is given as the maximum error across the four entries of each tensor as a percentage of the range. We used $\xi = 0.001$ for compression.

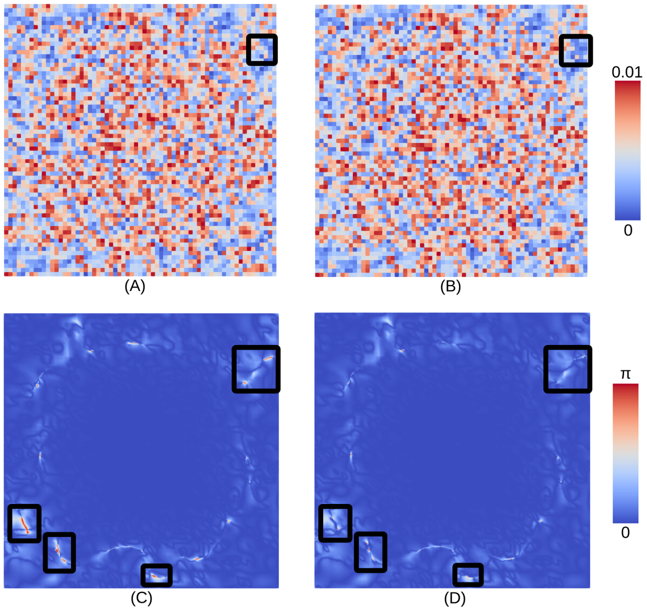


Fig. 13: Error maps of the stress B dataset compressed with SZ3 and augmented SZ3. Top row: error map of the maximum error across the four entries of each tensor as a percentage of the range (A) before augmentation and (B) after augmentation. Each pixel represents a tensor. Bottom row: error map of the angle of the tensorlines (C) before augmentation and (D) after augmentation. We interpolate the tensor field to show distortions in the interior of cells. We used $\xi = 0.01$ for compression.

misclassified according to the eigenvector and eigenvalue partitions are less than the total number of cells whose topology is distorted according to the eigenvector and eigenvalue partitions, respectively. This is logical, because whenever a point has the wrong classification type according to the eigenvalue or eigenvector partition, it distorts the topology of all six surrounding cells. Likewise, we found that, in most cases, correcting the vertices of a cell is sufficient to correct the internal topology of the cell, and further correction of the internal topology of a cell is not needed. We can see that there are typically very few errors related to degenerate points. To generate these numbers, we used the largest error bound for each combination of a dataset and base compressor listed in Tab. 5.

D.4 Visual Artifacts

One limitation of TFZ is that the decompressed data may exhibit visual artifacts, notably when using tensorline LIC visualization for symmetric data, and eigenvector/eigenvalue partition visualization for asymmetric data. In the symmetric case, such artifacts primarily originate from the base compressor, rather than our augmentation layer. One can verify mathematically that the eigenvector directions are determined entirely by the value θ in the decomposition equation. Our augmentation layer can only improve the value of θ produced from the base compressor, and thus will not further distort the eigenvector directions. As shown in Fig. 13, our strategy produces an error profile for the eigenvector directions that is very similar to the base compressor, except for a few areas where TFZ has corrected some distortions. We observed that distortions are most prominent in regions where tensors magnitudes are low compared to the absolute error bound, allowing for greater relative distortion.

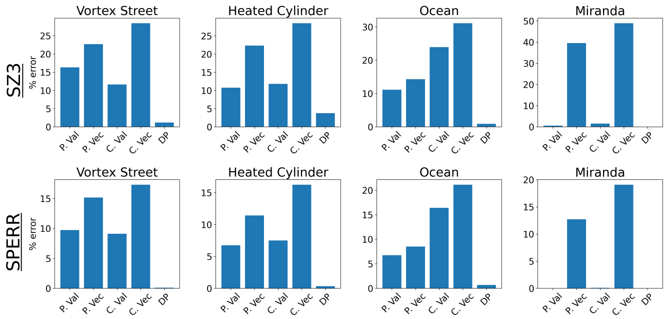


Fig. 14: Histograms demonstrating the percentage of points and cells in each asymmetric dataset that exhibit certain types of errors before augmentation with TFZ. P. Val: Incorrect point classification in eigenvalue partition. P. Vec: Incorrect point classification in eigenvector partition. C. Val: Incorrect cell classification in eigenvalue partition. C. Vec: Incorrect cell classification in eigenvector partition (excluding degenerate points). DP: Errors pertaining to degenerate points within each cell.

In the asymmetric case, our strategy can cause visual artifacts. We particularly notice jagged artifacts, such as those in the zoomed-in region of Fig. 1. These artifacts are most prominent in regions where the decomposition coefficients $(\gamma_d, \gamma_r, \gamma_s)$ are small compared to the absolute error bound ξ. In such cases, if our strategy replaces e.g. $\gamma_d' \leftarrow \gamma_d' + \xi$, then the magnitude of γ_d' can be significantly increased. If one coefficient of a tensor is increased in this manner while a neighboring tensor still has very small coefficients, the resulting discrepancy can produce jagged artifacts. However, our strategy ensures that the boundaries of partition regions will not shift by more than the width of a cell. Thus, in high resolution datasets, such as Fig. 9, these artifacts will be less noticeable.

E Algorithm Details

In this section, we provide additional details about our topology-preserving compression algorithm, for both symmetric (Appendix E.1) and asymmetric (Appendix E.2) cases.

E.1 Compression of Symmetric Tensor Fields

Let σ be a (triangular) cell. Let T_1, T_2, and T_3 be the tensors at the vertices of σ for the ground truth data, and T_1', T_2' and T_3' be the corresponding tensors in the reconstructed data. Following the decomposition in Eq. (3), we obtain coefficients $\gamma_{d,1}$ for T_1, $\gamma_{d,1}'$ for T_1' and so forth.

Suppose that we have handled case 1 of the cell correction step identified in Sec. 4.2. It is possible that, due to the limited precision afforded by linear-scaling quantization, the cell topology will not be preserved. In this case, we store the deviators of each tensor losslessly, one at a time, until the cell topology is preserved. In particular, for $i \in \{1, 2, 3\}$, we store $\Delta_i = \gamma_{s,i}\cos(\theta)$ and $F_i = \gamma_{s,i}\sin(\theta)$ losslessly. During decompression, we compose a tensor according to

$$T_i'' \leftarrow \gamma_{d,i}' + \begin{pmatrix} \Delta_i & F_i \\ F_i & -\Delta_i \end{pmatrix}. \tag{9}$$

In extremely rare cases, even storing all three deviators losslessly will not preserve the cell topology or respect the error bound, in which case we preserve the tensors completely losslessly. Likewise, it is possible that after adjusting one of the θ_i' using linear-scaling quantization, the error bound will not be respected. If this occurs, we store the deviator losslessly (and store the tensor losslessly if necessary).

E.2 Compression of Asymmetric Tensor Fields

In rare cases, it is possible for the vertex correction step in Sec. 4.3.1 to fail. There are two ways that it can fail.

First, although each coefficient individually respects the error bound, it is possible—though unlikely—that T'' (the reconstructed tensor) does not. In such cases, if θ' is not currently quantized, we quantize it using the same strategy as in the symmetric case. Otherwise, we revert θ' to its unquantized value and apply logarithmic-scaling quantization to γ_d', γ_r', and γ_s', using their values from the intermediate data prior to the procedure described in Sec. 4.3.1. By Lemma 3, these values deviate from their respective ground truth values by at most ξ, ξ, and $\sqrt{2}\,\xi$, respectively. We then use logarithmic-scaling quantization to halve these error bounds and repeat the logic from the vertex correction step. This process of quantizing θ' and halving each error bound is repeated up to ten times, after which T is stored losslessly.

Second, two coefficients may have equal magnitude in the reconstructed data but not in the ground truth, potentially causing classification issues. In such cases, as well as when an error is detected in the internal cell topology during the topology preservation step, we apply the process described above.

F Additional Description on Run Time Analysis

In this section, we provide additional information pertaining to the run times of TFZ. We explore the tradeoff between the total compression time, augmentation time, and the error bound ξ in Appendix F.1. We report in Appendix F.2 the run times for asymmetric data when preserving one type of partition (but not both). We give a disaggregation of the run times during the cell correction step; see Appendix F.3. We give statistics on how many times cells are visited in Appendix F.4. Finally, we demonstrate the abilities of TFZ to compress data with larger 2D slices in Appendix F.5.

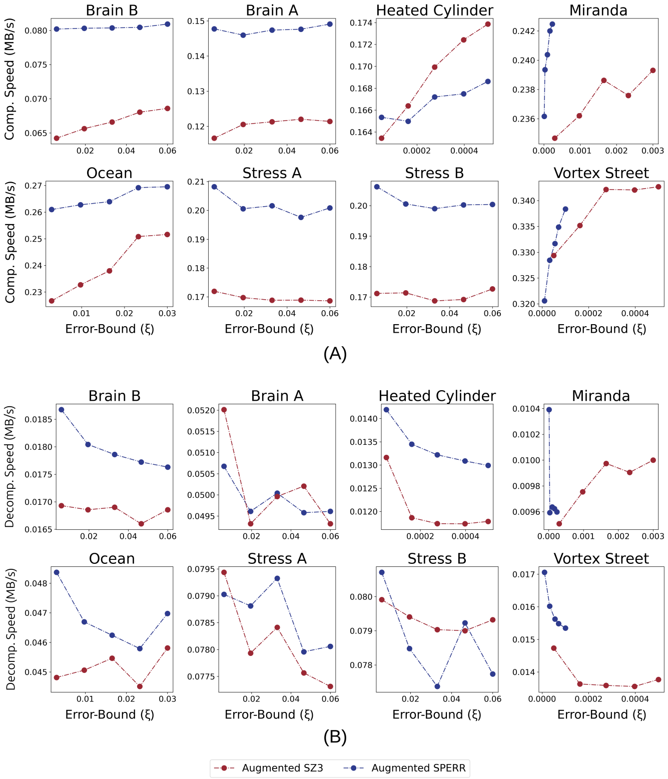


Fig. 15: Plots showing throughput (in MB/s) versus error bound ξ for each dataset, using augmented SZ3 and augmented TTHRESH: (A) throughput of compression; (B) throughput of decompression.

F.1 Run Time Versus Error Bound

In Fig. 15, we provide plots demonstrating the throughput in MB/s versus error bound ξ for both augmented SZ3 and augmented TTHRESH on all eight datasets. We display the throughput for compression in (a) and decompression in (b). In general, there is no clear trend between the error bound and the throughput of either compression or decompression.

F.2 Preserving Eigenvalue or Eigenvector Partitions

In Tab. 8, we provide the compression and decompression times for SZ3, augmented SZ3, SPERR, and augmented SPERR. We also provide the run times preserving eigenvalue or eigenvector partition (but not both). All times reported are for $\xi = 0.001$.

Preserving the topology of only one partition (eigenvector or eigenvalue) can reduce compression time by up to about 45%, with more time saved when preserving the eigenvector partition (as opposed to the eigenvalue partition). Decompression time can be reduced by up to about 40%. The gains in decompression time are generally smaller than those for compression time, typically under 10%. However, the time savings vary across datasets and (base) compressors.

F.3 Run Time Analysis of Cell Compression Step

For asymmetric data, the cell correction step consists of three main parts: (1) correcting the vertex classifications, (2) preserving the degenerate points of the dual-eigenvector fields, and (3) preserving the topology of each cell. In Tab. 9, we present the run times (in seconds) for each step across datasets and compressors. It is evident that preserving cell topology requires the most time in each trial, although all three steps contribute significantly to the overall run time of the cell correction process.

F.4 Statistics on Iterations

In this section, we analyze how many times cells are visited to provide insight into the overall running times. In Fig. 16, we show how many times each cell is processed for each dataset for SZ3. The numbers are similar for SPERR. Here, we can see that almost all points (90-95%) are processed only once in general, while very few cells are processed six or more times. The notable exception is the Vortex Street dataset, where many points are processed six or more times. One can verify that the Vortex Street dataset has the lowest throughput overall, and this may be the cause.

To generate Fig. 16, we use the largest error bound for each dataset given in Tab. 5 for augmented SZ3.

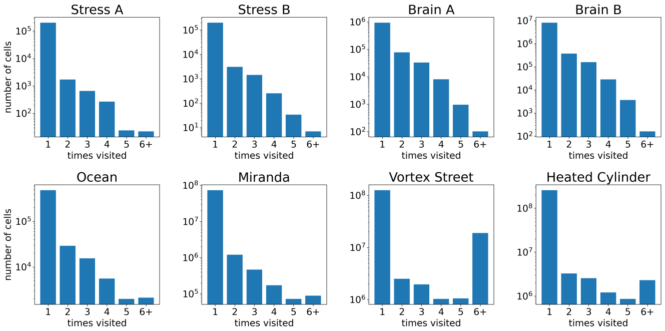


Fig. 16: Histograms demonstrating the distribution of how many times cells were processed. For example, the "1" bar shows the number of cells that were processed only once. "6+" gives the number of cells that were processed six or more times.

F.5 Testing on Data With Large Slices

In most of our experiments, we use relatively small 2D slices—smaller than those in typical use cases. Since our algorithm runs in linear time, we believe these experiments are sufficient to demonstrate the framework's throughput. To verify that the throughput observed on smaller datasets also scales to larger data, we run TFZ on datasets with larger

Table 8: Run times (compression and decompression times) for SZ3, augmented SZ3, SPERR, and augmented SPERR on each of the four asymmetric datasets. We also provide the run times for augmented compressors preserving one type of partition, as A-SZ3 (val), A-SZ3 (vec), etc. We provide similar run times for augmented SPERR. $\xi = 0.001$. Times are in seconds.

Dataset	SZ3	A-SZ3	A-SZ3 (val)	A-SZ3 (vec)	SPERR	A-SPERR	A-SPERR (val)	A-SPERR (vec)
Total Compression Time								
Ocean	0.35	1.90	1.59	1.61	0.39	2.28	1.87	1.97
Miranda	7.00	274.14	161.84	213.26	9.53	287.16	171.51	221.46
Vortex Street	24.34	784.85	526.90	755.19	28.86	835.99	490.62	776.49
Heated Cylinder	37.98	762.21	655.79	717.39	48.23	713.11	609.04	692.66
Decompression Time								
Ocean	0.34	0.38	0.39	0.38	0.36	0.42	0.42	0.42
Miranda	4.73	11.44	7.06	11.21	6.34	12.60	9.12	11.93
Vortex Street	19.78	33.76	32.99	32.96	22.81	35.13	34.57	34.97
Heated Cylinder	29.01	50.78	46.56	49.56	36.68	53.44	51.22	53.07

Table 9: Run times (in seconds) during cell correction for three asymmetric datasets. We display the times for augmented SZ3 in the top and augmented SPERR in the bottom. 'Vertices': correcting the vertex classifications; 'Degen. Pts.': preserving the degenerate points of the dual-eigenvector fields; 'Cell Top.': preserving the topology of each cell.

Dataset	Vertices	Degen. Pts.	Cell Top.
Augmented SZ3			
Ocean	0.23	0.17	0.68
Miranda	53.00	31.52	153.31
Vortex Street	192.46	125.35	368.00
Heated Cylinder	179.34	102.34	334.12
Augmented SPERR			
Ocean	0.25	0.17	0.79
Miranda	54.19	32.12	156.66
Vortex Street	207.51	139.99	372.44
Heated Cylinder	166.35	91.92	283.27

2D slices. Specifically, we derive an asymmetric tensor field from the E.U. Copernicus Marine Service Global Ocean Physics Reanalysis dataset (the source of our main ocean dataset). We select the largest rectangular slice available that does not significantly overlap with land. Our slice is taken from the South Pacific, bounded by the following coordinates: North: 6.870530532659802, East: -80.59961032889228, South: -71.2633145566259, and West: -181.429859882464. The depth ranges from 1.54 m at the top to 2533.3359375 m at the bottom. We derive the asymmetric tensor field using the same strategy applied to the other asymmetric datasets, and obtain a symmetric tensor field by taking the symmetric part of the tensor field.

Overall, our tensor fields have 42 slices of size 1210×938. The symmetric tensor field has a file size of 1144MB, while the asymmetric tensor field is 1525MB. We compress the symmetric dataset with $\xi = 0.01$ and the asymmetric dataset with $\xi = 0.001$. We obtain compression times of 121.7s for the symmetric tensor field, and 416.9s for the asymmetric tensor field, yielding respective throughput of 9.4 MB/s and 3.6 MB/s. These numbers are comparable to those yielded by the smaller datasets. The symmetric dataset has a bit-rate of 4.9 and a PSNR of 47.3, while the asymmetric dataset has a bit-rate of 13.6. and a PSNR of 65.7. These numbers are consistent with the reconstruction qualities reported on the smaller datasets.

G Additional Renderings for Visual Comparison

In this section, we provide additional renderings of our experimental datasets for visual comparison. In Fig. 17, we visualize Stress A and Brain A datasets compressed with SZ3, augmented SZ3, SPERR, and augmented SPERR, respectively. We chose an error bounds ξ such that each augmented compressor achieves a similar compression ratio to its corresponding base compressor. For the Stress A dataset, we highlight a region of interest. And for the Brain A dataset, we provide a zoomed-in view to mark the discrepancies between the base compressors and the ground truth, not visible in the augmented compressors. In Fig. 18, we provide a similar view for the Vortex Street and Heated Cylinder datasets. In Fig. 19, we include a variation of Fig. 1 using SPERR and augmented SPERR (instead of SZ3 and augmented SZ3). We report the error bounds ξ used to generate each figure in Appendix C.

H Edge Cases During Asymmetric Cell Correction

We handle various edge cases that arise in our topology-preserving framework. We describe edge cases pertaining to vertex correction in Appendix H.1 and cell topology preservation in Appendix H.2. We describe how we handle issues arising from floating point precision in Appendix H.3.

H.1 Vertex Correction

We assign values to the variables D, R, RS, and D_m to ensure that, for each vertex of the mesh, the following conditions hold if and only if they hold in the ground truth:

- $|\gamma'_r| = \gamma'_s$
- $\gamma'_r = 0$
- $|\gamma'_d| = |\gamma'_r| > \gamma'_s$
- $|\gamma'_d| = \gamma'_s > |\gamma_r|$
- $|\gamma'_r| = |\gamma'_d| = |\gamma_s|$

To accomplish this, we adjust the decompressed data using strategies similar to those in Sec. 4.3.1.

H.2 Cell Topology Preservation

Sometimes, edge cases arise when computing the topological invariant for preserving cell topology using TFZ. Each edge case yields a number of sub-cases. We describe non-transverse intersections in Appendix H.2.1, junction points on cell boundaries in Appendix H.2.2, and intersections with cell vertices in Appendix H.2.3. Finally, we describe how we handle degenerate conics in Appendix H.2.4.

H.2.1 Non-Transverse Intersections

When computing the topological invariant, we trace the curves $\gamma_d^2 = \gamma_s^2$ and $\gamma_r^2 = \gamma_s^2$. An edge case occurs when one of these curves intersects the cell boundary or the other curve non-transversally (i.e, an intersection that does not satisfy the transversality condition). It is also possible for the two conics to have significant overlap but not be equal. We handle each case using a virtual perturbation. We describe these cases in Tab. 10. In the left column, we provide a description of the case and how we handle it. In the middle column, we visualize the case. In the right column, we visualize the virtual perturbation.

There are also a few other cases of non-transverse intersections that cannot be easily visualized. We describe them below:

- If, for the entire cell, $|\gamma_d| = |\gamma_r|$, then we proceed as though $|\gamma_r| > |\gamma_d|$.
- If, for the entire cell, $|\gamma_d| = \gamma_s$, then we proceed as though $\gamma_s > |\gamma_d|$.
- If, for the entire cell, $|\gamma_r| = \gamma_s$, then we proceed as though $\gamma_s > |\gamma_r|$.
- If there exists a point $z \in \sigma$ where $\gamma_d(z) = \gamma_r(z) = \gamma_s(z) = 0$, we note this in our invariant. We also note whether it occurs in the interior, on an edge, or on a vertex.
- If, for the entire cell, $\gamma_d = \gamma_r = \gamma_s = 0$, then we note this as part of our invariant, and save the cell losslessly.

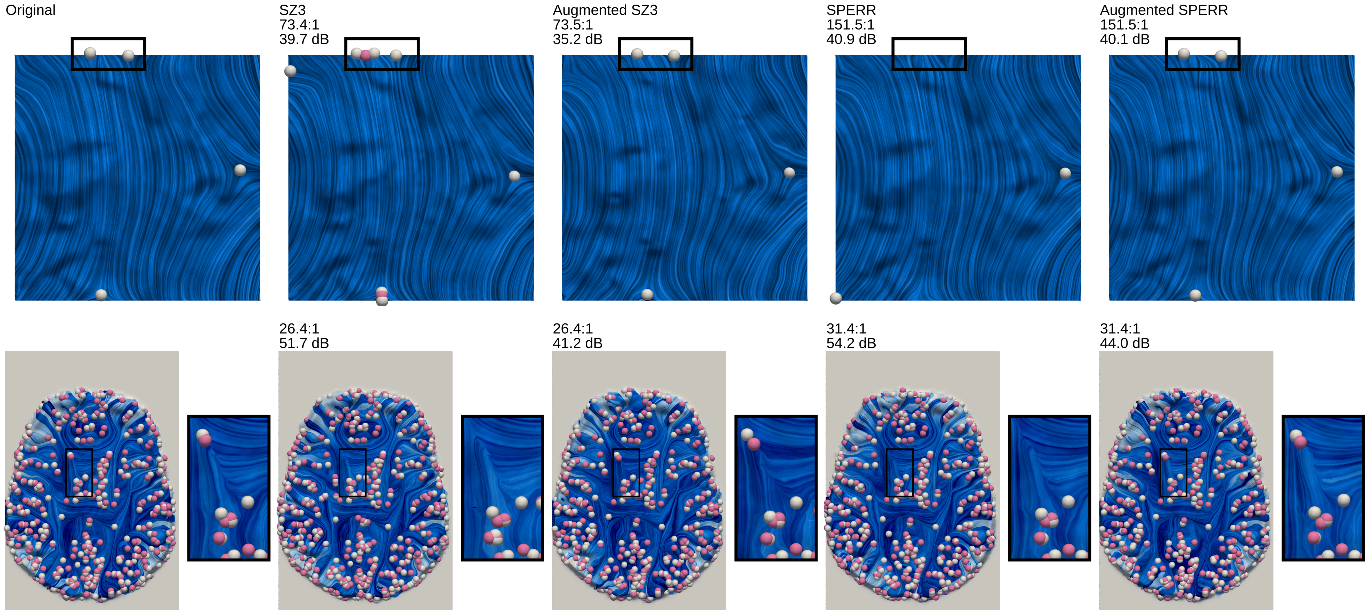


Fig. 17: LIC visualization of the eigenvector fields of two 2D symmetric second-order tensor fields compressed with SZ3, augmented SZ3, SPERR, and augmented SPERR, along with the ground truth. Trisectors are in white, wedges are in pink. Top: Stress A data slice 13. Bottom: Brain A data slice 50. In the top row, we highlight a region of interest in black boxes. In the bottom row, we provide a zoomed-in view of a region of interest encloded by black boxes. The Z position of each point corresponds to the Frobenius norm with smoothing applied.

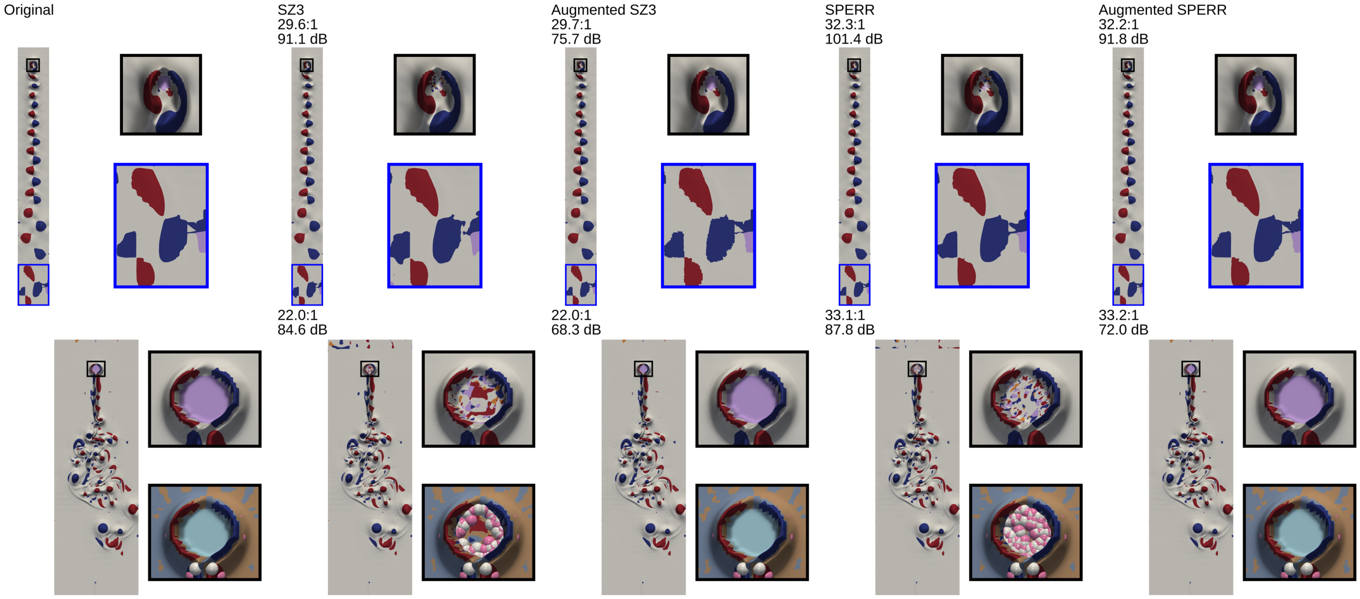


Fig. 18: Visualizing the eigenvalue partition of the Vortex Street (top) and Heated Cylinder (bottom) datasets (slices 1000 and 800 resp.) compressed with SZ3, augmented SZ3, SPER, and augmented SPERR, along with the ground truth. We provide zoomed-in views (of black and blue boxes) that highlight the differences between the compressors and the ground truth. For the heated cylinder dataset, one zoomed in view corresponds to the eigenvector partition. We also label compression ratio and PSNR. We use the same colormap as Fig. 3. The Z position of each point corresponds to the Frobenius norm with smoothing applied.

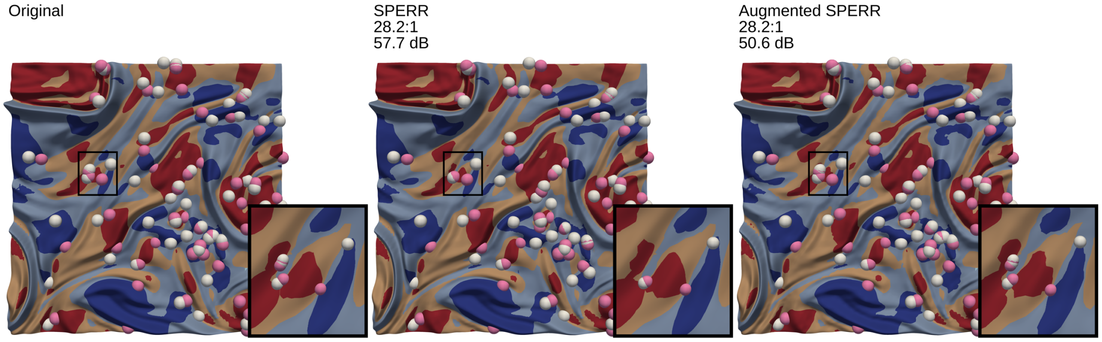
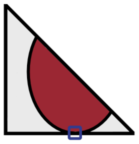
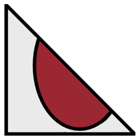
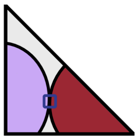
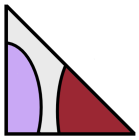
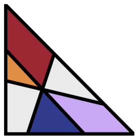
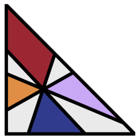


Fig. 19: Visualizing the eigenvector partition of the Ocean dataset compressed with SPERR and augmented SPERR. Left: input data visualized with degenerate points of the dual-eigenvector field. Trisectors are in white, wedges are in pink. Middle: reconstructed data using SPERR, labeled with compression ratio and PSNR. Right: reconstructed data using augmented SPERR along with compression ratio and PSNR. We also provide zoomed-in views that highlight the differences between the classic and augmented SPERR results. We use the same colormap as Fig. 3. The Z position of each point corresponds to the Frobenius norm with smoothing applied.

Finally, consider the case where the curves $|\gamma_r| = \gamma_s$ and $|\gamma_d| = \gamma_s$ are the same curve, but $|\gamma_d| \neq |\gamma_r|$ for the entire cell. Because γ_r and γ_d are both PL functions over σ, this can occur only if $|\gamma_r| = \gamma_s$ is a line. In such a case, one of $|\gamma_r|$ or $|\gamma_d|$ will be smaller on the entire cell; we ignore whichever is smaller.

Table 10: Descriptions and images of edge cases that arise in internal cell topology computation from non-transverse intersections. We describe each case and how it is handled in the first column. In the second column, we provide a visualization of the case. In the third column, we provide an image of the virtual perturbation used to handle the case.

Case	Image	Perturbation
Non-transverse edge intersection: If one of the conics intersects the edge non-transversally, then we apply a virtual perturbation such that the intersection never occurs.		
Non-transverse intersection between conics: If two of the conics intersect non-transversally, then we apply a virtual perturbation such that the intersection never occurs.		
Partial overlap of conics: If the conics are degenerate (e.g., intersecting lines, parallel lines, etc.), they may not be exactly identical but can still exhibit substantial overlap. In this case, we perturb the conics so that their intersection has measure zero.		

H.2.2 Junction Point at Cell Boundary

If a junction point occurs on the boundary of a cell, this can lead to ambiguity. There are many variations of this case. We handle them all using virtual perturbations. We describe how we handle such cases in Tab. 11.

Table 11: Descriptions and images of edge cases that arise in internal cell topology computation when a junction point occurs on the cell boundary. We describe each case and how it is handled in the first column. In the second column, we provide a visualization of the case. In the third column, we provide an image of the virtual perturbation used to handle the case.

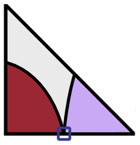
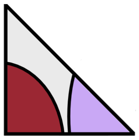
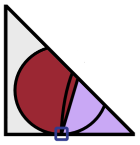
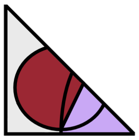
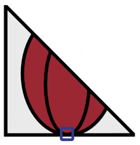
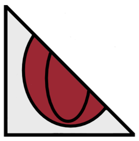
Case	Image	Perturbation
Junction point at an edge with transverse intersection: If a junction point occurs at an edge, and the two conics both intersect the edge transversally, then we apply a virtual perturbation such that the junction point does not occur.		
Junction point at an edge with single non-transverse conic-edge intersection: If a junction point occurs at an edge, and exactly one conic intersects the edge non-transversally at the junction point, then we apply a virtual perturbation so that the nontransverse intersection does not occur.		
Junction point at an edge with only non-transverse intersections: If a junction point occurs at an edge, and all intersections are non-transverse, then we apply a virtual perturbation so that none of the intersections occur.		

Table 12: Descriptions and images of edge cases that arise in internal cell topology computation when a conic section intersects a vertex. In the first column we describe each case and how it is handled. In the second column we provide a visualization of the case.

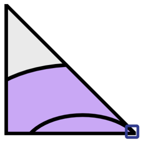
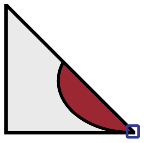
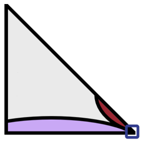
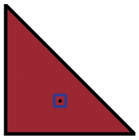
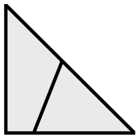
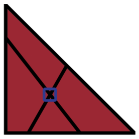
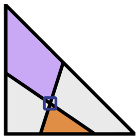
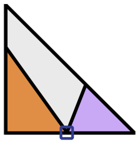
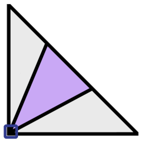
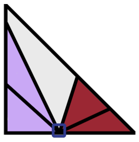
Case	Image
Conic intersects a vertex (no topological effect): If one conic section intersects a vertex, but that intersection does not mark the location of any topological change, then we ignore it.	
Conic intersects a vertex (topologically significant intersection): If one conic section intersects a vertex, and that intersection does affect the topology, then we track this intersection in our invariant.	
Two conics intersect at a vertex: If two conic intersects intersect at a vertex, we ignore the junction point that occurs at the vertex. We treat each of the two conics as separately intersecting the vertex and handle them according to the two previous cases.	

Table 13: Descriptions and images of edge cases that arise in internal cell topology computation from degenerate conic sections. In the first column we describe each case and how it is handled. In the second column we provide a visualization of the case.

Case	Image
One conic is a single point: In this case, we apply a virtual perturbation so that the conic disappears.	
One conic does not separate two regions appropriately: In particular, if the conic $\gamma_d = \gamma_s$ has $\gamma_s > \lvert\gamma_d\rvert$ on both sides, or if the conic $\gamma_r = \gamma_s$ has $\gamma_s > \lvert\gamma_r\rvert$ on both sides, we ignore the conic.	
One conic consists of two intersecting lines. The intersection point does not lie on the boundary of two regions: If the intersection point does not divide regions of different classifications, we ignore the intersection point. In future cases, assume that any conic that is two intersecting lines lies on the boundary between different regions.	
One conic consists of two intersecting lines. The intersection point lies on the boundary of two regions: In this case, we track the location of the intersection point relative to any edge intersections and junction points in the invariant.	
One conic consists of two intersecting lines. The intersection point is on an edge: In this case, we track the location of the intersection point relative to any edge intersections and junction points in our invariant, noting the edge on which it occurs.	
One conic consists of two intersecting lines. The intersection point is on a vertex: In this case, we track the location of the intersection point relative to any edge intersections and junction points in our invariant, noting the vertex on which it occurs.	
Both conics, each composed of two intersecting lines that share the same point of intersection: In this case, we do not track any junction point. We handle each conic separately according to the previous cases.	

H.2.3 Intersections at Vertices

If one of the conics intersects a vertex of the cell, this can lead to ambiguity. We describe how such cases are handled in Tab. 12. In the left column, we describe the case and how we handle it. In the right column, we provide a visualization. We also handle these cases in a similar fashion if the curves $|\gamma_r| = |\gamma_d|$ or $\gamma_r = 0$ intersects any vertex.

H.2.4 Degenerate Conic Sections

A conic section is typically a circle, ellipse, parabola, or hyperbola. However, it can also take degenerate forms, such as a single line, two parallel lines, two intersecting lines, or a single point.

If one of the conic sections, $\gamma_d^2 = \gamma_s^2$ or $\gamma_r^2 = \gamma_s^2$, is degenerate, it can cause issues when computing the invariant. Unless the conic is two parallel lines, our algorithm's output may be affected.

For a single point, TFZ can be influenced because, although the point has no topological impact, the algorithm still detects its presence. For a single line, problems may arise if $\gamma_d = \gamma_s$ forms a line but $\gamma_s > |\gamma_d|$ holds on both sides, leading TFZ to misinterpret the topology. A similar issue can occur with the curve $\gamma_r = \gamma_s$. When the conic consists of two intersecting lines, our algorithm struggles because the intersection point alters the cell topology, yet it is neither a junction point nor an intersection with the cell boundary, making it difficult to handle directly.

We describe how we handle these cases in Tab. 13. In the left column, we describe each case and how it is handled. In the right column, we provide an illustration.

H.3 Numerical Precision

The finite precision of floating-point values can cause difficulties for TFZ in certain cases. To mitigate these issues, we introduce several fixes. Admittedly, TFZ may still make mistakes in extreme scenarios, in which case it might be better implemented using integer representations of floating-point numbers.

Close values. We consider two numbers x and y to be equal if $|x - y| < 10^{-10}\max(|x|, |y|)$. Similarly, we consider x to be greater than y if $x - y > 10^{-10}\max(|x|, |y|)$ and consider x to be less than y if $x - y < -10^{-10}\max(|x|, |y|)$.

When computing the classification according to the eigenvector manifold, we consider γ_r to be equal to zero if $|\gamma_r| < 10^{-10}$. Similarly, if the number u is the dot product of two unit vectors, the output of a trigonometric equation, or the x or y position of a point in the unit square, we consider u to be equal to zero if $|u| < 10^{-10}$.

Tensor normalization. When computing the topology of a cell σ with tensors T_1, T_2 and T_3 at the vertices, we compute:

$$x = \min\{|y| : y \text{ is an entry of } T_1, T_2 \text{ or } T_3 \text{ and } |y| > 10^{-10}\}.$$

Set $T_1' \leftarrow \frac{1}{|x|}T_1$, $T_2' \leftarrow \frac{1}{|x|}T_2'$ and $T_3 \leftarrow \frac{1}{|x|}T_3'$. We then proceed to compute the topology of σ using T_1', T_2', and T_3'.

Quadratic formula stability. We have found that the quadratic formula can become unstable with very large or small values. Suppose that we are computing the roots of $ax^2 + bx + c$. First, we compute $k = \max(|a|, |b|, |c|)$. If $k = 0$, then there are infinitely many solutions. Otherwise, we set $a' \leftarrow \frac{a}{k}$, $b' \leftarrow \frac{b}{k}$ and $c' \leftarrow \frac{c}{k}$. Then, if any of a', b', or c' has a magnitude less than 10^{-10}, we set it equal to zero. We then compute the roots with the standard quadratic formula using a', b' and c'.

When computing the discriminant $(b')^2 - 4a'c'$, if $\frac{|(b')^2-4a'c'|}{(|a'|+|b'|+|c'|)^2} < 10^{-10}$ (i.e., the discriminant is very small compared to the largest coefficient) then we treat it as being equal to zero.

Relative error bound. After decompressing the data, it is possible that some entries of a tensor will be very large or small in magnitude compared to one another. For example, it is possible one entry will be 10^{-8} while another is 10^{-2}. Such variability can exacerbate numerical precision issues. To resolve this issue, we enforce a relative error bound of 20 on all entries of a given tensor. That is, if x is an entry of a tensor, and x' is a guess for x, if $x' \neq 0$ and $x \neq 0$, we require that $\frac{|x|}{|x'|} < 20$ and $\frac{|x'|}{|x|} < 20$.

I Proofs of Theoretical Results

In this section, we prove most of the lemmas from the background section. The proofs of Lemma 6 and Lemma 7 are substantially more involved, and are therefore given separately in Appendix J. We begin with several supporting lemmas before proving the main lemmas from Sec. 4. With a slight abuse of notation, we write $M = 0$ for a matrix M if all of its entries are zero.

Lemma 8. *Let M be a symmetric matrix. Then M is degenerate if and only if $D(M) = 0$.*

Proof. According to [10], the eigenvalues of M are $\gamma_d \pm \gamma_s$. Thus, M is degenerate if and only if $\gamma_s = 0$. Following Eq. (5), we have

$$D(M) = \gamma_s \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ \sin(\theta) & -\cos(\theta) \end{pmatrix}.$$

Thus, $D(M) = 0$ if and only if $\gamma_s = 0$, which occurs if and only if M is degenerate. □

Lemma 9. *Let M_1 and M_2 be two symmetric matrices such that $D(M_1) \neq 0$ and $D(M_2) \neq 0$. Then $l_{1,2} = 0$ if and only if there exists some $k \in \mathbb{R}$ such that $D(M_1) = kD(M_2)$.*

Proof. Denote the entries of $D(M_1)$ by

$$D(M_1) = \begin{pmatrix} \Delta_1 & F_1 \\ F_1 & -\Delta_1 \end{pmatrix}$$

Denote the entries of $D(M_2)$ similarly.

First, suppose that $D(M_1) = kD(M_2)$. Then $\Delta_1 = k\Delta_2$ and $F_1 = kF_2$. Thus, $F_2\Delta_1 - F_1\Delta_2 = F_2(k\Delta_2) - (kF_2)\Delta_2 = 0$, so $l_{1,2} = 0$.

Now suppose that $l_{1,2} = 0$. Then $F_2\Delta_1 - F_1\Delta_2 = 0$. We check two cases:

Case 1: $\Delta_2 \neq 0$. Set $k = \frac{\Delta_1}{\Delta_2}$. Notice that $F_1\Delta_2 = F_2\Delta_1$ meaning that $F_1 = \frac{\Delta_1}{\Delta_2}F_2 = kF_2$. And clearly, $\Delta_1 = \frac{\Delta_1}{\Delta_2}\Delta_2 = k\Delta_2$. Thus, $D(M_1) = kD(M_2)$

Case 2: $\Delta_2 = 0$. Since $D(M_2) \neq 0$ and $\Delta_2 = 0$, we must have $F_2 \neq 0$. Thus, set $k = \frac{F_1}{F_2}$ and proceed similarly to case 1. □

Lemma 10. *Suppose that $x, y \in \mathbb{R}$ and $x', y' \in \mathbb{R}$ are respectively guesses for x and y. Let ξ be an error bound such that $|x - x'| \leq \xi$ and $|y - y'| \leq \xi$. Then (a) $\left|\frac{x+y}{2} - \frac{x'+y'}{2}\right| \leq \xi$ (b) $\left|\sqrt{x^2+y^2} - \sqrt{(x')^2+(y')^2}\right| \leq \xi\sqrt{2}$*

Proof. This can be verified with simple algebra. □

Lemma 1. *Let T_1 and T_2 be two 2×2 symmetric tensors with nonzero deviators. Let $l_{1,2}$ be defined following Eq. (2). Then $l_{1,2}$ depends only on θ_1 and θ_2, where each θ_i is taken from the decomposition in Eq. (5).*

Proof. Let γ_{d_1}, $\gamma_{s,1}$ and θ_1 be the coefficients from decomposing M_1 according to Eq. (5). Let $\gamma_{d,2}$, $\gamma_{s,2}$ and θ_2 be the coefficients from decomposing M_2. The deviator of M_1 is:

$$D(M_1) = \gamma_{s,1} \begin{pmatrix} \cos(\theta_1) & \sin(\theta_1) \\ \sin(\theta_1) & -\cos(\theta_1) \end{pmatrix}$$

Let Δ_1 and F_1 be the entries of $D(M_1)$. Then $\Delta_1 = \gamma_{s,1}\cos(\theta_1)$ and $F_1 = \gamma_{d,1}\sin(\theta_1)$. Similarly, we have $\Delta_2 = \gamma_{s,2}\cos(\theta_2)$ and $F_2 = \gamma_{s,2}\sin(\theta_2)$.

Now observe that:

$$\begin{aligned} l_{1,2} &= \Delta_2 F_1 - \Delta_1 F_2 \\ &= (\gamma_{s,2}\cos(\theta_2))(\gamma_{s,1}\sin(\theta_1)) - (\gamma_{s,1}\cos(\theta_1))(\gamma_{s,2}\sin(\theta_2)) \\ &= \gamma_{s,1}\gamma_{s,2}(\cos(\theta_2)\sin(\theta_1) - \cos(\theta_1)\sin(\theta_2)) \end{aligned}$$

By definition, $\gamma_{s,1}$ and $\gamma_{s,2}$ are both nonnegative. By assumption, both deviators are nonzero, meaning that $\gamma_{s,1} \neq 0$ and $\gamma_{s,2} \neq 0$. Thus, the sign of $l_{1,2}$ only depends on θ_1 and θ_2. □

Lemma 2. *Let x_1, x_2, and x_3 be the vertices of a cell.*

(a) Suppose that exactly one vertex has a tensor T with a deviator $D(T)$ equal to zero. (w.l.o.g., suppose $D(f(x_1)) = 0$).

(i) If $l_{2,3} \neq 0$, then x_1 is the only degenerate point in the cell.

(ii) If $l_{2,3} = 0$, then there exists some k such that $D(f(x_2)) = kD(f(x_3))$. If $k > 0$ then x_1 is the only degenerate point in the cell. If $k < 0$ then the cell contains a degenerate line.

(b) If exactly two vertices have a tensor with a deviator equal to zero, then the cell contains a degenerate line connecting them.

Proof. Denote the cell as σ. We first show (a). Suppose that there exists some $x \in \sigma$ with $x \neq x_1$ but $D(x) = 0$. There exists $t_1, t_2, t_3 \in [0, 1]$ such that $x = t_1 f(x_1) + t_2 f(x_2) + t_3 f(x_3)$. Because the deviator is a linear operator, $D(f(x)) = t_1 D(f(x_1)) + t_2 D(f(x_2)) + t_3 D(f(x_3))$.

Since $D(f(x)) = 0$, and $D(f(x_1)) = 0$, but $D(f(x_2)) \neq 0$, x cannot lie on the edge between x_1 and x_2. Similarly, x cannot lie on the edge between x_1 and x_3. Thus, $t_2 \neq 0$ and $t_3 \neq 0$.

Because $D(f(x)) = t_2 D(f(x_2)) + t_3 D(f(x_3)) = 0$, it follows that $D(f(x_2)) = \frac{-t_3}{t_2} D(f(x_3))$. So there exists k such that $D(f(x_2)) = kD(f(x_3))$. By Lemma 9, this implies that $l_{2,3} = 0$, proving (a) (i).

For (a) (ii), Lemma 9 tells us that if $l_{2,3} = 0$ there exists k such that $D(f(x_2)) = kD(f(x_3))$. We just showed that if there exists some internal point $x \in C$ with $D(x) = 0$, then $D(x_2) = kD(x_3)$ with $k < 0$.

To finish (a) (ii), we must show that if $k < 0$ then there exists some $y \neq x_1$ such that $D(f(y)) = 0$. Suppose that $k < 0$ and let $t_1 = 0$, let $t_2 = \frac{1}{1-k}$ and $t_3 = \frac{-k}{1-k}$. Then $t_1, t_2, t_3 \in (0, 1)$ and $t_1 + t_2 + t_3 = 1$. Let $y = t_1 x_1 + t_2 x_2 + t_3 x_3$, so $y \in \sigma$. Similar to before, $D(f(x)) = t_2 D(f(x_2)) + t_3 D(f(x_3))$.

Notice that $t_2 D(f(x_2)) + t_3 D(f(x_3)) = \frac{1}{1-k} D(f(x_2)) + \frac{-k}{1-k} D(f(x_3)) = \frac{1}{1-k}(D(f(x_2)) - kD(f(x_3))) = 0$, so y is degenerate. This finishes the proof for (a) (ii).

To prove (b), note that since the two degenerate vertices have a deviator equal to zero and the deviator is a linear operator, all tensors interpolated between them will also be degenerate. □

Lemma 3. *Suppose T' and T are 2×2 tensors, where each entry of T' differs from the corresponding entry of T by at most ξ, i.e., $|T_{1,1} - T'_{1,1}| \leq \xi$, and so on. Denote their coefficients from the decomposition as γ_d, γ_r, γ_s, γ'_d, γ'_r, and γ'_s, respectively. Then $|\gamma_d - \gamma'_d| \leq \xi$, $|\gamma_r - \gamma'_r| \leq \xi$, and $|\gamma_s - \gamma'_s| \leq \sqrt{2}\xi$.*

Proof. This follows from Lemma 10. □

Lemma 4. *Suppose that $x \in \mathbb{R}$ and x' is a guess of x within an error bound ξ, i.e., $|x - x'| \leq \xi$.*

- *If $x > 0$ but $x' < 0$. Then $x' + \xi > 0$ and $|x - (x' + \xi)| \le \xi$. That is, $x' + \xi$ is a valid guess for x that is within an error bound ξ and has the same sign as x.*
- *If $x < 0$ but $x' > 0$, then $x' - \xi$ is analogously valid.*

Proof. Let $x > 0$ and $x' < 0$. Then we have $|x - x'| = x - x'$. Thus, $x - x' < \xi$, so $x' + \xi > x > 0$.

Since $(x + \xi) > x$, it follows that $|x - (x' + \xi)| = (x' + \xi) - x = \xi + (x' - x)$. Because $x' - x < 0$, it follows that $\xi + (x' - x) < \xi$. Putting these inequalities together yields $|x - (x' + \xi)| < \xi$.

The proof for the case where $x < 0$ is similar. □

Lemma 5. *Suppose that $x, y \in \mathbb{R}$ and x' and y' are guesses for x and y, respectively, within an error bound ξ, such that $|x - x'| \le \xi$ and $|y - y'| \le \xi$. Suppose that $|x| > |y|$ but $|x'| < |y'|$.*

Let $x'' = sign(x')|y'|$, $y'' = sign(y')|x'|$. Then $|x - x''| \le \xi$ and $|y - y''| \le \xi$, and $|x''| > |y''|$. That is, by swapping the magnitudes of x' and y', we can obtain two valid guesses x'' and y'' with $|x''| > |y''|$.

Proof. We show that $|x - y''| \le \xi$. The proof that $|y - x''| \le \xi$ is similar. Recall that $||x| - |x'|| \le |x - x'|$ so $||x| - |x'|| \le \xi$. Similarly, $||y| - |y'|| \le \xi$. We check two cases for $|x|$:

Case 1: $|y'| \le |x|$: Then $|x'| < |y'| \le |x|$. Since $||x| - |x'|| \le \xi$, it must follow that $||x| - |y'|| \le \xi$.

Case 2: $|y'| > |x|$: Then $|y| < |x| < |y'|$. Since $||y| - |y'|| \le \xi$, it must follow that $||x| - |y'|| \le \xi$.

In either case, it follows that $||x| - |y'|| \le \xi$. If $x > 0$, then $x = |x|$ and $y'' = |y'|$, so $|x - y''| \le \xi$. If $x < 0$, the proof is similar. □

J Correctness Proofs of Topological Invariant

In this section, we build toward the proofs of Lemma 6 and Lemma 7. We provide definitions and assumptions in Appendix J.1. We prove preliminary lemmas in Appendix J.2. We prove Lemma 6 and Lemma 7 in Appendix J.3. Finally, we prove how we handle edge cases in Appendix J.4. For the eigenvector partition, we ignore degenerate points of the dual eigenvector field, and only focus on partition regions.

J.1 Definitions and Assumptions

In this section, we specify all of the notations and assumptions that we will be using for the remainder of the section.

Function Definitions. Let $\sigma \subset \mathbb{R}^2$ be a 2D triangular cell, and let $f : \sigma \to \mathbb{T}$ be a PL tensor field on σ. With an abuse of notation, define the function $\gamma_d : \sigma \to \mathbb{R}$ as the function that maps each point $p \in \sigma$ to the value of γ_d at p. Define similar functions for γ_r, γ_s, and θ.

Let $c : \sigma \to \mathbb{R}$ and $s : \sigma \to \mathbb{R}$ be defined by

$$c(p) = \gamma_d(p)\cos(\theta(p)) \qquad s(p) = \gamma_d(p)\sin(\theta(p))$$

Then notice that

$$\gamma_s(p) = \sqrt{c(p)^2 + s(p)^2}$$

Since f is PL, all of its coordinate functions are affine in $\sigma \subset \mathbb{R}^2$. Thus, f can be extended to all of $\mathbb{R}^2$. All of the other functions that we have just defined can thus also be extended to $\mathbb{R}^2$.

Subsets and Regions. We first define some subsets of σ.

Eigenvalue partition: Let

$$D_+ = \{p \in \sigma : \gamma_d > |\gamma_r| \text{ and } \gamma_d > \gamma_s\}$$
$$D_- = \{p \in \sigma : -\gamma_d > |\gamma_r| \text{ and } -\gamma_d > \gamma_s\}$$

Define R_+, and R_- similarly. Let S be the region where $\gamma_s > |\gamma_d|$ and $\gamma_s > |\gamma_r|$.

Eigenvector partition: Define r_{r+} as the subset of σ where $\gamma_r > \gamma_s$ and r_{r-} as the subset where $-\gamma_r > \gamma_s$. Define s_{r+} as the subset where $\gamma_s > \gamma_r > 0$ and s_{r-} as the subset where $\gamma_s > -\gamma_r > 0$.

Other subsets: While not part of either partition, let d_{d+} be the subset of σ where $\gamma_d > \gamma_s$ and d_{d-} be the subset where $-\gamma_d > \gamma_s$. Notice that, by definition,

- $D_+ \subset d_{d+}$
- $D_- \subset d_{d-}$

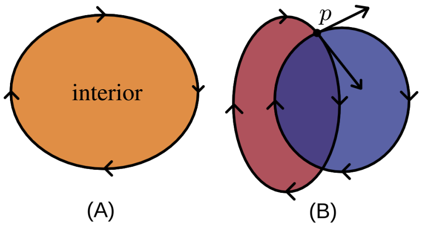


Fig. 20: (A) A connected region with the interior shaded. Arrows denote clockwise orientation. (B) Two connected regions overlap. At point p, they intersect. Here, the curve bounding the red region enters the blue region. The curve bounding the blue region leaves the red region.

- $R_+ \subset r_{r+}$
- $R_- \subset r_{r-}$.

We define a *region* of σ as a connected subset of σ. We say that a region R is of *type* D_+ if $R \subset D_+$. Extend this defintition of type to the other subsets that we have defined. We say that two regions *border* each other if their boundaries intersect.

Topologically Significant Intersections. If e is an edge of σ, and p lies on the intersection of e with the curve $\gamma_d = \gamma_s$, say that p is *topologically significant* if it lies on the boundary between D_+ and S. Define topological significance analogously for $-\gamma_d = \gamma_s$, $\gamma_r = \gamma_s$, and $-\gamma_r = \gamma_s$.

We say that an intersection between $\gamma_d^2 = \gamma_s^2$ and e is topologically significant if it separates a region of type S from a region of either type D_+ or D_-. Define topologically significant intersections analogously for $\gamma_r^2 = \gamma_s^2$.

Edge Cases. For now, we make the following assumptions in order to avoid edge cases. Some of these cases are avoided by TFZ using a symbolic perturbation (see Appendix H), and thus we can assume that they never occur. Other cases can occur. In Appendix J.4, we demonstrate that when such cases occur that our strategy still works. We assume that:

(i) $\gamma_d^2 = \gamma_s^2$ and $\gamma_r^2 = \gamma_s^2$ only intersect each other and each edge of σ transversally.
(ii) No junction points occur on the edges of σ.
(iii) No two of the functions $|\gamma_d|$, $|\gamma_r|$, or γ_s are exactly equal on all of $\mathbb{R}^2$.
(iv) The conic sections $\gamma_d^2 = \gamma_s^2$ and $\gamma_r^2 = \gamma_s^2$ are either empty or have infinitely many points.
(v) The curve $\gamma_d = \gamma_s$ will border one region of type d_{d+} and another where $\gamma_s > |\gamma_d|$. Make similar assumptions about $-\gamma_d = \gamma_s$, $\gamma_r = \gamma_s$, and $-\gamma_r = \gamma_s$.
(vi) $|\gamma_d|$, $|\gamma_r|$ and γ_s are never equal on a vertex of σ.
(vii) There are no vertices of σ where $\gamma_r = 0$.
(viii) There are no points in σ where $\gamma_d = \gamma_s = 0$ or $\gamma_r = \gamma_s = 0$.

Clockwise Orientation. Suppose that c is a closed curve that divides the plane into two regions. Suppose that one of them is labeled as the "interior." Then we define *clockwise orientation* or c to be such that, when traveling along c, the "interior" of c is on the right of c. We demonstrate this orientation in Fig. 20(A).

If a curve c divides the plane into two regions R_1 and R_2, and c is said to be clockwise oriented, then label one of R_1 or R_2 to be the interior of c consistent with the previous definition.

Suppose that c_1 and c_2 are two clockwise oriented curves. Let p be a point where they intersect transversally. We say that c_1 *enters* the interior of c_2 at p if the oriented vector tangent to c_1 at p points into the interior of c_2. In Fig. 20(B), we illustrate the case where one curve enters another. At point p, the curve bounding the red region enters the blue region, while the curve bounding the blue region leaves the red region.

We assume that $\gamma_r = \gamma_s$ separates a region of type d_{d+} from a region where $\gamma_s > |\gamma_d|$. In this case, we denote that d_{d+} is the interior of $\gamma_d = \gamma_s$, and orient $\gamma_d = \gamma_s$ using clockwise orientation. We proceed similarly for the curves $-\gamma_d = \gamma_s$, $\gamma_r = \gamma_s$, and $-\gamma_r = \gamma_s$.

J.2 Supporting Lemmas

In order to prove Lemma 6 and Lemma 7, we first prove some intermediate results.

Lemma 11. *The functions γ_d and γ_r, c and s are affine over $\sigma \subset \mathbb{R}^2$.*

Proof. For $p \in \sigma$, recall that

- $\gamma_d(p) = \frac{1}{2}(f(p)_{1,1} + f(p)_{2,2})$
- $\gamma_r(p) = \frac{1}{2}(f(p)_{2,1} - f(p)_{1,2})$
- $c(p) = \frac{1}{2}(f(p)_{1,1} - f(p)_{2,2})$
- $s(p) = \frac{1}{2}(f(p)_{1,2} + f(p)_{2,1})$

Because each $f_{i,j}$ is affine, so are the functions above. □

Lemma 12. *The function γ_s is convex.*

Proof. Fix $p_1, p_2 \in \sigma$. Denote $\theta_1 := \theta(p_1)$ and $\theta_2 := \theta(p_2)$. Notice that

$$\begin{aligned} & c(p_1)c(p_2) + s(p_1)s(p_2) \\ &= \gamma_s(p_1)\cos(\theta_1)\gamma_s(p_2)\cos(\theta_2) + \gamma_s(p_1)\sin(\theta_1)\gamma_s(p_2)\sin(\theta_2) \\ &= \gamma_s(p_1)\gamma_s(p_2)(\cos(\theta_1)\cos(\theta_2) + \sin(\theta_1)\sin(\theta_2)) \\ &= \gamma_s(p_1)\gamma_s(p_2)\cos(\theta_1 - \theta_2) \\ &\le \gamma_s(p_1)\gamma_s(p_2) \end{aligned}$$

Thus, $c(p_1)c(p_2) + s(p_1)s(p_2) \le \gamma_s(p_1)\gamma_s(p_2)$. Using this fact, let $t \in [0, 1]$ and observe that

$$\begin{aligned} & \gamma_s(tp_1 + (1-t)p_2)^2 \\ &= c(tp_1 + (1-t)p_2)^2 + s(tp_1 + (1-t)p_2)^2 \\ &= (tc(p_1) + (1-t)c(p_2))^2 + (ts(p_1) + (1-t)s(p_2))^2 \\ &= t^2(c(p_1)^2 + s(p_1)^2) + 2t(1-t)(c(p_1)c(p_2) + s(p_1)s(p_2)) \\ &\quad + (1-t)^2(c(p_2)^2 + s(p_2)^2) \\ &= t^2\gamma_s(p_1)^2 + 2t(1-t)(c(p_1)c(p_2) + s(p_1)s(p_2)) \\ &\quad + (1-t)^2\gamma_s(p_2)^2 \\ &\le t^2\gamma_s(p_1)^2 + 2t(1-t)\gamma_s(p_1)\gamma_s(p_2) + (1-t)^2\gamma_s(p_2) \\ &= (t\gamma_s(p_1) + (1-t)\gamma_s(p_2))^2 \end{aligned}$$

Thus, $\gamma_s(tp_1 + (1-t)p_2)^2 \le (t\gamma_s(p_1) + (1-t)\gamma_s(p_2))^2$. Because γ_s is nonnegative, this implies that $\gamma_s(tp_1 + (1-t)p_2) \le t\gamma_s(p_1) + (1-t)\gamma_s(p_2)$. Thus, γ_s is convex. □

Lemma 13. *D_+, D_-, R_+, R_-, r_{r+}, r_{r-}, d_{d+}, and d_{d-} are convex.*

Proof. We provide a proof for D_+. The other region types follow similarly. Let $p_1, p_2 \in D_+$. Let p_3 lie on the segment between p_1 and p_2. Since γ_d is affine, and $|\gamma_r|$ and γ_s are convex, it must hold that $\gamma_d(p) > |\gamma_r(p)|$ and $\gamma_d(p) > \gamma_s(p)$. Thus, $p \in D_+$, so D_+ is convex. □

Corollary 1. *Each of D_+, D_-, R_+, R_-, r_{r+}, r_{r-}, d_{d+}, and d_{d-} have only one connected component each.*

Because each of the subsets in Corollary 1 have one connected component, we may refer to them hereafter as regions.

Lemma 14. *The following pairs of regions cannot border each other:*

(i) d_{d+} cannot border d_{d-}
(ii) r_{r+} cannot border r_{r-}
(iii) D_+ cannot border D_-
(iv) R_+ cannot border R_-

Proof. We show (i) and (iii), where (ii) and (iv) follow similarly. Suppose, to the contrary, that d_{d+} bordered d_{d-}. Let x be a point that separates them. Since $\gamma_d > 0$ one one side of x, and $\gamma_d < 0$ on the other, it must follow that $\gamma_d(x) = 0$. Since $|\gamma_d| > \gamma_s$ for all points nearby to x, it must follow that $|\gamma_d(x)| \ge \gamma_s(x)$, implying that $\gamma_d(x) = \gamma_s(x) = 0$. However, we previously assumed that this is impossible. This gives (i).

Because (i) is true and $D_+ \subset d_{d+}$ and $D_- \subset d_{d-}$, (iii) must follow. □

Lemma 15. *r_{r+} cannot border s_{r-} and r_{r-} cannot border s_{r+}.*

Proof. The proof is similar to that of Lemma 14. □

Lemma 16. *The following statements are true:*

(1) If $\gamma_d^2 = \gamma_s^2$ is an ellipse contained entirely within σ, then either $d_{d+} = \emptyset$ or $d_{d-} = \emptyset$.
(2) If $\gamma_d = \gamma_s$ is an ellipse contained within σ, then $d_{d-} = \emptyset$. If $-\gamma_d = \gamma_s$ is an ellipse contained within σ, then $d_{d+} = \emptyset$.

Analogous claims are true about $\gamma_r^2 = \gamma_s^2$.

Proof. We prove the claims for $\gamma_d^2 = \gamma_s^2$. Notice that because there are no points where $\gamma_d = \gamma_s = 0$, the curves $\gamma_d = \gamma_s$ and $-\gamma_d = \gamma_s$ must be disjoint.

We first show (1). The curve $\gamma_d^2 = \gamma_s^2$ must border d_{d+} or d_{d-}, so one of those sets is not empty. Without loss of generality, suppose that $d_{d+} \ne \emptyset$. Since $d_{d+} \ne \emptyset$, the region d_{d+} must have, as its boundary, the curve $\gamma_d = \gamma_s$. So the curve $\gamma_d = \gamma_s$ is not empty.

Notice that $\gamma_d^2 = \gamma_s^2$ will be the union of the curves $\gamma_d = \gamma_s$ and $-\gamma_d = \gamma_s$. Since $\gamma_d^2 = \gamma_s^2$ is an ellipse, it has one connected component. Since $-\gamma_d = \gamma_s$ cannot intersect $\gamma_d = \gamma_s$, and $\gamma_d = \gamma_s$ is not empty, it must follow that $-\gamma_d = \gamma_s$ is empty.

If $d_{d-} \ne \emptyset$, then it would need to be bounded by the non-empty curve $-\gamma_d = \gamma_s$. Thus, $d_{d-} = \emptyset$. This gives (1).

We now show (2). Without loss of generality, suppose that $\gamma_d = \gamma_s$ is an ellipse contained within σ. The conic section $\gamma_d^2 = \gamma_s^2$ is a conic section that contains the curve $\gamma_d = \gamma_s$ and the curve $-\gamma_d = \gamma_s$. Thus, $\gamma_d^2 = \gamma_s^2$ contains an ellipse. The only possible conic section that $\gamma_d^2 = \gamma_s^2$ could be, then, is an ellipse. Invoking (1) gives $d_{d-} = \emptyset$. □

Lemma 17. *If the conic section $\gamma_d^2 = \gamma_s^2$ is an ellipse contained entirely within σ, then the interior of $\gamma_d^2 = \gamma_s^2$ must intersect r_{r+} or r_{r-}. The analogous claim for $\gamma_r^2 = \gamma_s^2$ is also true.*

Proof. Extend f to $\mathbb{R}^2$. The conic section $\gamma_d^2 = \gamma_s^2$ is the union of the curve $\gamma_d = \gamma_s$ and $-\gamma_d = \gamma_s$. Since there is no point where $\gamma_d = \gamma_s = 0$, these curves do not intersect. Since $\gamma_d^2 = \gamma_s^2$ has only one connected component, it must be equal to $\gamma_d = \gamma_s$ or $-\gamma_d = \gamma_s$. Without loss of generality, assume that the conic section $\gamma_d^2 = \gamma_s^2$ is equal to the curve $\gamma_d = \gamma_s$.

By assumption, the curve $\gamma_d = \gamma_s$ separates a region of type d_{d+} from a region where $\gamma_s > |\gamma_d|$. Because d_{d+} is convex, and $\gamma_d = \gamma_s$ is an ellipse, d_{d+} must lie in the interior of the ellipse.

We claim that the gradient directions of c and s are linearly independent. Suppose, to the contrary, that they were not. Then there exists some unit vector v perpendicular to the gradients of c and s. Since c and s are affine, they both have constant gradients. Thus, c and s are constant in direction v, and thus so is γ_s.

Let $p \in d_{d+}$. Let l be the line through p in direction v. Since γ_d is affine, we can move along line l in one direction w (either $w = v$ or $w = -v$) so that γ_d does not decrease. Also, γ_s is constant along l. Thus, the ray in direction w rooted at p satisfies $\gamma_d > \gamma_s$, and the entire ray belongs to d_{d+}. This violates the assumption that d_{d+} is contained within the ellipse.

Thus, the gradient directions of c and s are independent. So the curves $c(x, y) = 0$ and $s(x, y) = 0$ are two lines that are not parallel. Thus, they will intersect, and there is a point q where $c(q) = s(q) = \gamma_s(q) = 0$. By Lemma 16, $d_{d-} = \emptyset$ so $\gamma_d(q) \ge 0$. By assumption , $\gamma_d(q) \ne \gamma_s(q) = 0$. Hence, $\gamma_d(q) > \gamma_s(q) = 0$, so $q \in d_{d+}$ and q lies in the interior of the ellipse $\gamma_d^2 = \gamma_s^2$.

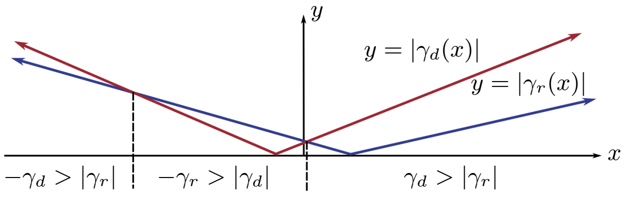


Fig. 21: The two functions $|\gamma_d|$ and $\gamma_r|$ can intersect at most twice. Further, every region where $|\gamma_d| > |\gamma_r|$ will border every region where $|\gamma_r| > |\gamma_d$

Trivially, $|\gamma_r(q)| > \gamma_s(q)$ so either $q \in r_{r+}$ or $q \in r_{r-}$. In either case, the claim is proven. □

Lemma 18. *Let e be a line segment. If $d_{d+} \cap e \neq \emptyset$ and $d_{d-} \cap e \neq \emptyset$, then one vertex of e is of type d_{d+} and the other is of type d_{d-}. The analogous claim for r_{r+} and r_{r-} is also true.*

Proof. Extend f to $\mathbb{R}^2$. Let $\phi : \mathbb{R} \to \mathbb{R}^2$ parametrize the line containing e, such that $\phi([0,1]) = e$. One can verify that the functions $(\gamma_d \circ \phi)^2$ and $(\gamma_s \circ \phi)^2$ are quadratic functions $[0,1] \to \mathbb{R}$.

Since $r_{r+} \cap e$ cannot border $r_{r-} \cap e$, there must be some interval $I := (t_1, t_2) \in [0,1]$ such that $\phi(I)$ lies between $r_{r+} \cap e$ and $r_{r-} \cap e$, and $\gamma_s \circ \phi > |\gamma_d \circ \phi|$ on all of I. Further, $(\gamma_d \circ \phi)^2$ and $(\gamma_s \circ \phi)^2$ must intersect at t_1 and t_2. Since those functions are quadratic, they can only intersect at most twice.

Thus, outside of I, there will be no other points where $|\gamma_d| = \gamma_s$. This implies that the only region of e where $\gamma_s > |\gamma_d|$ that borders $d_{d+} \cap e$ or $d_{d-} \cap e$ is $\phi(I)$, which lies between d_{d+} and d_{d-}. So one vertex of e must lie within d_{d+} and the other within d_{d-}. □

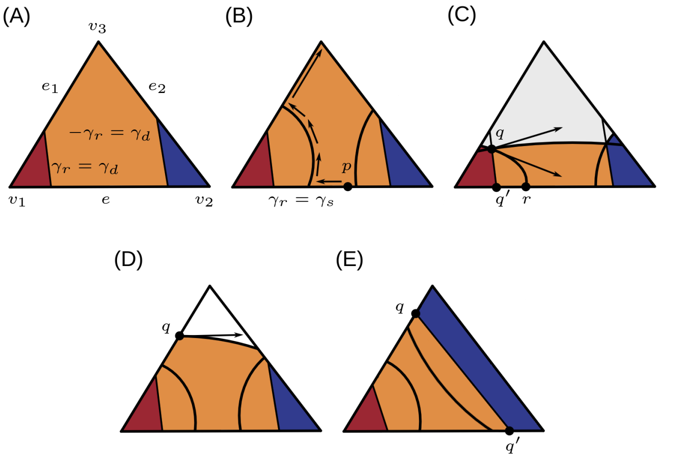


Fig. 22: Sample eigenvector partitions of σ as used in Lemma 20. The regions are Orange: D_+. Red: R_+. Blue: R_-. White S. (A) We label the edges and vertices of σ, along with the line $\gamma_r = \gamma_d$, which separates D_+ from R_+, and the line $-\gamma_r = \gamma_d$, which separates D_+ from R_-. (B) We demonstrate the path from p up to v_3. The path goes towards v_2, then up the curve $\gamma_r = \gamma_s$ until hitting e_1. It then continues from e_1 to v_3. In this case, $v_3 \in D_+$. (C) Here, q is a junction point between $\gamma_r = \gamma_s$ and $\gamma_d = \gamma_s$. We label the point r where $\gamma_r = \gamma_s$ intersects e. We also label the oriented tangent vectors of each curve at point p. Here $\gamma_r = \gamma_s$ enters D_+. (D) Here q is a topologically significant intersection point between $\gamma_d = \gamma_s$ and e_1. We show the oriented tangent vector of $\gamma_d = \gamma_s$ at q, where it points into σ. (E) Here q separates D_+ from R_-, and lies on the line $-\gamma_r = \gamma_d$. We also label the point q' used in the proof.

Lemma 19. *Let e be a line segment. Let $E_1 = \{p \in e : \gamma_d(p) > |\gamma_r(p)|\}$ and $E_2 = \{p \in e : \gamma_r(p) > |\gamma_d(p)|\}$. If E_1 and E_2 are not empty, then they will border each other. The analogous claims using $-\gamma_d$ or $-\gamma_r$ are also true.*

Proof. Let $\phi : [0,1] \to e$ parametrize e. Notice that the functions $|\gamma_d \circ \phi|$ and $|\gamma_r \circ \phi|$ are the absolute values of affine functions on $[0,1]$. Thus, every region where $|\gamma_d \circ \phi| \geq |\gamma_r \circ \phi|$ will border every region where $|\gamma_d \circ \phi| \leq |\gamma_r \circ \phi|$. We illustrate an example in Fig. 21. □

Corollary 2. *Let e be a line segment. Then at least one of the following sets must be empty*

- $\{p \in e : \gamma_d(p) > |\gamma_r(p)|\}$
- $\{p \in e : -\gamma_d(p) > |\gamma_r(p)|\}$
- $\{p \in e : \gamma_r(p) > |\gamma_d(p)|\}$
- $\{p \in e : -\gamma_r(p) > |\gamma_d(p)|\}$

Corollary 3. *The regions D_+, D_-, R_+ and R_- cannot all intersect a segment e.*

Lemma 20. *Let e be an edge of σ with vertices v_1 and v_2. Suppose that v_1 comes after v_2 when traversing σ clockwise. Let v_3 be the remaining vertex of σ. Let e_1 be the edge connecting v_1 to v_3, and let e_2 be the edge connecting v_2 to v_3. (See Fig. 22(A) for a diagram.)*

Suppose that D_+ is not empty, yet does not contain either v_1 or v_2, and the curve $\gamma_d = \gamma_s$ does not have any topologically significant intersections with e. Then D_+ will intersect edge e if and only if the following are true:

(a) One of v_1 or v_2 is of type R_+ and the other is R_-. (assume, w.l.o.g. that $v_1 \in R_+$).
(b) $\gamma_r^2 = \gamma_s^2$ has no topologically significant intersections with e
(c) At least one of the following is true:
(i) The curve $\gamma_d = \gamma_s$ never intersects e_1 or $\gamma_r = \gamma_s$ and v_3 is of type D_+.
(ii) The curve $\gamma_d = \gamma_s$ never intersects e_2 or $-\gamma_r = \gamma_s$ and v_3 is of type D_+.
(iii) There is exactly one junction point p between $\gamma_d = \gamma_s$ and $\gamma_r = \gamma_s$. At that junction point, $\gamma_r = \gamma_s$ enters d_{d+}.
(iv) There is exactly one junction point p between $\gamma_d = \gamma_s$ and $-\gamma_r = \gamma_s$. At that junction point, $-\gamma_r = \gamma_s$ leaves d_{d+}.
(v) The curve $\gamma_d = \gamma_s$ has exactly one topologically significant intersection with e_1, where it enters σ. $\gamma_d = \gamma_s$ never intersects $\gamma_r = \gamma_s$.
(vi) The curve $\gamma_d = \gamma_s$ has exactly one topologically significant intersection with e_2 where it leaves σ. $\gamma_d = \gamma_s$ never intersects $-\gamma_r = \gamma_s$.
(vii) The curve $\gamma_d = \gamma_s$ has no junction points or topologically significant intersections with any edge of σ. v_3 is of type R_-.

Proof. Suppose that D_+ borders e but $\gamma_d = \gamma_s$ has no topologically significant intersections with e, and neither vertex of e is D_+. We first show that if D_+ borders e, then (a) (b) and (c) must be true.

Because neither vertex is of type D_+, the region $D_+ \cap e$ must be an interval in the interior of e bordered by a different region on either side. $D_+ \cap e$ cannot border $D_- \cap e$. Because $\gamma_d = \gamma_s$ has no topologically significant intersections with e, $D_+ \cap e$ cannot border $S \cap e$. Thus, $D_+ \cap e$ can only border $R_+ \cap e$ and $R_- \cap e$.

Because R_+ and R_- are convex, so are $R_+ \cap e$ and $R_- \cap e$. Thus, $R_+ \cap e$ and $R_- \cap e$ have one connected component each, and $D_+ \cap e$ will border R_+ one one side and R_- on the other side.

Since R_+ and R_- both border e, by Lemma 18, one vertex must lie in r_{r+} and the other in r_{r-}. Without loss of generality, suppose that $v_1 \in r_{r+}$. By assumption, neither vertex is type D_+. By Corollary 3, neither vertex is type D_-. Also, neither vertex can be type S because $|\gamma_r| > |\gamma_s|$ at both vertices. Thus, we must have that v_1 is type R_+ and v_2 is type R_-. This gives (a).

Because (a) is true, both $R_+ \cap e$ and $R_- \cap e$ must only border D_+, meaning that $\gamma_r^2 = \gamma_s^2$ will have no topologically significant intersections with e, giving (b).

We now show (c). Because $\gamma_r(v_1) > 0$ and $\gamma_r(v_2) < 0$, there must be a point $p \in e$ such that $\gamma_r(p) = 0$. Since e only intersects R_+, D_+ and D_-, it must hold that $p \in R_+$. Because $\gamma_s(p) \geq 0$, it follows that $\gamma_r(p) \leq \gamma_s(p)$. However, $\gamma_r(v_1) > \gamma_s(v_1)$. Thus, if we move towards v_1 from p, we must encounter the curve $\gamma_r = \gamma_s$.

Thus, $\gamma_r = \gamma_s$ intersects edge e. Using similar reasoning, $-\gamma_r = \gamma_s$ must intersect e. Since the conic $\gamma_r^2 = \gamma_s^2$ cannot intersect e more than twice, the curves $\gamma_r = \gamma_s$ and $-\gamma_r = \gamma_s$ intersect e once each.

Because $\gamma_r = \gamma_s$ transversally intersects the boundary of σ at edge e, and $\partial\sigma$ is a closed loop, the curve $\gamma_r = \gamma_s$ must intersect the boundary of σ somewhere else. Because $\gamma_r = \gamma_s$ only intersects edge e once, it must intersect e_1 or e_2. We check both cases:

case 1: $\gamma_r = \gamma_s$ intersects e_1: In this case, we travel along the path from p to $\gamma_r = \gamma_s$, then up $\gamma_r = \gamma_s$ until reaching e_1, and then along e_1 until hitting vertex v_3. We illustrate this path in Fig. 22(B).

When traveling along this path, one will either stay in the region D_+ the entire time, or it will leave D_+. If we stay in D_+ the entire time, then v_3 will be of type D_+. Further, because the entire path lies within D_+, no junction point was encountered betewen $\gamma_d = \gamma_s$ and $\gamma_r = \gamma_s$, giving (c)(i). In this case, σ will look similar to the depiction in Fig. 22(B). Otherwise, there will be some point q where this path leaves D_+. We check three cases for q, and illustrate each case as Fig. 22.

Case 1.a: q lies on $\gamma_r = \gamma_s$: We illustrate this case in Fig. 22(C). In this case, q will be a junction point between $\gamma_d = \gamma_s$ and $\gamma_r = \gamma_s$. Since $v_1 \in r_{r+}$, the orientation of $\gamma_r = \gamma_s$ must be such that v_1 is on the right side of $\gamma_r = \gamma_s$. As a result, at the point r where $\gamma_r = \gamma_s$ intersects e (see Fig. 22(C)), the oriented tangent vector of $\gamma_r = \gamma_s$ must point outside of σ.

Thus, $\gamma_r = \gamma_s$ is oriented in the opposite direction as the path that we trace. Therefore, at junction point q, the oriented vector tangent to $\gamma_r = \gamma_s$ must point into the interior of the curve $\gamma_d = \gamma_s$, so $\gamma_r = \gamma_s$ enters d_{d+} at q. We now show that q can be the only junction point between γ_d and γ_r.

Since $\gamma_d(p) > \gamma_r(p)$, but $\gamma_r(v_1) > \gamma_d(v_1)$, then there must be some point $q' \in e$ between p and v_1 such that $\gamma_d(q') = \gamma_r(q')$. Notice that the line $\gamma_d = \gamma_r$ runs through q and q'. Denote that line by l. Any junction point must lie on line l.

Let $\phi : \mathbb{R} \to l$ parametrize l such that $\phi(0) = q'$ and $\phi(1) = q$. Suppose that there existed some $t \neq 1$ such that $\phi(t)$ was a junction point. We check three subcases:

Case 1.a.i: $t < 0$: Since $\phi(0)$ lies on the boundary of σ, and $\phi(1)$ lies in the interior, then $\phi(t)$ must lie outside of σ. Thus, $\phi(t)$ cannot be a junction point.

Case 1.a.ii: $0 \leq t < 1$: Notice that

$$\begin{aligned}(\gamma_d \circ \phi)(t) &= (\gamma_d \circ \phi)((1-t)(0) + (t)(1)) \\ &= (1-t)(\gamma_d \circ \phi)(0) + t(\gamma_d \circ \phi)(1) \\ &> (1-t)(\gamma_s \circ \phi)(0) + t(\gamma_s \circ \phi)(1) \\ &\geq (\gamma_s \circ \phi)((1-t)(0) + t(1)) \\ &= (\gamma_s \circ \phi)(t)\end{aligned}$$

Thus, $(\gamma_d \circ \phi)(t) > (\gamma_s \circ \phi)(t)$, so $\phi(t)$ cannot be a junction point.

Case 1.a.iii: $t > 1$: In this case, $\gamma_d(t) = \gamma_s(t)$. Notice that

$$1 = \left(1 - \frac{1}{t}\right)(0) + \left(\frac{1}{t}\right)(t)$$

And therefore

$$\begin{aligned}(\gamma_d \circ \phi)(1) &= (\gamma_d \circ \phi)\left(\left(1 - \frac{1}{t}\right)(0) + \left(\frac{1}{t}\right)(t)\right) \\ &= \left(1 - \frac{1}{t}\right)(\gamma_d \circ \phi)(0) + \left(\frac{1}{t}\right)\gamma_d(t) \\ &> \left(1 - \frac{1}{t}\right)(\gamma_s \circ \phi)(0) + \left(\frac{1}{t}\right)(\gamma_s \circ \phi)(t) \\ &\geq (\gamma_s \circ \phi)(1)\end{aligned}$$

Thus, $(\gamma_d \circ \phi)(1) > (\gamma_s \circ \phi)(1)$, which is a contradiction. Thus, $\phi(1)$ is the only junction point between $\gamma_d = \gamma_s$ and $\gamma_r = \gamma_s$. This proves (c)(iii).

Case 1.b: q lies on e_1 and q separates D_+ from S: We illustrate this case in Fig. 22(C). Similar reasoning to the previous case gives (c)(v).

Case 1.c: q lies on the border between D_+ and R_-: In this case $D_+ \cap e_1$ will not border $S \cap e_1$, as it borders $R_+ \cap e_1$ and $R_- \cap e_1$. Thus, there must be no topologically significant intersections between $\gamma_d = \gamma_s$ and edge e_1. By assumption, there can be no junction points between $\gamma_d = \gamma_s$ and $\gamma_r = \gamma_s$. We illustrate this case in Fig. 22(D).

By assumption, $\gamma_d(q) = -\gamma_r(q)$. Let q' be the point on edge e where $\gamma_d(q') = -\gamma_r(q')$. Let l denote the line connecting q to q'. Because γ_d and γ_r are affine, then $\gamma_d = -\gamma_r$ on all of l. Thus, l is the line $\gamma_d = -\gamma_r$ and all junction points between $\gamma_d = \gamma_s$ and $-\gamma_r = \gamma_s$ must occur on l.

By assumption, $\gamma_d(q) > \gamma_s(q)$ and $\gamma_d(q') > \gamma_s(q')$. Because γ_d is affine and γ_s is convex, it follows that $\gamma_d > \gamma_s$ on all of l. Thus, there can be no junction points between $\gamma_d = \gamma_s$ and $-\gamma_r = \gamma_s$.

Because $v_1 \in R_+$, and R_- intersects e_1, it follows from Lemma 18 that $v_3 \in r_{r-}$. Thus, either $v_3 \in R_-$ or $v_3 \in D_-$. But because D_+ borders e_1, from Corollary 3, it cannot hold that $v_3 \in D_-$. So $v_3 \in R_-$. Since $v_2 \in R_-$, the entire edge $e_2 \subset R_-$. Thus, there are no topologically significant intersections between $\gamma_d = \gamma_s$ and e_3. This gives (c)(vii).

Thus, in all of case 1, we will either have (c)(i), (c)(iii), (c)(v), or (c)(vii).

Case 2: $\gamma_r = \gamma_s$ intersects e_2: In this case, we can proceed analogously to case 1, but instead of moving from p towards v_1, we move from p towards v_2. We then travel up the curve $-\gamma_d = \gamma_s$, which will intersect e_2. Arguing analogously to the previous case will give either (c)(ii), (c)(iv), (c)(vi), or (c)(vii).

Thus, if D_+ borders edge e, then conditions (a), (b), and (c) will be true. Now suppose that (a), (b), and (c) are true. If one of (c)(i) - (c)(vi) are true, we can find a point $p \in D_+$. By arguing in reverse, we can find a path to the edge e that never leaves D_+, so D_+ will intersect e.

Now suppose that (c)(vii) is true. On edge e, $R_+ \cap e$ cannot border $R_- \cap e$. Since $\gamma_r^2 = \gamma_s^2$ has no topologically significant intersections with e, then $R_+ \cap e$ and $R_- \cap e$ cannot border any region of type S. Thus, each of $R_+ \cap e$ or $R_- \cap e$ must border $D_+ \cap e$ or $D_- \cap e$. We show that $D_- \cap e = \emptyset$.

Instead, suppose that $D_- \cap e \neq \emptyset$. Then define the following sets.

- $E_1 = \{p \in \sigma : \gamma_d(p) > \gamma_r(p) \text{ and } \gamma_d(p) > -\gamma_r(p)\}$
- $E_2 = \{p \in \sigma : -\gamma_d(p) > \gamma_r(p) \text{ and } -\gamma_d(p) > -\gamma_r(p)\}$
- $E_3 = \{p \in \sigma : \gamma_r > \gamma_d \text{ and } \gamma_r > -\gamma_d\}$
- $E_4 = \{p \in \sigma : -\gamma_r > \gamma_d \text{ and } -\gamma_r > -\gamma_d\}$

Notice that $D_+ \subset E_1$, $D_- \subset E_2$, $R_+ \subset E_3$, and $R_- \subset E_4$. Thus, none of the E_i are empty. It is easy to verify that each E_i is connected, and the $\{E_i\}$ are disjoint. Further, whenever two of the E_i border each other, the boundary will lie along one of the lines $\gamma_r = \gamma_d$, or $-\gamma_r = \gamma_d$.

In order for this to occur, the lines $\gamma_r = \gamma_d$ and $-\gamma_r = \gamma_d$ must divide σ into at least four regions (one for each of the E_i). This can only happen if $\gamma_r = \gamma_s$ intersects $-\gamma_r = \gamma_s$ within σ. Let q be the intersection point. Then $\gamma_r(q) = -\gamma_r(q)$, implying that $\gamma_r(q) = \gamma_d(q) = 0$. By assumption, it cannot hold that $\gamma_s(q) = 0$, so $q \in S$.

Let $p \in D_+$. Let s be the segment between p and q. Then $S \cap s \neq \emptyset$ and $D_+ \cap s \neq \emptyset$. Because $\gamma_d(p) > \gamma_r(p)$ and $\gamma_d(q) = \gamma_r(q)$, and γ_d and γ_r are both affine, there will be no point on s where $\gamma_r > \gamma_d$, so $R_+ \cap s = \emptyset$. Similarly, $R_- \cap s = \emptyset$. Also, it is not possible for $D_+ \cap s$ to border $D_- \cap s$. Thus, $D_+ \cap s$ will border $S \cap s$, meaning that D_+ borders S.

Because D_+ borders S it follows that the curve $\gamma_d = \gamma_s$ is not empty. Further, a portion of the curve $\gamma_d = \gamma_s$ separates D_+ from S. Because $\gamma_d = \gamma_s$ has no junction points and no topologically significant intersections with any edge, it follows that $\gamma_d = \gamma_s$ must be a circle or ellipse in the interior of σ. By Lemma 16, this would imply that $D_- = \emptyset$.

Thus, D_- cannot intersect σ at all. Therefore, $R_+ \cap e$ borders $D_+ \cap e$, implying that D_+ borders e. □

Lemma 21. *Within σ, D_+ will border R_+ if and only if $D_+ \neq \emptyset$, $R_+ \neq \emptyset$, and one of the following is true:*

(a) There is a junction point between $\gamma_d = \gamma_s$ and $\gamma_r = \gamma_s$.

(b) There exists an edge e of σ, such that $D_+ \cap e \neq \emptyset$ and $R_+ \cap e \neq \emptyset$ and one of the following statements is true:

(i) One vertex of e is of type D_+, and $\gamma_d = \gamma_s$ has no topologically significant intersections with e.

(ii) One vertex of e is of type R_+ and $\gamma_r = \gamma_s$ has no topologically significant intersections with e.

(iii) Neither vertex is of type R_+ or D_+, and $\gamma_d = \gamma_s$ and $\gamma_r = \gamma_s$ each have at most one topologically significant intersection with e.

Analogous lemmas can be made with each pair in $\{D_+, D_-\} \times \{R_+, R_-\}$.

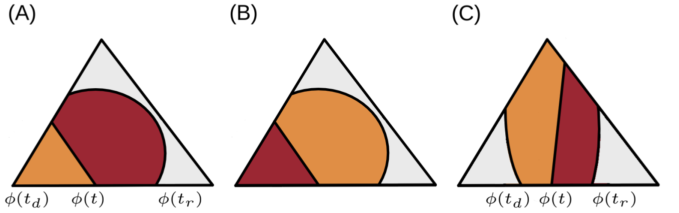


Fig. 23: We demonstrate various cases in Lemma 21. Here the edge e corresponds to the bottom edge of the triangle. The orange region is D_+ while the red region is R_+. (A) Case 1 of the proof, where one vertex is of type D_+. (B) Case 2 of the proof, where one vertex is of type R_+. (C) Case 3 of the proof, where neither vertex is of type R_+ or D_+.

Proof. We first show that if D_+ border R_+, then one of statements (a) or (b) must be true. If D_+ borders R_+, then consider the line $\gamma_r = \gamma_d$. Choose some point $p \in \sigma$ on that line that is on the border between D_+ and R_+, and trace the line. When tracing along the line, one will either encounter a region of type S, whereby a junction point is reached, giving (a), or one will hit an edge e.

If an edge e is hit, then trivially $D_+ \cap e \neq \emptyset$ and $R_+ \cap e \neq \emptyset$. Let $\phi : [0, 1] \to e$ be a parametrization. Let $t \in [0, 1]$ be such that $\phi(t)$ is the point where $\gamma_d = \gamma_r$ intersects e. Then there will be intervals $I_R, I_D \in [0, 1]$ that have t as an endpoint such that $\phi(I_1) = R_+ \cap e$ and $\phi(I_2) = D_+ \cap e$. Let $t_r, t_d \in [0, 1]$ be such that $I_R = [t_r, t]$ and $I_D = [t, t_d]$.

We now check three cases for the vertices of e. We illustrate each case in Fig. 23, where the bottom edge of the triangle corresponds to e..

Case 1: One vertex is of type D_+: We illustrate this case in Fig. 23(A). This case implies that $t_d = 1$. Thus, the only region that $D_+ \cap e$ borders is $R_+ \cap e$ with boundary at $\phi(t)$. Thus, $D_+ \cap e$ will never border $S \cap e$, so $\gamma_d = \gamma_s$ will have no topologically significant intersections with e. This gives (b)(i).

Case 2: One vertex is of type R_+: We illustrate this case in Fig. 23(B). This case is similar to case 1 and gives (b)(ii)

Case 3: Neither vertex is of type D_+ or R_+: We illustrate this case in Fig. 23(C). In this case, $D_+ \cap e$ and $R_+ \cap e$ border each other at $\phi(t)$. Thus, $\gamma_d = \gamma_s$ could only have a topologically significant intersection at $\phi(t_d)$ and $\gamma_r = \gamma_s$ could only have a topologically significant intersection at $\phi(t_r)$. Thus, each curve could only have at most one topologically significant intersection. This gives (b)(iii).

Thus, we have shown that R_+ bordering D_+ implies that (a) or (b) is true. We now prove the opposite direction. We show that each condition implies that D_+ borders R_+. It is trivial that (a) implies that D_+ borders R_+. We now show this for (b). Let $\phi : [0, 1] \to e$ parametrize e.

(b)(i): By Lemma 19, the region $E_1 := \{p \in e : \gamma_d > |\gamma_r|\}$ will border the region $E_2 := \{p \in e : \gamma_r(p) > |\gamma_d(p)|\}$. Because D_+ contains a vertex v of e, E_1 contains a vertex v of e (w.l.o.g. suppose that E_1 contains $\phi(0)$). Thus, there exists some $t \in [0, 1]$ so that $E_1 = \phi([0, x])$. Because E_1 contains a vertex of e, it can only border one of E_2 or $E_3 := \{p \in e : -\gamma_r(p) > |\gamma_d(p)|\}$ at $\phi(x)$. Because E_1 borders E_2, it thus cannot border E_3.

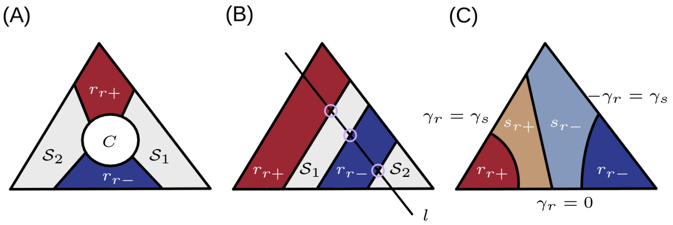


Fig. 24: (A) We show regions r_{r+}, r_{r-}, $\mathcal{S}_1$ and $\mathcal{S}_2$ from the proof of Lemma 23. It is not possible to fill in the region C to prevent r_{r+} from bordering r_{r-} and $\mathcal{S}_1$ from bordering $\mathcal{S}_2$. (B) We show regions r_{r+}, r_{r-}, $\mathcal{S}_1$ and $\mathcal{S}_2$ from the proof of Lemma 23. In this case, the line l would intersect $\gamma_r^2 = \gamma_s^2$ three times (marked in purple). (C) A partition corresponding to the only possible eigenvector graph if $r_{r+} \neq \emptyset$ and $r_{r-} \neq \emptyset$.

Because $\gamma_d = \gamma_s$ does not have any topologically significant intersections with e, the region D_+ must border either R_+ or R_-. Notice that $D_+ \subset E_1$, $R_+ \subset E_2$, and $R_- \subset E_3$. Because E_1 does not border E_3, then D_+ cannot border R_-. Thus, D_+ borders R_+.

(b)(ii): This is similar to (b)(i).

(b)(iii): Because neither vertex is of class D_+, the region $D_+ \cap e$ must border a different region on both sides. Since $\gamma_d = \gamma_s$ has at most one topologically significant intersection with e, then one of the regions that borders D_+ must be R_+ or R_-. We check two cases:

case 1: $R_- \cap e = \emptyset$: In this case, D_+ cannot border R_- so D_+ borders R_+.

case 2: $R_- \cap e \neq \emptyset$: In this case, by Corollary 3, we must have $D_- \cap e = \emptyset$. Arguing symmetrically, we can find that R_+ must border D_+ or D_-. Since $D_- = \emptyset$, it must follow that R_+ borders D_+. □

Lemma 22. *Suppose that r_{r+} and r_{r-} both intersect σ. Excluding degenerate points, the eigenvector graph of σ will be a linear graph of the structure r_{r+}–s_{r+}–s_{r-}–r_{r-}.*

Proof. Let $\mathcal{S} = s_{r+} \cup s_{r-}$. We claim that $\mathcal{S}$ has only one connected component. Suppose, to the contrary, that $\mathcal{S}$ had two connected components, which we will call $\mathcal{S}_1$ and $\mathcal{S}_2$.

It is not possible for both r_{r+} and r_{r-} to border both $\mathcal{S}_1$ and $\mathcal{S}_2$. Because r_{r+} and r_{r-} do not border each other, and $\mathcal{S}_1$ and $\mathcal{S}_2$ do not border each other, such a case would be geometrically impossible, as we demonstrate in Fig. 24(A). In Fig. 24(A), there is no way to fill in the region C such that r_{r+} does not border r_{r-} and s_{r+} does not border s_{r-}. Thus, at least one of r_{r+} or r_{r-} borders only one of $\mathcal{S}_1$ or $\mathcal{S}_2$. Without loss of generality, suppose that r_{r+} borders $\mathcal{S}_1$, but not $\mathcal{S}_2$.

Let l be a line connecting r_{r+} to $\mathcal{S}_2$. Since r_{r+} does not border r_{r-} or $\mathcal{S}_2$, it follows that, after exiting r_{r+}, the line l must enter $\mathcal{S}_1$. Thus, the line l passes between $\mathcal{S}_1$ and $\mathcal{S}_2$. Because r_{r+} and r_{r-} are convex, and l has already left r_{r+}, l must pass through r_{r-} between $\mathcal{S}_1$ and $\mathcal{S}_2$. In total, when traveling from r_{r+} to $\mathcal{S}_2$, the line l passes through $r_{r+} \to \mathcal{S}_1 \to r_{r-} \to \mathcal{S}_2$. We illustrate this case in Fig. 24(B), where we highlight the intersectiosn between $\gamma_r^2 = \gamma_s^2$ and l in purple.

Whenever l passes from a region where $\gamma_r^2 > \gamma_s^2$ to one where $\gamma_s^2 > \gamma_r^2$ (or vice versa), l must intersect the curve $\gamma_r^2 = \gamma_s^2$. In the path from $r_{r+} \to \mathcal{S}_1 \to r_{r-} \to \mathcal{S}_2$, then, the line l must intersect $\gamma_r^2 = \gamma_s^2$ exactly three times. However, since $\gamma_r^2 = \gamma_s^2$ is a conic section, it is impossible for a line to intersect it exactly three times. Thus, there is only one connected component of type S.

Therefore, $\gamma_r^2 = \gamma_s^2$ will divide σ into three regions: r_{r+}, r_{r-}, and $\mathcal{S}$. Since there is no point where $\gamma_r = \gamma_s = 0$, the line $\gamma_r = 0$ must not intersect $\gamma_r^s = \gamma_s^2$. Thus, the line $\gamma_r = 0$ will divide $\mathcal{S}$ into two connected components corresponding to s_{r+} and s_{r-}. We illustrate the final partition in Fig. 24(C). Trivially, s_{r+} borders s_{r-}. r_{r+} must only border s_{r+}, and r_{r-} must only border s_{r-}. Thus, we σ will have en eigenvector graph of the structure r_{r+}–s_{r+}–s_{r-}–r_{r-}. □

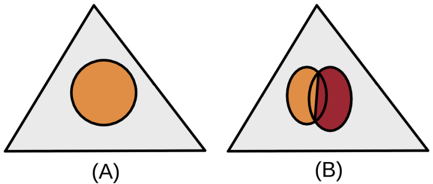


Fig. 25: We depict the two possible ways that a region of type S can completely surround another region. (A) The case where D_+ (or whichever region is surrounded) has no junction points. (B) The case where D_+ has exactly two junction points with R_+ (where either D_+ or R_+ can be substituted for D_- or R_-).

Lemma 23. *Let e be an edge of σ. Then r_{r+} will border e if and only if r_{r+} contains a vertex of e, or if $\gamma_r = \gamma_s$ intersects e. The analogous claim for r_{r-} is also true.*

Proof. It is trivial that these conditions are sufficent for r_{r+} to border e. We now show that they are necessary. Suppose that r_{r+} borders e. If r_{r+} borders a vertex of e, then the condition is met. Otherwise, $r_{r+} \cap e \neq e$. From Lemma 14 and Lemma 15, this implies that $r_{r+} \cap e$ must border $s_{r+} \cap e$ in which case $r_{r+} \cap e$ will be separated from its neighbor along the curve $\gamma_r^2 = \gamma_s^2$, whereby $\gamma_r = \gamma_s$ will intersect e. □

Lemma 24. *Let e be an edge of σ. Suppose that s_{r+} has only one connected component. Then s_{r+} will intersect e if and only if s_{r+} contains a vertex of e, if $\gamma_r = \gamma_s$ intersects e, or if $\gamma_r = 0$ intersects e. The analogous claim for s_{r-} is also true.*

Proof. The proof is very similar to that of Lemma 23. □

Lemma 25. *We say that that a region $R \subset \sigma$ of type S completely surrounds D_+ if $D_+ \neq \emptyset$ and all paths starting from any point in D_+ will encounter S before an edge of σ.*

(1) This situation will occur if and only if all of the following are true:

(a) D_+ intersects the interior of σ.

(b) $\gamma_d = \gamma_s$ has no topologically significant intersections with any edges of σ.

(c) All vertices of σ are of type S.

(d) D_+ has no junction points, or the following are true:

(i) The curve $\gamma_d = \gamma_s$ has exactly two junction points. The junction points are either both with $\gamma_r = \gamma_s$, or both with $-\gamma_r = \gamma_s$ (w.l.o.g. suppose they are with $\gamma_r = \gamma_s$).

(ii) The curve $\gamma_r = \gamma_s$ has exactly two junction points. Both junction points are with $\gamma_d = \gamma_s$.

(iii) The curve $\gamma_r = \gamma_s$ has no topologically significant intersections with any edge of σ.

(2) If a region R of type S completely surrounds D_+, then the eigenvector partition of σ will only be one region of type S, and any regions completely surrounded by S.

Analogous lemmas can be shown for D_-, R_+, and R_-.

Proof. In Fig. 25, we illustrate the two ways that a region of type S can completely surround a region of another type. Fig. 25(A) corresponds to the case where D_+ has no junction points, whereas (B) corresponds to the case where D_+ has exactly two junction points with R_+.

We first show (1). First, suppose that R completely surrounds D_+. (a) is trivial. Also, D_+ cannot intersect any edges, giving (b).

In order for R, which is type S, to surround D_+, it must follow that d_{d+} is also completely surrounded by R. As such, the curve $\gamma_d = \gamma_s$ is a connected component of a conic section that does not intersect any edges of σ, so it must be a circle or ellipse whose interior is of type d_{d+}. By Lemma 17, either r_{r+} or r_{r-} must intersect $\gamma_d = \gamma_r$. Without loss of generality, suppose that r_{r+} intersects $\gamma_d = \gamma_r$.

By definition, $r_{r+} \cap S = \emptyset$. By assumption, every path starting at d_{d+} that passes through r_{r+} must intersect S before intersecting an edge of σ. Since d_{d+} intersects r_{r+}, it follows that S also completely surrounds r_{r+}. Thus, the curve $\gamma_r = \gamma_s$ is also a circle or ellipse in the interior of R. From Lemma 16, $d_{d-} = \emptyset$ and $r_{r-} = \emptyset$, implying that $D_- = \emptyset$ and $R_- = \emptyset$. Since d_{d+} nor r_{r+} are ellipses in the interior of σ, and $d_{d-} = \emptyset$ and $r_{r-} = \emptyset$, it follows that all three vertices of σ must be of type S, giving (c).

If $\gamma_r = \gamma_s$ lies completely inside of $\gamma_d = \gamma_s$, then any points in r_{r-} must lie in the interior of D_+. However, since D_+ is convex, this forces $R_+ = \emptyset$. Thus, D_+ will not border any other regions, giving (d).

Now suppose that $\gamma_r = \gamma_s$ does not lie completely inside of $\gamma_d = \gamma_s$. Then the two curves will intersect at points along the line where $\gamma_d = \gamma_r$. Since $\gamma_d = \gamma_s$ and $\gamma_r = \gamma_s$ are both subsets of conic sections, the line $\gamma_d = \gamma_r$ can only intersect each curve at most twice. Thus, $\gamma_d = \gamma_s$ and $\gamma_r = \gamma_s$ will intersect exactly two times. Since $d_{d-} = \emptyset$ and $r_{r-} = \emptyset$, these will be the only junction points along either curve. This fact gives (d)(i) and (d)(ii).

Also, because $\gamma_r = \gamma_s$ is an ellipse in the interior of σ, (d)(iii) must follow, so we have (d).

Thus, if R contains D_+ in its interior, then (a), (b), (c), and (d) must follow.

Now suppose that conditions (a), (b), (c), and (d) are true. Without loss of generality, assume that $\gamma_d = \gamma_s$ has no junction points with $-\gamma_r = \gamma_s$.

We claim that D_+ will not border any edge e. Suppose, to the contrary, that it did. Since no vertex is of type D_+, $D_+ \cap e$ will border two other regions within e. If either region is of type S, then $\gamma_d = \gamma_s$ will have a topologically significant intersection between that region and D_+, which we assume cannot occur. Thus, $D_+ \cap e$ can only border $R_+ \cap e$ and $R_- \cap e$. Since $R_+ \cap e$ and $R_- \cap e$ are connected, $D_+ \cap e$ will border both $R_+ \cap e$ and $R_- \cap e$, intersecting one region on each side. In that case, by Lemma 18, then, one vertex of e is of type r_{r+} and the other is of type r_{r-}. In that case, neither vertex can be of type S, violating assumption (c). Thus, D_+ does not border any edge e.

We also claim that D_+ cannot border R_-. Suppose that it did. By Lemma 21, if D_+ did border R_-, then either $\gamma_d = \gamma_s$ and $-\gamma_r = \gamma_s$ would share a junction point, or D_+ and R_- would both intersect some edge e of σ. However, neither case is possible by assumption. Thus, D_+ does not border R_-.

We now check two cases:

Case 1: $\gamma_d = \gamma_s$ has no junction points: Then, by Lemma 21, D_+ will not border R_+. Thus, D_+ can only border a region R of type S, and does not border any vertices or edges of σ. Therefore, D_+ must be completely surrounded by R.

Case 2: $\gamma_d = \gamma_s$ has exactly two junction points with $\gamma_r = \gamma_s$: In this case, then clearly D_+ will border R_+. Using similar reasoning to before, we can show that R_+ will only border D_+ and a region of type S. We can also show that R_+ will not border any edges of σ. Thus, $R_+ \cup D_+$ will only border regions of type S, and will not border any edges of σ. This is possible only if $R_+ \cup D_+$ is completely surrounded by a single region R of type S, meaning that both R_+ and D_+ lie within the interior of R. This completes the proof of (1).

To show (2), in the reasoning above, we demonstrated that if D_+ lies in the interior of a region R of type S, then the curves $\gamma_d = \gamma_s$ and $\gamma_r = \gamma_s$ are both ellipses contained in the interior of σ. Further, either $R_+ = \emptyset$, or R_+ and D_+ are both completely surrounded by R.

By Lemma 16, it follows that $d_{d-} = \emptyset$ and $r_{r-} = \emptyset$. Thus, the only regions within σ will be D_+, R_+ (if it is not empty), and R, which has type S. □

J.3 Proofs of Main Results

We now prove the main results. For Lemma 6, the goal is to use our invariant to recover the structure of the eigenvector graph of σ. We also wish to recover: for each node n of the eigenvector graph, which vertices and edges of σ border the regions within σ corresponding to n.

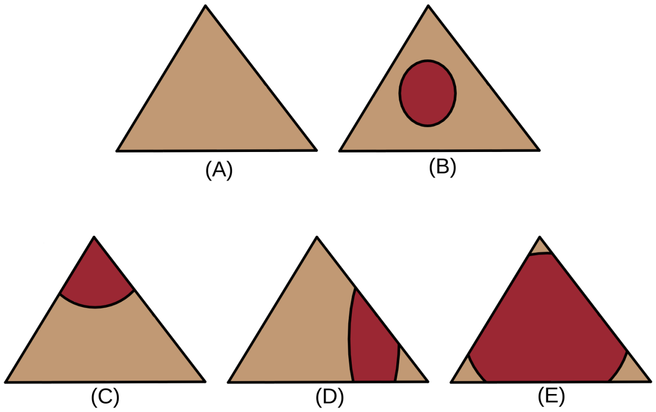


Fig. 26: We demonstrate the different possible ways that $\gamma_d = \gamma_s$ can divide σ into multi regions. The red region is the interior of $\gamma_d = \gamma_s$, and the orange region is the outside. (A) $\gamma_d = \gamma_s$ does not intersect σ. (B) $\gamma_d = \gamma_s$ does not intersect the boundary of σ, but forms a region within its interior. (C) $\gamma_d = \gamma_s$ intersects the boundary of σ twice, forming two regions. (D) $\gamma_d = \gamma_s$ intersects the boundary of σ four times, forming three regions. (E) $\gamma_d = \gamma_s$ intersects the boundary of σ six times, forming four regions.

Lemma 6. *Except in special cases, the topology of the eigenvector partition of σ is determined by the following conditions:*

(a) Whether each of the curves $\gamma_r = \gamma_s$ and $-\gamma_r = \gamma_s$ intersects the interior of σ.
(b) How many times the curves $\gamma_r = \gamma_s$ and $-\gamma_r = \gamma_s$ intersect each edge of σ.
(c) The classification of each vertex of σ according to the eigenvector partition.

Proof. If r_{r+} and r_{r-} are both inside of σ, then by Lemma 22, there is only one possible eigenvector graph. That eigenvector graph will have one node of each type r_{r+}, s_{r+}, s_{r-} and r_{r-}. Which regions border which vertices can be determined from the classifications of each vertex, and Lemma 23 gives which edges are bordered by r_{r+} and r_{r-}. From the vertex classification, we know whether $\gamma_r > 0$ for each vertex. From that information, we can deduce which edges intersect the line $\gamma_r = 0$. Then, Lemma 24 allows us to deduce which edges border s_{r+} and s_{r-}.

Now suppose that r_{r+} and r_{r-} do not both intersect σ. Without loss of generality, suppose that r_{r-} does not intersect σ. Since r_{r+} is convex, its boundary, given by $\gamma_r = \gamma_s$, will intersect the boundary of σ either zero, two, four, or six times. In doing so, $\gamma_r = \gamma_s$ will divide σ into, respectively, one, two, three, or four regions. If $\gamma_r = \gamma_s$ never intersects the boundary of σ, it can also possibly be an ellipse in the interior of σ, thus dividing σ into two regions. We illustrate each of these cases in Fig. 26. Further, the line $\gamma_r = 0$ We now check each case.

Case 1: $\gamma_r = \gamma_s$ intersects σ zero times: We check four subcases:

Case 1(i): $\partial r_{r+} = \emptyset$; all vertices are the same classification: In this case, every vertex must be of the same class, either r_{r+}, s_{r+} or s_{r-}. The eigenvector graph will be a single node, and its corresponding region will border every vertex and edge of σ.

Case 1(ii): $\partial r_{r+} = \emptyset$; some vertices have different classifications: In this case, σ will have regions of multiple classifications. None of them can be r_{r+}, as otherwise ∂r_{r+} would separate r_{r+} from another region. Thus, some vertices will be s_{r+} and others s_{r-}. By inspecting the vertex classifications, we can find which edges will intersect the line $\gamma_r = 0$. The line $\gamma_r = 0$ will divide σ into two regions, one of type s_{r+} and the other of type s_{r-}. Using Lemma 24, we can deduce which edges intersect each of the two regions.

Case 1(iii): $\partial r_{r+} \neq \emptyset$; all vertices are the same classification: In this case, r_{r+} will be a convex region contained within the interior of σ. Because r_{r+} can only border s_{r+}, it must be contained within the interior of s_{r+}. Thus, every vertex must be s_{r+}. As a result, the eigenvector graph will have two nodes, one of type r_{r+} and the other of type s_{r+}. The region of type s_{r+} will border every vertex and edge.

Case 1(iv): $\partial r_{r+} \neq \emptyset$; not all vertices are the same classification: In this case, r_{r+} will be a convex region in the interior of σ. As before, it must be contained within s_{r+}. In this case, some vertices must be of type s_{r+} and others of type s_{r-}. The topology can be found by using similar reasoning to before.

Case 2: $\gamma_r = \gamma_s$ intersects $\partial\sigma$ twice: In this case, $\gamma_r = \gamma_s$ will divide σ into two regions. We can determine which is which by inspecting the classifications of the vertices.

We can inspect the classifications of the vertices to detect if the line $\gamma_r = 0$ runs through σ. If it does, it will divide the region where $\gamma_r < \gamma_s$ into two regions, one of type s_{r+} and the other s_{r-}, where r_{r+} will border s_{r+}. If $\gamma_r = 0$ does not run through σ, then σ will be divided into two regions, one of type s_{r+} and the other of type r_{r+}. In either case, we can compute the eigenvector graph. Then, which regions intersect which vertices and edges can be computed from Lemma 23 and Lemma 24.

Case 3: $\gamma_r = \gamma_s$ intersects $\partial\sigma$ four times: In this case, $\gamma_r = \gamma_s$ will divide σ into three regions. In two regions, $\gamma_s > \gamma_r$. The third region, which lies between the other two, will satisfy $\gamma_r > \gamma_s$.

Because r_{r+} is convex, any region where $\gamma_s > \gamma_r$ (that lies outside of r_{r+}) must contain at least one vertex of σ. One can verify that two regions where $\gamma_s > \gamma_r$ will be connected if and only if both regions border some edge e that does not intersect $\gamma_r = \gamma_s$. By inspecting which vertices and edges are intersected by $\gamma_r = \gamma_s$, we can uniquely identify the different regions where $\gamma_r > \gamma_s$.

Using similar reasoning previous cases, we can compute the regions that intersect the line $\gamma_r = 0$. Any region where $\gamma_s > \gamma_r$ not intersected by the line $\gamma_r = 0$ will be entirely classified as s_{r+}. Any region where $\gamma_s > \gamma_r$ intersected by $\gamma_r = 0$ will be divided into two regions, one of type s_{r+} and the other s_{r-}. The region of type s_{r+} border r_{r+}, while s_{r-} will not border r_{r+}.

Which vertices intersect r_{r+} follow from their classifications, and Lemma 23 determines which edges border r_{r+}. For any region where $\gamma_s > \gamma_r$, if that region borders an edge e, then it must border some vertex v of e. Thus, we can identify all edges and vertices bordering that region. For any region where $\gamma_s > \gamma_r$ that intersects the line $\gamma_r = 0$, by inspecting the position of the line we can determine which edges intersect the sub-regions of type s_{r+} and s_{r-}.

Case 4: $\gamma_r = \gamma_s$ intersects $\partial\sigma$ six times: In this case, $\gamma_r = \gamma_s$ will divde σ into four regions. There will be three regions where $\gamma_s > \gamma_r$. From here, computing the topology follows closely to the previous case. □

Lemma 7. *Except in special cases, the topology of the eigenvalue partition of σ is determned by the following conditions:*

(a) The classification of each vertex in the eigenvalue partition.
(b) For each of the curves $\gamma_d = \gamma_s$, $-\gamma_d = \gamma_s$, $\gamma_r = \gamma_s$, and $-\gamma_r = \gamma_s$, determine the order in which the following points are encountered when traveling counterclockwise around the curve.
  (i) Each boundary point (i.e, a point on the boundary between two regions of the eigenvalue partition) where the curve enters or leaves σ, and the orientation of the curve at that point.
  (ii) Each junction point, along with the orientations of the curves that intersect at that point.
(c) For each edge e of σ, for each point identified in (b.i) that lies on e, determine which point is closest to each vertex of e.
(d) For each region type in the eigenvalue partition, except where $\gamma_s > |\gamma_d|$ and $\gamma_s > |\gamma_r|$, if its boundary curve does not intersect any edges or other curves, check whether that region is present in the eigenvalue partition of σ.

Proof. We assume that every region of the eigenvalue partition does not completely surround any other region. Because D_+, D_-, R_+, and R_- are convex, the only type of region that could surround another

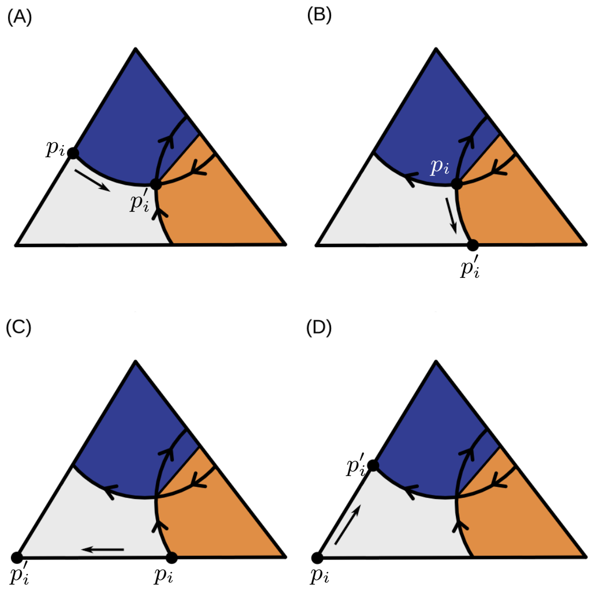


Fig. 27: If p is the vertex of the region of type S, we identify the next vertex along the boundary of S when traveling clockwise. (A) p_i is a point where a conic section enters σ. (B) p_i is a junction point. (C) p_i is a point where a conic section leaves σ. (D) p_i is a vertex of σ.

region is the region of type S. Lemma 25(1) gives a necessary and sufficient condition for such a situation to occur that can be determined from (a), (b), (c), and (d). Further, Lemma 25(2) demonstrates that, when such a case arises, the topology is very easy to recover from (a) (b) (c) and (d).

We first claim that the number of nodes in the eigenvector graph of each region type can be recovered.

If any vertex is of type D_+, as in (a), or if $\gamma_d = \gamma_s$ has any junction points or topologically significant intersections with edges of σ, as described in (b), then clearly $D_+ \cap \sigma \neq \emptyset$. If $\gamma_d = \gamma_s$ does not have any intersections, then from (d) we will know whether or not $D_+ \cap \sigma = \emptyset$. The same is true for D_-, R_+, and R_-. Thus, we will know whether there is at least one node of each of these types in the eigenvalue graph. By Lemma 13, D_+, D_-, R_+, and R_- each have at most one connected component, so there can be at most one node of each of these types in the eigenvalue graph. Thus, we can recover the number of nodes in the eigenvalue graph of types D_+, D_-, R_+, and R_-.

We are only left to find the number of type S. We will compute the number of nodes of type S by tracing the boundary of each region within the eigenvalue partition of σ using the points from (b).

Let $P = \{p_1, p_2, \dots, p_n\}$ be all of the points from (b), along with the vertices of σ that are of type S (which we can identify from (a)). It is easy to show that each $p \in P$ borders one region of type S.

The boundary of each region R of type S must consist of sections of the conics $\gamma_d^2 = \gamma_s^2$ and $\gamma_r^2 = \gamma_s^2$, as well as sections of edges of σ. Any time that two of these segments meet on the boundary of R, they will intersect at some point $p \in P$. Thus, for each region of type S, by traversing the boundary one will encounter a loop of points in P.

Now let $p_i \in P$ be a vertex of the boundary of some region R of type S. We claim that (b) and (c) allow us to compute the next vertex p'_i that immediately follows p_i when tracing R clockwise. We check four cases. We illustrate each case in Fig. 27.

Case 1: A conic section l leaves σ at p_i: In this case, p'_i will be the next point in P that l intersects when traveling counterclockwise along l. We can find this point p'_i from (b). We illustrate this case in Fig. 27(A).

Case 2: p_i is a junction point: Denote the two curves that meet at the junction point as l_1 and l_2. At the intersection, one of the conics will be entering the interior of the other at p_i. Without loss of generality, suppose that l_1 enters the interior of l_2. Then we find the point p'_i along l_1 that comes after p_i when traveling counterclockwise along l_1. we illustrate this case in Fig. 27(B).

Case 3: A conic section l enters σ at p_i: Let e be the edge where l enters σ. Let v_1 and v_2 be the vertices of e, such that v_1 comes after v_2 when traversing σ clockwise. In this case, then the boundary of R will travel towards v_1. Then, p'_i will be the closest point in P to p_i along e when traveling towards v_1. We can identify p'_i by identifying the relative positions of each point in $P \cap e$ relative to vertices v_1 and v_2. We illustrate this case in Fig. 27(C).

Case 4: p_i is a vertex of σ: In this case, let e be the edge that comes after p_i when traversing σ in clockwise order. Then p'_i will be the point on e closest to p_i. We illustrate this case in Fig. 26(D).

Across all cases, we can find the next point p'_i that follows after point p_i. This allows us to define a permutation on the points in P. Each cycle of the permutation will correspond to a different connected component of the boundary of some region R of type S. Since no region of type S surrounds any other region, the boundary of each region R of type S will be connected. Thus, we can uniquely identify each region of type S.

Next, we find which edges and vertices are adjacent to each region. Clearly, the construction of each region of type S will yield which vertices and edges it borders.

For the other types of regions, we can inspect the vertex classifications and edge intersections to identify which regions intersect each edge. For D_+, if a vertex v is of type D_+, then D_+ will border v as well as the two edges the join at v. Also, D_+ must border an edge e if $\gamma_d = \gamma_s$ has a topologically significant intersection with e. If, for some edge e, D_+ does not intersect either vertex of e, and $\gamma_d = \gamma_s$ has no topologically significant intersections with e, then Lemma 20 gives a necessary and sufficient condition for D_+ to border e that can be determined from (a), (b), (c), and (d). We can use the same approach for D_-, R_+, and R_-.

Finally, we must determine which regions of the eigenvalue partition border one another. The construction of each region of type S will yield which other regions it borders. Lemma 21 provides a necessary and sufficient condition for two regions of types D_+, D_-, R_+, or R_- to border each other that can be computed from (a), (b), (c), (d), and which edges and vertices each region borders. □

J.4 Handling of Edge Cases

In the previous proofs, we made the following assumptions:

(i) $\gamma_d^2 = \gamma_s^2$ and $\gamma_r^2 = \gamma_s^2$ only intersect each other and each edge of σ transversally.
(ii) No junction points occur on the edges of σ.
(iii) No two of the functions $|\gamma_d|$, $|\gamma_r|$, or γ_s are exactly equal on all of $\mathbb{R}^2$.
(iv) The conic sections $\gamma_d^2 = \gamma_s^2$ and $\gamma_r^2 = \gamma_s^2$ are either empty or have infinitely many points.
(v) The curve $\gamma_d = \gamma_s$ will border one region of type d_{d+} and another where $\gamma_s > |\gamma_d|$. Make similar assumptions about $-\gamma_d = \gamma_s$, $\gamma_r = \gamma_s$, and $-\gamma_r = \gamma_s$.
(vi) $|\gamma_d|$, $|\gamma_r|$ and γ_s are never equal on a vertex of σ.
(vii) There are no vertices of σ where $\gamma_r = 0$.
(viii) There are no points in σ where $\gamma_d = \gamma_s = 0$ or $\gamma_r = \gamma_s = 0$.

In Appendix H, we ensure (i)-(v) using symbolic perturbations. Thus, these are fair assumptions to make in our proofs.

For (vi) or (vii), it is possible that $|\gamma_d| = |\gamma_r|$ or $|\gamma_d| = \gamma_s$ or $|\gamma_r| = \gamma_s$ or $\gamma_r = 0$ on some vertex of σ. In that case, we track it in our topological invariant. It is not difficult to modify our existing proofs to accommodate these cases.

(viii) is more difficult to handle. If assumption (viii) does not hold, and there exists some $z \in \sigma$ satisfying $\gamma_d(z) = \gamma_s(z) = 0$, but $|\gamma_r(z)| > 0$, then the point z will lie within R_+ or R_- and will not lie on the boundary between any different regions. Thus, we can apply a small perturbation to γ_d or γ_s to ensure that $\gamma_d(z) \neq 0$ or $\gamma_s(z) \neq 0$ (ensuring that (viii) holds) without altering the topology. It is similar if $\gamma_r(z) = \gamma_s(z) = 0 < |\gamma_d(z)|$.

Thus, we are only left to handle the case where there is a point z where $\gamma_d(z) = \gamma_r(z) = \gamma_s(z) = 0$. To handle the topology, we employ the following lemmas:

Lemma 26. *Suppose that $z \in \sigma$ is a point where $\gamma_d(z) = \gamma_r(z) = \gamma_s(z) = 0$. Let p lie on the boundary of σ. Suppose that $p \in D_+$. Let q lie on the segment between p and z. Then $q \in D_+$.*

Similar lemmas are true for D_-, R_+, R_-, r_{r-}, r_{r+} and r_{r-}.

Proof. This follows from the fact that γ_d and γ_r are affine, and γ_s is convex. □

Lemma 27. *Suppose that $z \in \sigma$ is a point where $\gamma_d(z) = \gamma_r(z) = \gamma_s(z) = 0$. Let p lie on the boundary of σ. Suppose that $p \in s_{r+}$. Let q lie on the segment between p and z. Then $q \in s_{r+}$.*

Similar lemmas are true for s_{r-} and S.

Proof. Let l be the line connecting p to z. Let ϕ parametrize l with $\phi(0) = z$ and $\phi(1) = p$. Notice that $(\gamma_r \circ \phi)^2$ and $(\gamma_s \circ \phi)^2$ are both quadratic functions. Both of these functions will have a vertex at $t = 0$, where they intersect. Since $(\gamma_r \circ \phi)(1)^2 < (\gamma_s \circ \phi)(1)^2$, it follows that $(\gamma_r \circ \phi)(t)^2 < (\gamma_s \circ \phi)(t)^2$ for every $t > 0$. Thus, $\gamma_s(q)^2 > \gamma_r(q)^2$.

Additionally, because γ_r is affine, $\gamma_r(p) > 0$, and $\gamma_r(z) = 0$, it follows that $\gamma_r(q) > 0$. Thus, $q \in s_{r+}$. □

Thus, for both the eigenvector and eigenvalue partitions, the toplogy of σ is determined entirely based on the classifications of points along the boundary. By analyzing where the curves $\gamma_d = \gamma_s$ and $\gamma_r = \gamma_s$ have topologically significant intersections with the boundary of σ, as well as the classifications of each vertex, we are able to determine the topology around the boundary of σ. If there is a point $z \in \sigma$ where $\gamma_d(z) = \gamma_r(z) = \gamma_s(z) = 0$, we also note this in our invariant. Thus, when there is a point where $\gamma_d(z) = \gamma_r(z) = \gamma_s(z)$, our invariant is enough to recover the topology of σ.

References

[1] Data from Computer Graphics Lab at ETH Zurich. https://cgl.ethz.ch/research/visualization/data.php. 1

[2] Scientific data reduction benchmarks. https://sdrbench.github.io/. 1

[3] T. Günther, M. Gross, and H. Theisel. Generic objective vortices for flow visualization. *ACM Transactions on Graphics (TOG)*, 36(4):1–11, 2017. doi: 10.1145/3072959.3073684 1

[4] E. C. M. S. Information. Global ocean physics reanalysis, 2025. doi: 10.48670/moi-00021 1

[5] G. Kindlemann. Diffusion tensor MRI datasets. https://www.sci.utah.edu/~gk/DTI-data/. 1

[6] M. Patel and D. H. Laidlaw. Visualization of 3D stress tensor fields using superquadric glyphs on displacement streamlines. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 27(7):3264–3276, 2021. doi: 10.1109/TVCG.2020.2968911 1

[7] M. Patel and D. H. Laidlaw. Stress tensor field data, 2022. doi: 10.21227/06j7-kx51 1

[8] S. Popinet. Free computational fluid dynamics. *ClusterWorld*, 2(6), 2004. 1

[9] Q. Tian, Q. Fan, T. Witzel, M. N. Polackal, N. A. Ohringer, C. Ngamsombat, A. W. Russo, N. Machado, K. Brewer, F. Wang, K. Setsompop, J. R. Polimeni, B. Keil, L. L. Walk, B. R. Rosen, E. C. Klawiter, A. Nummenmaa, and S. Y. Huang. Comprehensive diffusion MRI dataset for in vivo human brain microstructure mapping using 300 mT/m gradients. *Scientific Data*, 9(1):7, 2022. doi: 10.1038/s41597-021-01092-6 1

[10] E. Zhang, H. Yeh, Z. Lin, and R. S. Laramee. Asymmetric tensor analysis for flow visualization. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 15(1):106–122, 2009. doi: 10.1109/TVCG.2008.68 9

[11] K. Zhao, S. Di, X. Lian, S. Li, D. Tao, J. Bessac, Z. Chen, and F. Cappello. SDRBench: Scientific data reduction benchmark for lossy compressors. In *IEEE International Conference on Big Data (Big Data)*, pp. 2716–2724, 2020. doi: 10.1109/BigData50022.2020.9378449 1