

Local Extraction of 3D Time-Dependent Vector Field Topology: Supplemental Material

Lutz Hofmann  and Filip Sadlo 

Heidelberg University, Germany

This document provides supplemental material for our paper “Local Extraction of 3D Time-Dependent Vector Field Topology”.

1. Algorithm Details

In this section, we provide details for our method, that were left out in the main paper for brevity. An overview of our method can be found in Algorithm 1.

1.1. Frame of Reference Computation

We use the method by Günther and Theisel [GT20] for computing a Galilean-invariant frame of reference. The least-squares problem

$$\int_{\mathbf{x} \in U} \|\nabla \mathbf{u}(\mathbf{x}, t) \mathbf{f}(\mathbf{x}, t) + \mathbf{u}_t(\mathbf{x}, t)\|^2 \rightarrow \min \quad (1)$$

is solved for $\mathbf{f}(\mathbf{x}, t)$ in its discretized form on a uniform grid. At grid node (i, j, k) , we sum over the discrete neighborhood

$$U = \{i - 10, \dots, i + 10\} \times \quad (2)$$

$$\{j - 10, \dots, j + 10\} \times \quad (3)$$

$$\{k - 10, \dots, k + 10\}, \quad (4)$$

$$\mathbf{A} = \sum_{\mathbf{x} \in U} \nabla \mathbf{u}(\mathbf{x}, t) \nabla \mathbf{u}(\mathbf{x}, t)^\top, \quad (5)$$

$$\mathbf{b} = \sum_{\mathbf{x} \in U} \nabla \mathbf{u}(\mathbf{x}, t)^\top \mathbf{u}_t(\mathbf{x}, t). \quad (6)$$

Algorithm 1 LCS extraction.

- 1: **for all** time steps $t_i = t_1, \dots, t_N$ **do**
 - 2: Compute Galilean-invariant frame of reference $\mathbf{w}(\mathbf{x}, t_i)$
 - 3: Extract parallel vectors lines $\mathbf{w}(\mathbf{x}, t_i) \parallel \boldsymbol{\eta}(\mathbf{x}, t_i)$
 - 4: Extract saddle-type critical points in $\mathbf{w}(\mathbf{x}, t_i)$
 - 5: Discard critical points that are contained in a PV line
 - 6: **end for**
 - 7: Track PV lines as surfaces over time
 - 8: Generate candidate lines from surface pathlines
 - 9: Track critical points as candidate lines over time
 - 10: Refine all obtained candidate lines toward HTs
 - 11: Determine seeding lengths along the HTs
 - 12: Extract LCS as streak surfaces seeded along the HTs
-

Nodes outside of the grid are assumed to have zero values in these equations. Finally, the linear system $\mathbf{A}\mathbf{f} = -\mathbf{b}$ is solved using a Householder QR decomposition with full pivoting. Like in the original work [GT20], the summation is performed over the entire grid using a three-dimensional summed area table for better computational performance.

1.2. Hyperbolic Trajectory Refinement

We employ the refinement procedure for distinguished hyperbolic trajectories proposed by Branicki and Wiggins [BW09], in its extended form as proposed for 2D flows by Hofmann and Sadlo [HS20]. We summarize the algorithm for 3D flows here.

An initial candidate $\tilde{\mathbf{x}}(t)$ is given as a polyline $(\mathbf{x}_0, t_0), \dots, (\mathbf{x}_N, t_N) \in \mathbb{R}^3 \times \mathbb{R}$ in space-time. Using the following iterative process, the polyline is refined toward a hyperbolic trajectory.

Evaluating the Jacobian of the flow along the candidate line, $\mathbf{J}_0 = \nabla \mathbf{u}(\mathbf{x}_0, t_0), \dots, \mathbf{J}_N = \nabla \mathbf{u}(\mathbf{x}_N, t_N)$, we compute the singular value decomposition

$$\mathbf{X}(t) = \mathbf{B}(t) e^{\boldsymbol{\Sigma}(t)} \mathbf{R}(t)^\top, \quad (7)$$

of the fundamental solution matrix $\mathbf{X}(t)$, which is the solution of the initial value problem

$$\frac{d}{dt} \mathbf{X}(t) = \mathbf{J}(t) \mathbf{X}(t), \quad \mathbf{X}(t_0) = \mathbb{I}. \quad (8)$$

To ensure numerical stability, Equation 8 is integrated in its decomposed form Equation 7, where the factors $\boldsymbol{\Sigma}(t)$ grow linearly instead of exponentially, thus avoiding floating point overflow. For this, the continuous SVD method by Dieci et al. [DE08] is employed. We use an embedded Runge–Kutta 4/5 scheme with dense output to obtain the solutions at the discrete time steps t_0, \dots, t_N . During integration, we monitor the maximum singular value $\max(\boldsymbol{\Sigma}(t))$. If this factor exceeds the floating point precision, we split the time interval at this point in time, and perform the refinement on each interval separately. For each split, we perform an additional refinement over a time interval containing the splitting point at the center and average the results at overlapping time steps.

From Equation 7, we compute the coordinate transforms $\mathbf{T}(t)$

and their inverses at the discrete time steps:

$$\mathbf{T}(t) = e^{(t-t_0)\mathbf{D}}\mathbf{R}(t_N)^\top \mathbf{R}(t)e^{-\mathbf{\Sigma}(t)}\mathbf{B}(t)^\top, \quad (9)$$

$$\mathbf{T}^{-1}(t) = \mathbf{B}(t)e^{\mathbf{\Sigma}(t)}\mathbf{R}(t_N)e^{(t_0-t)\mathbf{D}}, \quad (10)$$

$$\mathbf{D} = \frac{1}{t_N - t_0}\mathbf{\Sigma}(t_N). \quad (11)$$

Finally, using $\mathbf{w}(t) = \mathbf{T}(t)(\mathbf{x}(t) - \tilde{\mathbf{x}}(t))$, we solve the integral equations

$$\mathbf{w}_i(t) = \begin{cases} \int_{t_0}^t e^{d_i(t-s)}\mathbf{h}_i(\mathbf{w}(s),s)ds, & \text{if } d_i < 0, \\ -\int_t^{t_1} e^{d_i(t-s)}\mathbf{h}_i(\mathbf{w}(s),s)ds, & \text{otherwise,} \end{cases} \quad (12)$$

where \mathbf{w}_i and \mathbf{h}_i denote the i th component of \mathbf{w} and \mathbf{h} , and d_i the diagonal entries of \mathbf{D} , and \mathbf{h} is given by

$$\begin{aligned} \mathbf{h}(\mathbf{w}(t),t) = & \mathbf{T}(t)\mathbf{u}\left(\mathbf{T}^{-1}(t)\mathbf{w}(t) + \tilde{\mathbf{x}}(t),t\right) \\ & - \mathbf{T}(t)\nabla\mathbf{u}(\tilde{\mathbf{x}}(t),t)\mathbf{T}^{-1}(t)\mathbf{w}(t) - \mathbf{T}(t)\dot{\tilde{\mathbf{x}}}(t). \end{aligned} \quad (13)$$

The tangent of the candidate line, $\dot{\tilde{\mathbf{x}}}(t)$, is evaluated using central differences. Equation 12 is solved using a fixed-point iteration, where the integrals are repeatedly evaluated using the trapezoidal rule, until the distance between two iterations $\|\mathbf{w}^{(j+1)} - \mathbf{w}^{(j)}\|$ drops below a threshold τ_f . From this computation, we obtain an approximate hyperbolic trajectory as

$$\tilde{\mathbf{x}}_{\text{DHT}}(t) = \tilde{\mathbf{x}}(t) + \mathbf{T}^{-1}(t)\mathbf{w}(t). \quad (14)$$

Since the localization of the flow (Equation 8) is fixed, the process is iterated until $\|\tilde{\mathbf{x}}_{\text{DHT}}^{(i+1)}(t) - \tilde{\mathbf{x}}_{\text{DHT}}^{(i)}(t)\|$ drops below a threshold τ_i . Additionally to the approximate DHT $\tilde{\mathbf{x}}_{\text{DHT}}(t)$, we compute the Lyapunov vectors $\xi_k(t) = \mathbf{T}^{-1}(t)\mathbf{e}_k$ along the refined trajectory, which is used for streak seeding as discussed in the main paper.

In our experiments, we used $\tau_f = \tau_i = 10^{-10}$. The Runge–Kutta 4/5 integration was performed using a relative tolerance of 10^{-3} and absolute tolerance of 10^{-6} , with the mean time step $t_{i+1} - t_i$ as maximum step size Δ_{max} , and $\Delta_{\text{max}}/10$ as initial step.

2. Additional Evaluation

2.1. Stability under Perturbation

We analyze the stability of the DHT refinement under perturbation of initial candidates. The vector field is defined component-wise as $u_i(\mathbf{x},t) = d_i x_i + A_i \sin(\omega_i t)$, which has an analytical ground-truth DHT for all times [ISW02],

$$x_i(t) = -\text{sign}(d_i)A_i(d_i^2 + \omega_i^2)^{-\frac{1}{2}}\sin(\omega_i t + \arctan(\omega_i/d_i)). \quad (15)$$

We fix the parameters $d_1 = 5$, $d_2 = -5$, $d_3 = -4$, $\omega_1 = 2$, $\omega_2 = 3$, $\omega_3 = 4$, $A_1 = A_2 = A_3 = 1$, and sample the analytical field on a regular grid with 101^4 nodes over the domain $[-10, 10]^3 \times [1, 3]$. The path of the saddle-type critical point in the frame of reference defined by the feature flow flow is taken as initial candidate line. Initial candidate (Case A) and ground-truth DHT (Case B) are perturbed symmetrically (Case 1) by $\Delta = 0.1(t - 2)$, and asymmetrically (Case 2) by $\Delta = 0.1(t - 1)$, identically in all three coordinates. Figure 1a shows an overview of initial candidates and ground-truth.

Since the refinement relies on integration along the candidate line, its quality increases with available integration time, i.e., the

point-wise error is minimized at the center of the available time interval (Figure 1b). By considering the repelling (x , red in Figure 1) and attracting (y and z , green and blue in Figure 1) directions separately, we see, that the repelling direction is refined closer to the ground truth toward the beginning of the time interval, while the attracting directions are refined better toward the end of the time interval (Figures 1d and 1i–1l). The bends of the refined lines at the ends of the time interval are caused by zero available integration time for refinement ($t = 1$ for y - and z -coordinates, $t = 3$ for x -coordinate), thus the coordinates of the refined lines are identical to the candidate line there. Only in Case 2B (Figure 1l), where the ground truth was perturbed with zero perturbation at the beginning of the time interval, all three coordinates are refined best at the beginning of the time interval.

During streak integration, the streak manifolds are attracted toward the corresponding LCS (toward the attracting LCS in forward-time, and toward the repelling LCS in backward-time). This means, that the error of the streak manifolds decreases for long integration times, even when started from an erroneous hyperbolic trajectory. To demonstrate this, we compute a backward-time streak manifold seeded at the refined hyperbolic trajectory. The point-wise distance to the streak manifold seeded with the same offset from the ground-truth DHT is shown mapped to color (white low to black high) on the streak manifold at times $t_1 = 2.5$, $t_2 = 2.0$, and $t_3 = 1.2$ in Figures 1c and 1e–1h. We see, that in all cases, the error of the streak surface decreases with increasing integration time. The streak surface at time t_3 in Case 1A (Figure 1e) exhibits a large error, where it deviates from the ground-truth, possibly caused by the choice of seeding offset.

2.2. Half Cylinder Flow

This flow behind a half cylinder was computed for varying Reynolds numbers using the Gerris flow solver [Pop04] and is provided by Rojo and Günther [RG20]. The dataset, which was computed on an adaptive grid, was resampled to a uniform grid with dimensions $640 \times 240 \times 80 \times 151$ on the space-time domain $[-0.5, 7.5] \times [-1.5, 1.5] \times [-0.5, 0.5] \times [0, 15]$. We use two members of the ensemble with Reynold numbers $Re = 160$ and $Re = 320$ to analyze the behavior of our method at varying degree of turbulence. Seeding lengths for streak manifolds were determined using the method described in Section 4.3.2 of the main paper, using an FTLE percentage of 50%.

At $Re = 160$, the flow exhibits low turbulence. We employ our proposed Galilean-invariant frame of reference for varying sizes of neighborhoods (Equation 4). Direct inversion of the Jacobian to compute the feature flow field corresponds to an infinitesimally small neighborhood $N = 0$. Due to numerical noise, some initial candidates are missed, which leads to low quality in our extracted hyperbolic path surfaces (Figures 2a–2c). A too large neighborhood of radius $N = 41$ nodes, on the other hand, leads to overly smoothing and thus false negatives (Figures 2g–2i). Our proposed rather small neighborhood of $N = 10$ nodes, provides the best results (Figures 2d–2f). For comparison, VFT in the steady, displacement-invariant frame of reference [RG20] is not well aligned with LCS for both neighborhood sizes (the authors propose to use $N = 41$ in their work). However, these features provide good initial candidates

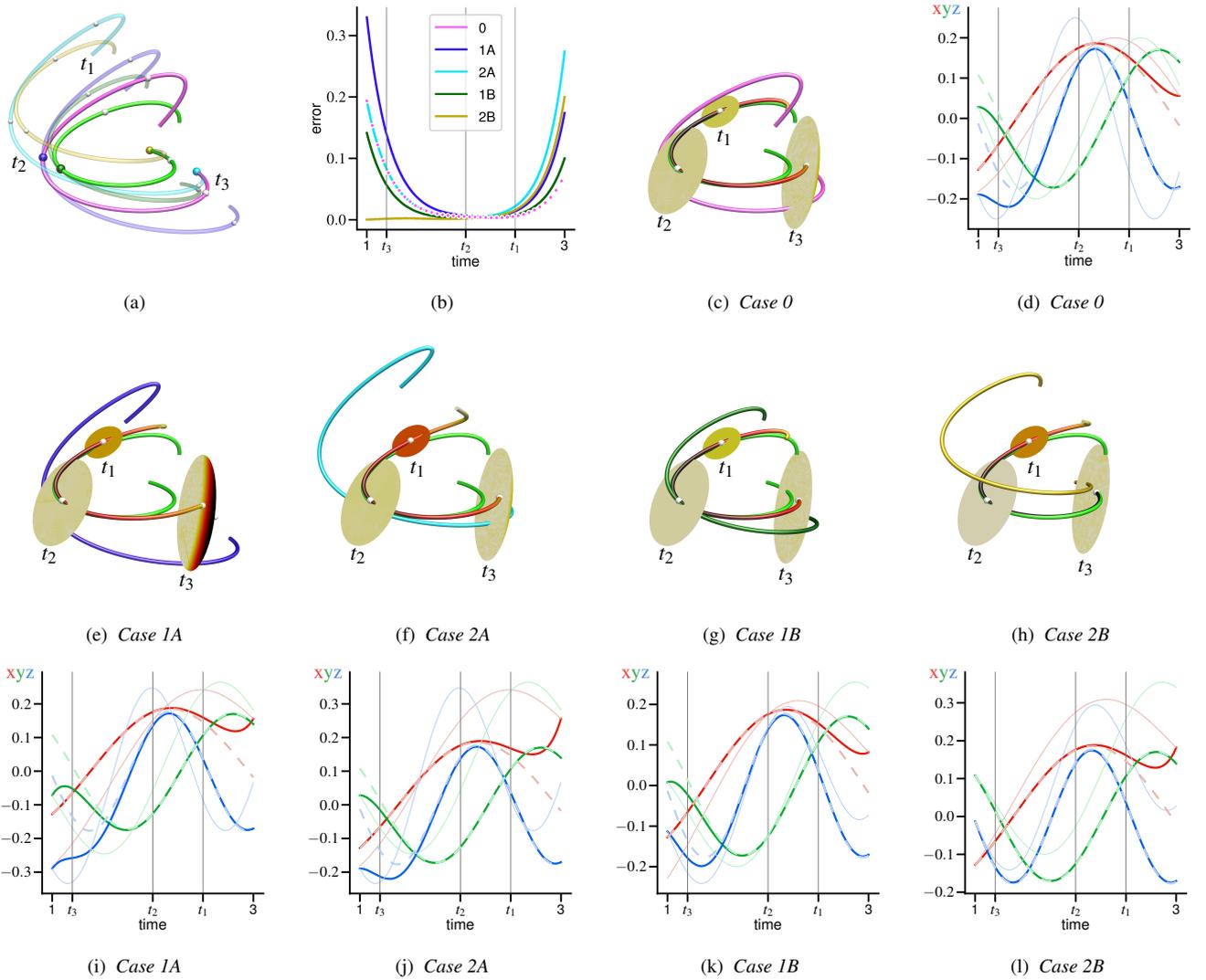


Figure 1: Perturbation of initial candidate (A) and ground-truth DHT (B), symmetrically (1) and asymmetrically (2). Overview shown in (a), with ground truth DHT (green), candidate line (magenta), and the four different perturbation cases (colors as in (b)), zero perturbation marked by colored sphere). (c),(e)–(h) Streak manifolds (surfaces, colored by error black to white) computed from the refined HT (colored by error black to white) further compensate errors with increasing advection times. (d),(i)–(l) Plots of the coordinate functions (x red, y green, z blue) of the initial candidate (thin desaturated), the refinement (solid saturated), and the ground truth (dashed). Note, that the dashed lines correspond to the green lines, the thin desaturated lines correspond to the single colored lines, and the solid lines correspond to the lines colored by error in the 3D views (c),(e)–(h).

for our refinement, which leads to a more extendedly extracted topology (Figures 2j–2l). Note, that this comes at additional computational costs at about 930 min, as well as a rather large memory requirement of about 360 GB of RAM to store the summed area table as well as the dataset and its derivatives, compared to the Galilean-invariant optimization, which took about 150 min and required about 50 GB of RAM.

At $Re = 320$, where the flow is more turbulent, all frames of reference fail to provide suitable initial candidates. We employ the same methods as in the previous case in Figure 3. Since all frame of references to not provide long enough initial candidates for our

refinement, the obtained hyperbolic path surfaces have large distances to the actual LCS intersections. The streak manifolds, however, are still attracted toward the LCS during integration, and are thus better aligned with them.

References

- [BW09] BRANICKI M., WIGGINS S.: An adaptive method for computing invariant manifolds in non-autonomous, three-dimensional dynamical systems. *Physica D: Nonlinear Phenomena* 238, 16 (2009), 1625–1657. 1
- [DE08] DIECI L., ELIA C.: SVD algorithms to approximate spectra of

- dynamical systems. *Mathematics and Computers in Simulation* 79, 4 (2008), 1235–1254. [1](#)
- [GT20] GÜNTHER T., THEISEL H.: Hyper-objective vortices. *IEEE Transactions on Visualization and Computer Graphics* 26, 3 (2020), 1532–1547. [1](#)
- [HS20] HOFMANN L., SADLO F.: Extraction of distinguished hyperbolic trajectories for 2D time-dependent vector field topology. *Computer Graphics Forum* 39, 3 (2020), 303–315. [1](#)
- [ISW02] IDE K., SMALL D., WIGGINS S.: Distinguished hyperbolic trajectories in time-dependent fluid flows: analytical and computational approach for velocity fields defined as data sets. *Nonlinear Processes in Geophysics* 9 (2002), 237–263. [2](#)
- [Pop04] POPINET S.: Free computational fluid dynamics. *ClusterWorld* 2, 6 (2004). [2](#)
- [RG20] ROJO I. B., GÜNTHER T.: Vector field topology of time-dependent flows in a steady reference frame. *IEEE Transactions on Visualization and Computer Graphics* 26, 1 (2020), 280–290. [2](#), [5](#), [6](#)

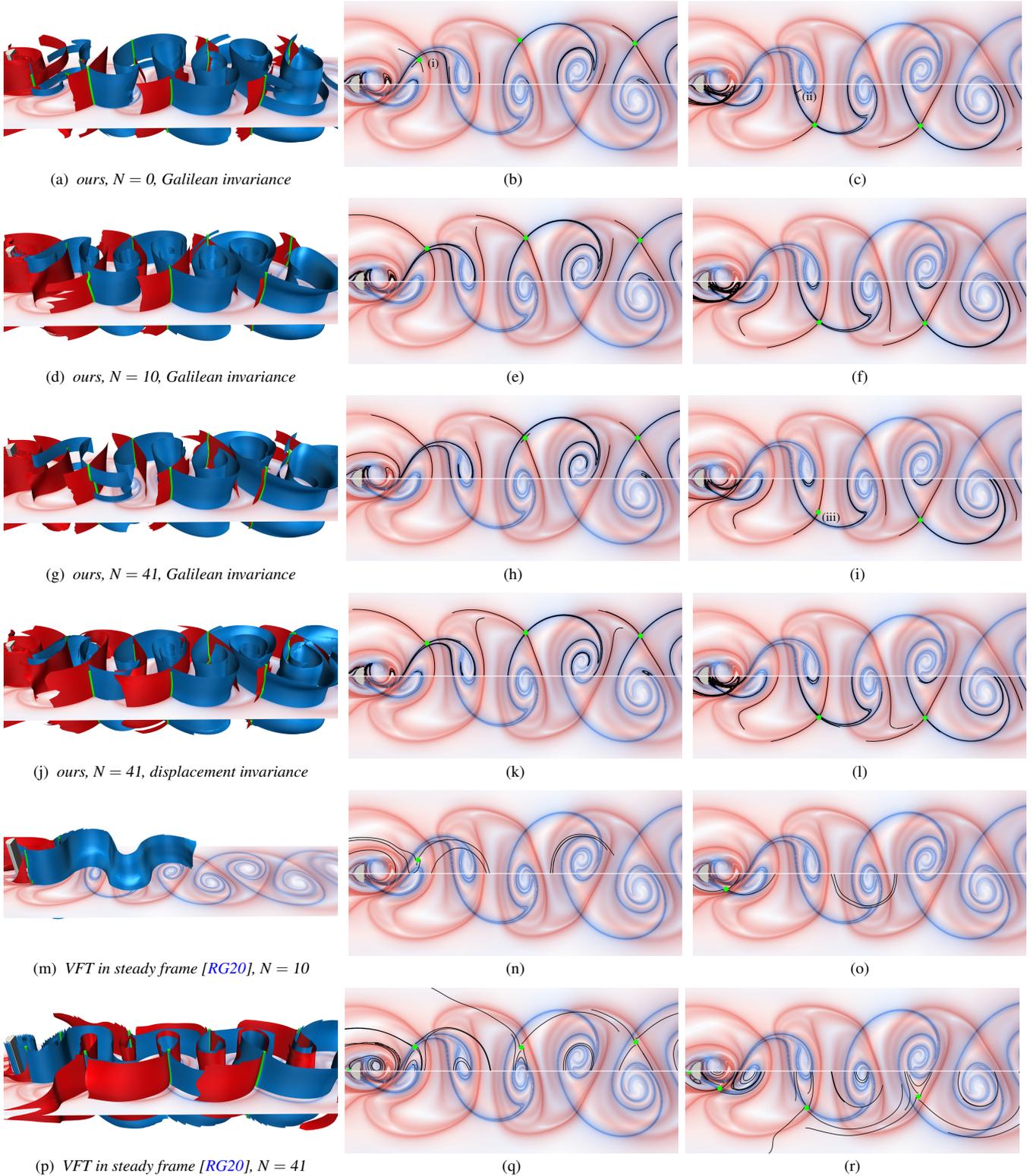


Figure 2: Half Cylinder Flow with modest turbulence at $Re = 160$. Black lines in mid and right column represent sections of manifolds at shown slice in left column. With our proposed Galilean-invariant extraction of candidate lines, direct inversion of the Jacobian, (a)–(c), suffers from numerical noise, and thus yields worse initial candidates (i) than using a neighborhood of N nodes. A too large neighborhood with $N = 41$, (g)–(i), has a too strong smoothing effect (iii). We thus prefer $N = 10$, (d)–(f). While steady VFT in the displacement-invariant frame of reference [RG20] is not well aligned with the LCS, (m)–(r), it provides good initial candidates for our method, (j)–(l).

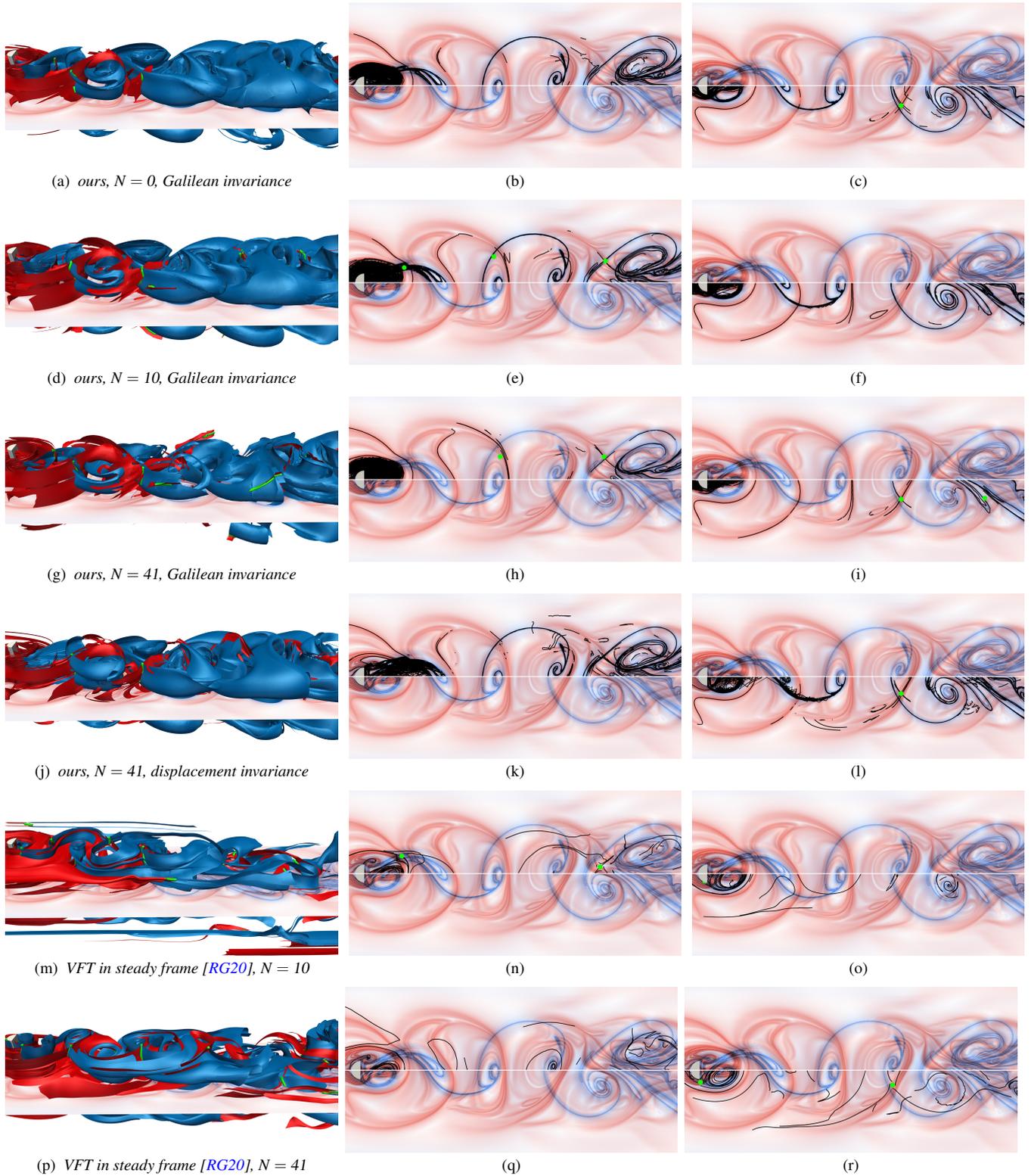


Figure 3: Half Cylinder Flow with higher turbulence at $Re = 320$, same methods applied as in Figure 2. Both the Galilean-invariant and displacement-invariant [RG20] frames of reference fail to extract suitable candidate lines in this turbulent dataset. Our methods with $N = 10$ again performs best, but also still misses large parts of the LCS. Nevertheless, streak integration is attracted by the respective LCS for long integration times, thus partially correcting initial errors.