

Objective Observer-Relative Flow Visualization in Curved Spaces for Unsteady 2D Geophysical Flows

Peter Rautek, Matej Mlejnek, Johanna Beyer, Jakob Troidl,
Hanspeter Pfister, Thomas Theußl, and Markus Hadwiger

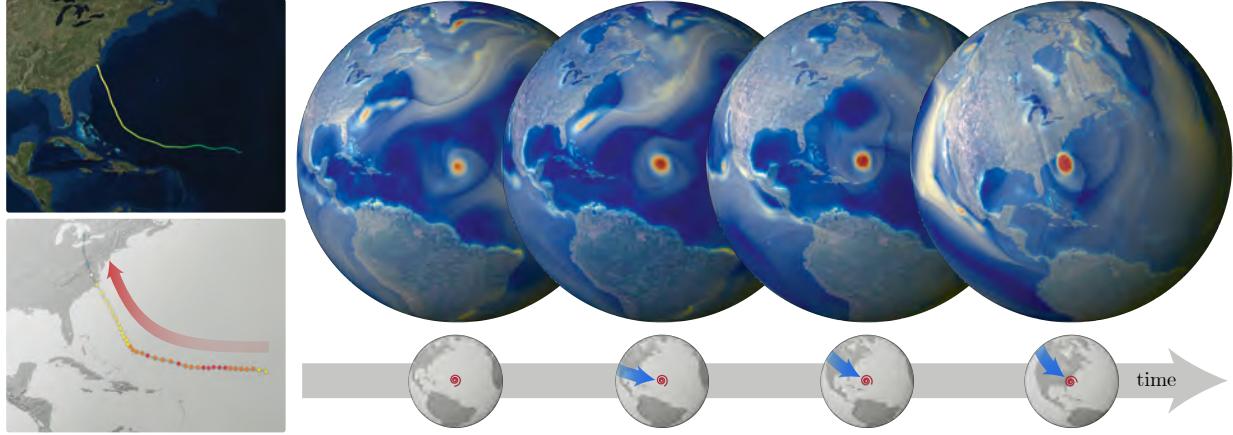


Fig. 1. **Observer motion relative to the time evolution of features:** Hurricane Isabel in a time-dependent global wind data set. (Bottom left) The *actual* path of Isabel (data from Wikipedia). (Top left) Our observer field \mathbf{u} automatically follows the motion of Isabel without explicit tracking of its path. The shown path is simply a path line of \mathbf{u} . (Right) Feature-relative visualization, focused on Isabel in the center, enabling analysis of its time evolution “in place.” The hurricane appears steady, with the Earth moving inversely underneath.

Abstract—Computing and visualizing features in fluid flow often depends on the observer, or reference frame, relative to which the input velocity field is given. A desired property of feature detectors is therefore that they are objective, meaning independent of the input reference frame. However, the standard definition of objectivity is only given for Euclidean domains, and cannot be applied in curved spaces. We build on methods from mathematical physics and Riemannian geometry to generalize objectivity to curved spaces, using the powerful notion of symmetry groups as the basis for defining objectivity. From this, we develop a general mathematical framework for computing objective observer fields for curved surfaces, relative to which other computed measures become objective. An important property of our framework is that it works intrinsically in 2D, instead of in the 3D ambient space. This enables a direct generalization of the 2D computation of observer fields via optimization from flat domains to curved domains, without having to perform optimization in 3D. We specifically develop the case of unsteady 2D geophysical flows given on spheres, such as the Earth. Observer fields on curved surfaces enable objective feature computation as well as the visualization of the time evolution of scalar and vector fields, such that the automatically computed reference frames follow moving structures like vortices in a way that makes them appear to be steady.

Index Terms—Flow visualization, observer fields, frames of reference, objectivity, symmetry groups, intrinsic covariant derivatives

1 INTRODUCTION

Flow visualization is concerned with velocity vector fields that describe fluid motion. Since the definition of velocity is the infinitesimal change of spatial position with respect to time, this description depends on

the notion of what is considered to be “the same position” over time. Mathematically, this is the notion of a *frame of reference*, or an *observer*. A frame of reference is a way of referencing positions in space over time, i.e., a time-dependent transformation of space, with respect to which derivatives are to be taken. (Not an observer as an actual person.)

For this reason, flow visualization as well as the computation of flow features, such as vortices, in general inherently depend on the chosen frame of reference. An important related concept therefore is *invariance* with respect to certain *changes of reference frame*. Galilean invariance [27] refers to frame changes with constant relative velocity, which is inherently related to *inertial frames* in physics [63]. Objectivity [30] considers frame changes with time-dependent translations and rotations. The standard definition used in visualization and continuum mechanics is the one of Truesdell and Noll [65], which has received significant attention for flow fields since the work of Haller [30]. A major basic limitation of this definition, however, is that it is given only for Euclidean space and cannot be used in curved spaces.

One area where objectivity is particularly relevant is that of *geophysical flows*, such as in the Earth’s atmosphere or oceans. The natural reference frame, in this case, is rotating [50, p.14], i.e., non-inertial, and Galilean invariance is not well-suited. As is common in flow visualization, we only consider motions by themselves (*kinematics*), not the

• Peter Rautek, Matej Mlejnek, and Markus Hadwiger are with King Abdullah University of Science and Technology (KAUST), Visual Computing Center, Thuwal, 23955-6900, Saudi Arabia.

E-mail: {markus.hadwiger, matej.mlejnek, peter.rautek}@kaust.edu.sa

• Johanna Beyer and Hanspeter Pfister are with Harvard University, Cambridge, MA, USA. E-mail: {jbeyer, pfister}@seas.harvard.edu

• Jakob Troidl is with King Abdullah University of Science and Technology (KAUST), Visual Computing Center, Thuwal, 23955-6900, Saudi Arabia, and TU Wien, Vienna, Austria. E-mail: jakob.troidl@googlemail.com.

• Thomas Theußl is with King Abdullah University of Science and Technology (KAUST), Core Labs, Thuwal, 23955-6900, Saudi Arabia. E-mail: thomas.theussl@kaust.edu.sa.

forces generating them (*dynamics*). From this perspective, accelerating reference frames can be used just as easily as inertial frames, because the differences in the forces acting are irrelevant. For example, in Fig. 2 we consider an airplane as an observer measuring motion relative to its own reference frame. Naturally, this does not change a phenomenon such as a cyclone, but it does change all relative velocity measurements.

Although the Earth's surface is embedded in a flat 3D space (ignoring relativistic effects), it is beneficial to be able to view the domain where a geophysical flow field is defined from an *intrinsic*, curved 2D perspective. This not only allows for simplified computations in 2D instead of in 3D, or considering different altitude or depth layers separately; it also corresponds perfectly to the reference frame of circular orbits, such as those of satellites or airplanes moving along great circles around the Earth, as shown in Fig. 2. For visualization purposes, it is also common to visualize spherical layers. Moreover, scientists have even discovered that some geophysical fluid systems, such as aspects of our atmosphere or even that of Jupiter [28, 56], can sometimes be modeled in a more realistic way in 2D than in 3D, because the atmosphere is so much thinner than wide. For this reason, it can be crucial to perform computations in 2D, because the physical behavior of 2D turbulence can be radically different from 3D turbulence [22, 34, 39].

With this motivation, we develop general techniques for flow visualization and computation of flow features in curved spaces. For practical relevance, we illustrate results for geophysical flows on spheres. We generalize the notion of observer fields from Euclidean space [29] to curved spaces. This enables following flow features such as vortices, and visualizing the corresponding time evolution of scalar and vector attributes “in place.” For example, Fig. 1 shows hurricane Isabel (2003) in a time-dependent global wind simulation. The computed observer field automatically follows the motion of the hurricane. Fig. 1 (right) shows feature-relative visualization, where the observer field makes the motion of Isabel appear to be steady. This enables “feature-centric” analysis, without having to explicitly follow the motion of the hurricane.

1.1 Observer Fields and Observed Time Derivatives

An observer field $\mathbf{u}(x, t)$ is a time-dependent velocity field that, instead of describing the motion of particles, describes the motion of a continuous field of observers [29]. The *path lines* of the observer field correspond to the *world lines* [3, p. 8] of individual spatial locations in the space of an observer that are viewed as being “the same point” over time. If an observer field \mathbf{u} describes a rigid motion, it is a *Killing vector field* [52]. Figs. 2 (a,b,c) depict three time steps of observer fields on a sphere. Fig. 3 shows one example input vector field \mathbf{v} describing a rotating vortex, as it is seen by four *different* observer fields \mathbf{u} .

The *observed time derivative* corresponding to a given $\mathbf{u}(x, t)$ measures the differential change of an arbitrary input field $\mathbf{v}(x, t)$ with respect to the observer motion described by \mathbf{u} . This concept was introduced in the context of Euclidean space [29]. We define this time derivative, on *any* differential manifold M , as the differential operator

$$\frac{\mathcal{D}}{\mathcal{D}t} := \frac{\partial}{\partial t} + \mathcal{L}_{\mathbf{u}}. \quad (1)$$

Here, $\mathcal{L}_{\mathbf{u}}$ denotes the *Lie derivative* [21, Ch. 4] with respect to the field \mathbf{u} . In order to “steady” an input vector field \mathbf{v} , e.g., to be able to compute and follow features, we compute observer fields \mathbf{u} by *minimizing* the derivative $(\mathcal{D}/\mathcal{D}t)(\mathbf{v} - \mathbf{u})$. If Fig. 3 is interpreted as showing four *different* input fields \mathbf{v} , the *observer-relative* visualization with respect to the optimized field \mathbf{u} will always look like Fig. 3 (d).

1.2 Mathematical Framework for Curved Spaces

We develop a novel *fully intrinsic* mathematical framework for 2D flow visualization and feature computation in curved spaces. Our major contribution is the generalization of a prior Euclidean framework [29] for computing objective observer fields to curved spaces. We compute these fields by solving an energy minimization problem, making unsteady input flows as steady as possible. We first introduce a generalized notion of objectivity for curved spaces. While equivalent to the standard one in Euclidean space [65], ours is fully intrinsic, using the concepts of *continuous symmetry groups* and *group actions* on

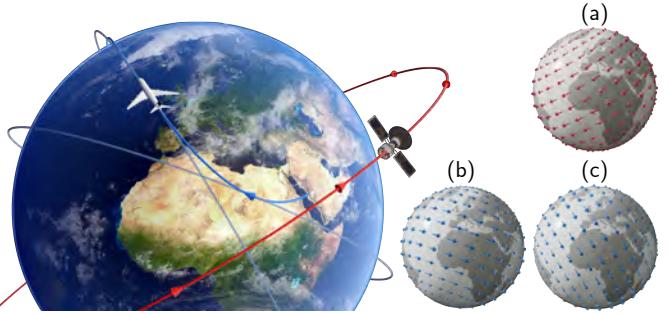


Fig. 2. Observers in curved spaces. An airplane (blue path) and a satellite (red path) are moving relative to the Earth. At each instant in time, with respect to the Earth's surface both fly in the direction of a geodesic (a great circle). Considering the observer (airplane or satellite) as stationary, the instantaneous relative motion of the Earth is a Killing vector field on the sphere, in the *opposite* direction (right). (a) The Killing field for the satellite is a steady field (assuming the satellite ground track is always the same geodesic); (b,c) The field for the airplane is unsteady: Each instant in time is determined by a different tangential velocity.

manifolds from mathematical physics. From general symmetry groups, we specialize to observer transformations modeled as elements of the *isometry group* of a manifold M . To simplify computations, we work with infinitesimal isometries, given explicitly by vector fields on M .

To be able to achieve this, we derive the differential geometric operators needed to compute observer fields in *curved spaces*: (1) Observed time derivatives; (2) Approximate Killing fields; and (3) The velocity gradient tensor $\nabla \mathbf{v}$ as a *covariant derivative* [21, Ch. 9], instead of the Jacobian that is commonly used in flow visualization.

Methodology. Our generalization of objectivity and observer fields to curved spaces requires mathematical machinery from Riemannian geometry and mathematical physics that is not common in the flow visualization literature. For this reason, we summarize background and our notation in the beginning, and provide extensive supplementary appendices. These techniques enable deriving a powerful generalization of previous flow visualization techniques from flat space to curved spaces in a clean and intrinsic manner. We believe that this powerful methodology can be very useful for flow visualization in general, and that it is worthwhile to introduce it to the flow visualization literature.

2 RELATED WORK

The framework presented here extends the work of Hadwiger et al. [29] from Euclidean space \mathbb{R}^n to curved spaces, focusing on curved two-manifolds. We also model a collection of observers by an *observer velocity field* describing their motion, which we also compute by solving a least-squares problem such that (1) it is objective, and (2) the input flow field becomes as steady as possible relative to it. However, the generalization to curved spaces is conceptually and technically challenging. We propose a fully *intrinsic* generalization, working “inside” the curved space. This results in natural and efficient generalizations, and facilitates computing 2D observer fields via 2D optimization.

Flow visualization on curved surfaces. All major flow visualization techniques have been extended from flat space to curved surfaces. For example, LIC [13] and variants [42, 55, 60] have been extended to curved surfaces, either for parametric surfaces [20], by texturing triangle meshes [7, 59, 62], or by computing LIC in screen space [6, 35]. Image-based texture advection techniques [36, 67] have been extended as well, such as image-based flow visualization on surfaces [38, 68]. Image space techniques typically project vector fields from 3D to 2D screen space for visualization. We refer to the survey by Edmunds et al. [19]. Texture atlases have also been used for flow visualization, for example flow charts [43]. We similarly use multiple coordinate charts.

Vector field design on surfaces. We extensively use differential properties of manifolds and vector fields. These are particularly important for computations with vector fields on surfaces, such as vector field design on surfaces [14, 15, 71]. We refer to the survey of Peng and Laramée [51]. We would like to particularly point out the use of geodesic polar maps and parallel transport along geodesics in the work

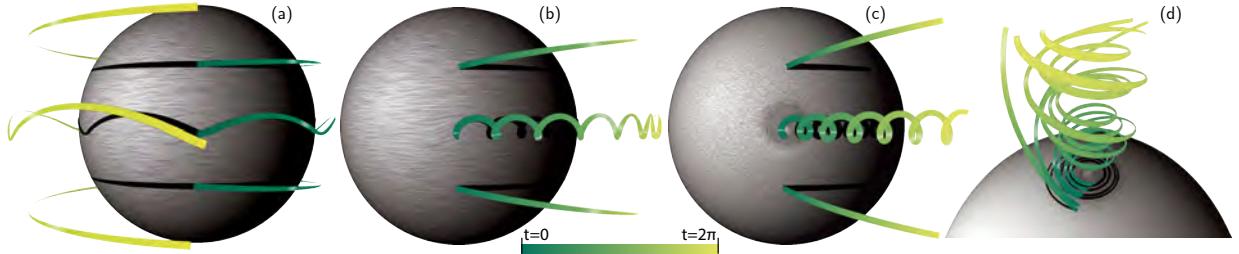


Fig. 3. A rotating vortex seen by four different observers. The observers rotate *leftward* around the sphere (here, around a vertical axis), with angular velocities $\omega = -2\pi, -\pi/4, -\pi/8, 0 [s^{-1}]$ (a,b,c,d). All path lines are *on* the sphere (the depicted shadows), but we visualize the progression of time as increasing radial distance and color (see inset). If these are seen as four *different* input fields \mathbf{v} , minimizing the observed time derivative $\mathcal{D}/Dt(\mathbf{v} - \mathbf{u})$ gives observer fields \mathbf{u} with *rightward* rotations $\omega = 2\pi, \pi/4, \pi/8, 0 [s^{-1}]$. (d) Observer-relative visualization is the *same* for all four cases.

of Palacios and Zhang [49]. The notion of covariant derivative, which we employ in our work, is inherently related to parallel transport.

Covariant derivatives have been used for mesh processing [17] and computing vector [16] and direction [69] fields on surfaces. An important focus is often their discretization on triangle meshes [5], e.g., using discrete exterior calculus [17]. We discretize on triangle vertices in 2D charts, and particularly focus on fully intrinsic 2D computations.

Interpolation. We use standard barycentric interpolation in triangles. However, spherical barycentric coordinates, i.e., mean value coordinates on the sphere, could also be used for better interpolation [37].

Killing vector fields on (Riemannian) manifolds [18] correspond to their infinitesimal isometries. Their flows correspond to the *intrinsic* isometries of a manifold. Basics of Killing fields for curved surfaces are discussed by Ben-Chen et al. [8] and Solomon [57]; a more advanced treatment is given by Petersen [52]. Approximate Killing fields have been used to compute approximate intrinsic isometries of curved surfaces [8], or to design approximate Killing fields on meshes [4, 5]. As-rigid-as-possible shape interpolation methods [1] work directly with isometries instead of with derivatives, and are computed extrinsically.

Observers. The concept of an observer as well as objectivity in Euclidean space has been recognized to be of importance in fluid mechanics [30] as well as in flow visualization [27]. A standard reference in continuum mechanics is Truesdell and Noll [65], whose definition for objectivity in Euclidean space is often used [24, 30, 31, 33, 47, 53].

Vortex detection and frame invariance. Vortices are an important topic in fluid mechanics [54] and flow visualization [27]. Many methods were originally defined for steady flow, such as Sujudi and Haimes [61], and later extended to unsteady flow [70]. Galilean invariance of these criteria is considered to be important [27]. Several vortex detection methods are Galilean-invariant, such as Weinkauf et al. [70], or the method of Bujack et al. [12]. The latter jointly considers multiple observers for 2D flow fields. Specific vortex detectors have been developed to be objective by design [30, 31]. Günther et al. [24, 53] present objective methods for Euclidean space that are generic by building on optimization, making non-objective vortex detectors objective. Other kinds of invariance have been defined particularly for flow fields, such as rotation invariance [25], or hyper-objective vortices [26].

Flow decomposition methods, such as variants of the Helmholtz-Hodge decomposition [9, 10], can also be used to remove background flow, corresponding to the harmonic component [11]. These decompositions can be computed on curved surfaces [64]. However, they apply only to time-independent fields, or alternatively they have to be applied to each time step individually, without considering time derivatives.

3 BACKGROUND AND NOTATION

Our framework enables all computations, most importantly global optimization of *observer fields*, to be 2D for curved surfaces, instead of requiring 3D ambient space computations. We perform all computations in an *atlas* of 2D *coordinate charts*. See Fig. 4. Moreover, this also simplifies visualization computations on curved surfaces by working directly in charts, for example for path line integration or Line Integral Convolution (LIC). See Fig. 3. However, already a surface as simple as a sphere cannot be covered by a single chart. All computations in charts therefore need to behave in a manner that is independent of coordinates. We ensure this invariance by using general tensor methods [23].

Vector fields. A smooth vector field \mathbf{v} on a manifold M is a smooth function giving a vector $\mathbf{v}(x)$ at every point $x \in M$ on the manifold, i.e.,

$$\begin{aligned} \mathbf{v}: M &\rightarrow TM, \\ x &\mapsto \mathbf{v}(x). \end{aligned} \quad (2)$$

TM refers to the *tangent bundle* of M , the manifold of all tangent spaces of M , and a vector field is also referred to as a *section* of TM . Most important, $\mathbf{v}(x)$ is an element of the tangent space at x , denoted by $T_x M$. Where no confusion arises, we also denote a single vector $\mathbf{v}(x)$ by \mathbf{v} .

Coordinate charts. Coordinates on a curved n -dimensional manifold M are not given by coordinate vectors, which is only possible in linear spaces, but by n *coordinate functions* $x^i: M \rightarrow \mathbb{R}$, referring each point $x \in M$ to a point in \mathbb{R}^n , e.g., the functions x^1, x^2 (or u, v), for charts $U \subset \mathbb{R}^2$. Basis vectors live in each tangent space $T_x M$, at $x \in M$.

Bases. Vectors are geometric objects independent of any chosen basis. In order to refer a vector \mathbf{v} to a basis, we expand it in components as $\mathbf{v} = v^i \mathbf{e}_i$, where $\{\mathbf{e}_i\}$ is a basis for $T_x M$, and, in general, the $T_x M$ at different $x \in M$ have different bases. We employ the Einstein summation convention [21, p.59], implying summation over indices occurring twice (once “upstairs” and “downstairs” each), e.g., $v^i \mathbf{e}_i := \sum_i v^i \mathbf{e}_i$.

Dual bases. We will also need the concept of a *dual basis* $\{\omega^i\}$, where $\omega^i(\mathbf{e}_j) = \delta_j^i$, with the Kronecker delta $\delta_j^i = 1$ if $i = j$, and zero otherwise. Each ω^i is a *covector*, or *1-form*, which is a linear function mapping a vector to a scalar. The dual basis $\{\omega^i\}$ reads off the components of a vector \mathbf{v} referred to $\{\mathbf{e}_i\}$, such that $\mathbf{v} = \omega^i(\mathbf{v}) \mathbf{e}_i$.

Metrics. Lengths and angles of vectors are determined in an intrinsic manner by defining a (Riemannian) metric \mathbf{g} on M [18, p.35], which is a second-order tensor field defining an inner product $\langle \mathbf{v}, \mathbf{w} \rangle := \mathbf{g}(\mathbf{v}, \mathbf{w})$ on each $T_x M$, varying smoothly over M . The vector norm is then defined as $\|\mathbf{v}\| := \langle \mathbf{v}, \mathbf{v} \rangle^{1/2}$. Computations with \mathbf{g} can use components $g_{ij} := \langle \mathbf{e}_i, \mathbf{e}_j \rangle$ referred to $\{\mathbf{e}_i\}$, computing $\langle \mathbf{v}, \mathbf{w} \rangle$ as $g_{ij} v^i w^j$. We also extensively use *covariant derivatives* compatible with a given metric \mathbf{g} .

Velocity gradients. An important derived quantity of a velocity vector field \mathbf{v} is the *velocity gradient tensor field* $\nabla \mathbf{v}$. In flow visualization, $\nabla \mathbf{v}$ is often seen as a Jacobian $\partial_j v^i$ of partial derivatives in Cartesian coordinates. However, these coordinates do not exist on curved manifolds, and in general $\nabla \mathbf{v} \neq \partial_j v^i$, because the latter is not tensorial [21, p.241].

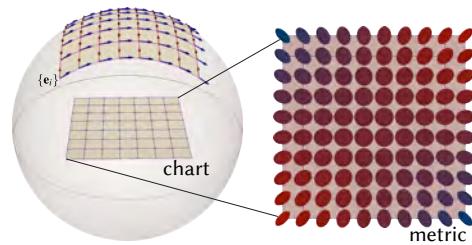


Fig. 4. Our intrinsic framework employs a perspective from *within* a curved manifold M . Thus, all computations on curved surfaces are 2D computations in 2D *coordinate charts*. Each point $x \in M$ has intrinsic differential properties: A (Riemannian) *metric*, and *covariant derivative* operators. A vector field \mathbf{v} is a vector $\mathbf{v}(x)$ in each tangent space $T_x M$. Per chart, each $T_x M$ has a coordinate basis $\{\mathbf{e}_i\}$, and a dual basis $\{\omega^i\}$. The symmetric metric tensor field is visualized with glyphs on the right.

We note that even in flat \mathbb{R}^2 with polar coordinates, $\partial_j v^i$ is not a tensor. This has important implications, such as that a tensor that vanishes in one coordinate system *must* vanish in *all* coordinate systems [23, p.82]. This is not true for $\partial_j v^i$, and thus the Jacobian behaves only as it should in special cases, such as by restriction to *Cartesian tensors* [2, Ch.2].

Our framework requires a more general definition of $\nabla \mathbf{v}$ as the *covariant derivative* $\bar{\nabla}_j v^i$, which is standard in mathematical physics. However, we have found no explicit use of this concept in the flow visualization literature. We derive this notion for our purposes in Sec. 6.

Approximate observer Killing fields. In general, an observer field is allowed to be given by an arbitrary vector field \mathbf{u} . However, it makes sense to restrict the motion described by \mathbf{u} to be “as rigid as possible.” We therefore optimize observer fields such that they correspond to *approximately isometric* deformations, i.e., to “almost-rigid” motions. Similar to Hadwiger et al. [29], we do this by minimizing the rate-of-deformation tensor of \mathbf{u} . However, in curved space M , this computation requires using the covariant derivative $\nabla \mathbf{u}$ and the metric \mathbf{g} on M .

4 OBSERVERS IN CURVED SPACES

The first crucial notion for us to define is the meaning of an observer in a curved space. In analogy with the standard notion of observer transformations in Euclidean space [65], it is natural to consider observer transformations to be time-dependent *isometries* of the underlying space. From an intrinsic perspective, this means that we consider the *intrinsic isometries* of a smooth manifold M with a (Riemannian) metric \mathbf{g} . This means that we consider *geodesic distances*, as determined by \mathbf{g} , instead of Euclidean distances. Our observer transformations are therefore *time-dependent intrinsic distance-preserving* transformations of the underlying space M . More generally stated, this means

Remark. *We employ a (non-relativistic) concept of frames of reference postulating that all observers agree on the pairwise spatial geodesic distances of events happening at the same time, and all observers also agree on the time difference between events (i.e., time is absolute).*

Mathematically, we model time-dependent observer transformations as paths $t \mapsto g(t) \in G$, through a Lie group $G := \text{Isom}(M)$ chosen as the *isometry group* of the manifold M with metric \mathbf{g} . Furthermore, computationally it is easier to work directly with derivatives of $g(t)$, with a corresponding path $t \mapsto X(t) \in \mathfrak{g}$ through the Lie algebra $\mathfrak{g} = \text{isom}(M)$ of the Lie group G , i.e., the Lie algebra of *infinitesimal isometries*.

4.1 Observers on the Sphere

Figs. 2 and 5 illustrate this idea for $M = \mathbb{S}^2$, the standard two-sphere. The motion of the Earth relative to a fixed observer, such as the airplane in Fig. 2, is given by a path of isometries $t \mapsto g(t) \in \text{SO}(3)$, which here are different rotations of the Earth. At any instant in time t , the derivative of this path corresponds to a Lie algebra element $X(t) \in \mathfrak{so}(3)$, which defines an infinitesimal isometry of M . However, $X(t)$ is just an anti-symmetric matrix (Fig. 5, bottom). The specific corresponding infinitesimal isometry of the sphere is given by an isomorphism between $\mathfrak{so}(3)$ and the Lie algebra of *Killing vector fields* on the sphere (Fig. 5, top). This isomorphism is mathematically described by a *Lie algebra action* (see App. J). In this way, the infinitesimal isometry on the surface of the Earth is the Killing vector field $\mathbf{x}(t)$ corresponding to $X(t)$. At each point $x \in M$, this gives a relative velocity vector $\mathbf{x}(x, t)$ (Fig. 5, top). We also know that \mathbf{x} describes an infinitesimal isometry of M , because $\nabla \mathbf{x}$, as the covariant derivative of \mathbf{x} , is anti-symmetric. In fact, any Killing field \mathbf{x} is uniquely determined by its value $\mathbf{x}(x)$ and covariant derivative $(\nabla \mathbf{x})_x$ at one point $x \in M$ [52, Proposition 8.1.4].

4.2 Observer Fields

Generalizing further from the concept of one observer, as one time-dependent isometric transformation of space, we want to be able to describe an *observer field* (many observers) on a manifold M as a vector field \mathbf{u} on M , where \mathbf{u} does not necessarily correspond to an exact (infinitesimal) isometry. One of the most important properties of such an observer field \mathbf{u} is that we can compute it such that it minimizes the *observed time derivative* $(\mathcal{D}/\mathcal{D}t)(\mathbf{v} - \mathbf{u})$, given an arbitrary input vector field \mathbf{v} , whose time-dependent behavior we want to “follow.”

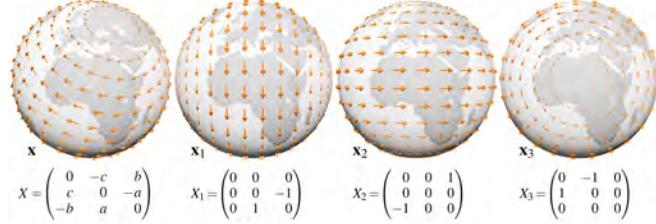


Fig. 5. **Killing vector fields** give the infinitesimal isometries of a manifold M , here the sphere \mathbb{S}^2 . The Lie algebra elements $X, X_i \in \mathfrak{so}(3)$ (bottom) generate the Killing fields \mathbf{x}, \mathbf{x}_i on M (top) through their *Lie algebra action*. Like the matrices X_i , the Killing fields \mathbf{x}_i are linearly independent, forming a basis of the Lie algebra of Killing fields on \mathbb{S}^2 . That is, as we can expand $X = aX_1 + bX_2 + cX_3$, we likewise get $\mathbf{x} = a\mathbf{x}_1 + b\mathbf{x}_2 + c\mathbf{x}_3$, where the latter means point-wise addition of vectors in each $T_x M$ at each $x \in M$.

The second important property of an observer field \mathbf{u} is that if \mathbf{u} is indeed a Killing vector field, it in fact describes a single observer, as defined above. Typically, we want to optimize the vector field \mathbf{u} such that it describes observers that are as similar as possible. We do this by quantifying how much (or how “fast,” in terms of spatial distance) the observers differ from one global isometry, by computing the *Killing energy* of the vector field \mathbf{u} over M . This is the integral

$$\int_M E\mathbf{u} \, dM. \quad (3)$$

The point-wise energy term $E\mathbf{u}$ is given by Eq. 29. Eq. 3 is a scalar measure for how much \mathbf{u} differs from an infinitesimal isometry on M . See Hadwiger et al. [29] for a more detailed rationale in Euclidean space. In this work, we now generalize this concept to curved spaces.

4.3 Feature-Relative Visualization

Fig. 1 and Fig. 6 illustrate how the computation of an observer field on the surface of the Earth enables focusing a visualization on a feature such as a hurricane. In this example, the input field $\mathbf{v}(x, t)$ is given by a time-dependent global wind data set from the European Copernicus project. The top row of Fig. 6 shows path lines following the motion of the hurricane over time. In this frame of reference, it is hard to compare features of the hurricane “in place,” due to its motion. In contrast, the bottom row of Fig. 6 shows a visualization that is relative to an automatically computed observer field $\mathbf{u}(x, t)$, following the motion of the hurricane. Since now a “feature-relative” observer is used, the hurricane has become steady. (Now the Earth is moving, relative to the new observer, underneath the hurricane.) Path lines seeded at the same positions as before now clearly highlight the vortex of the hurricane.

5 OBJECTIVITY IN CURVED SPACES

Given a particular observer transformation $t \mapsto g(t)$, it is natural to define that a tensor field is *objective* if it simply undergoes *the same* transformation as the observer. Because we model all possible observer transformations as elements of a *continuous symmetry group* G , which is a Lie group, we can now rigorously formulate a generalized concept of objectivity for arbitrary manifolds M in a fully intrinsic manner, using the concept of *group action* on M with a Lie group element g .

We first summarize the definition of Truesdell and Noll [65], which is standard in continuum mechanics [33, 47], but is restricted to Euclidean space. We then introduce our completely general definition that includes curved spaces, but is equivalent in Euclidean space. App. D derives the definition of Truesdell and Noll more generally, illuminating the correspondence between their approach and ours in more detail.

5.1 Objectivity in Euclidean space

Truesdell and Noll [65, p.41] define objectivity with respect to a change of reference frame in \mathbb{R}^3 via the distance-preserving transformation

$$\begin{aligned} \mathbf{x}^* &= \mathbf{c}(t) + \mathbf{Q}(t)\mathbf{x}, \\ t^* &= t - a. \end{aligned} \quad (4)$$

$\mathbf{Q}(t)$ is a proper orthogonal tensor (a rotation), $\mathbf{c}(t)$ is a point (position vector), and $a \in \mathbb{R}$. This transformation assumes *absolute time*. It is thus sufficient to consider $a = 0$, disregarding time shifts, giving $t^* = t$.

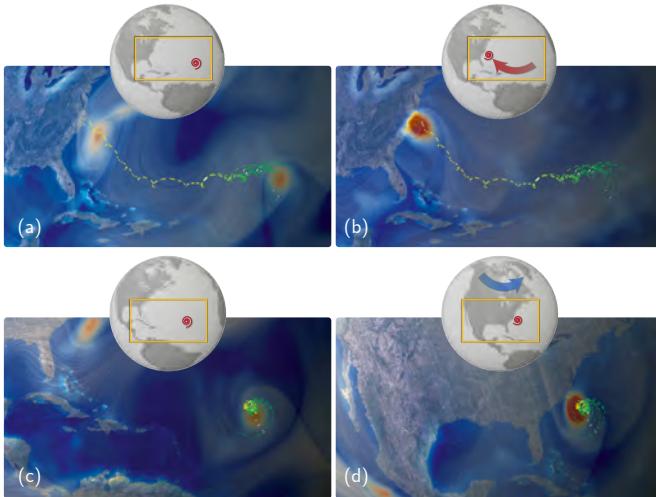


Fig. 6. Feature-centric visualization of hurricane Isabel. (a,b) Pathlines in the original flow field. (c,d) Observer-relative pathlines; the hurricane appears steady. (a,c) First time step. (b,d) Last time step. From (c) to (d), the Earth has moved underneath the steady hurricane.

With respect to this transformation, a scalar field is objective if it is unchanged; a vector field \mathbf{v} is objective if it transforms according to $\mathbf{v}^* = \mathbf{Q}(t)\mathbf{v}$; a second-order tensor field \mathbf{S} , as a linear transformation of vectors, is objective if it transforms as $\mathbf{S}^* = \mathbf{Q}(t)\mathbf{S}\mathbf{Q}(t)^T$ [65, p.42].

This entire definition depends on the domain being Euclidean: Points are position vectors; the difference between two points is a vector; all tangent spaces are copies of \mathbb{R}^3 with trivial parallel transport. This definition is therefore not valid for non-Euclidean (curved) manifolds.

5.2 Generalization of Objectivity

To generalize objectivity, we define this concept as a general notion of tensor fields being *invariant* with respect to a *continuous symmetry group*. (The term symmetry refers to a notion of being the same.) Such a symmetry group G is a *Lie group*. For example, if the group G is chosen as the *isometry group* of a (Riemannian) manifold, two tensor fields are “the same” if they are isometric. Two fields being symmetries of each other then means that there exists a group element $g \in G$, such that the transformation rules given below hold. Then, given any time-dependent observer transformation $t \mapsto g(t) \in G$, a given tensor field is *objective* if it simply follows the transformation for each $g = g(t)$.

5.2.1 Symmetry groups and group actions

Our notion of symmetry corresponds to the transformation behavior under a *group action* Φ , with a given Lie group element $g \in G$, where G is the chosen symmetry group. An action Φ , specifically a *smooth left action*, of a Lie group G on a manifold M , is a smooth map [32, p.209]

$$\begin{aligned} \Phi: G \times M &\rightarrow M, \\ (g, x) &\mapsto \Phi(g, x), \end{aligned} \tag{5}$$

such that for every $g \in G$, the map

$$\phi_g: M \rightarrow M, \text{ with } \phi_g(x) := \Phi(g, x), \text{ is a diffeomorphism.} \tag{6}$$

Here, we focus on the general use of group actions Φ in our context, and defer details to later sections. For now, it is sufficient to understand that the diffeomorphisms ϕ_g will correspond to the *flows* of specific vector fields on M . These vector fields are *generated* by the action of the *Lie algebra* \mathfrak{g} of the Lie group G on M . See App. J for details.

For example, if G is the group of all diffeomorphisms of M , these vector fields are all possible (smooth) vector fields on M . The important case for our framework is choosing the group G as the *isometry group* of M . The corresponding vector fields are then the *Killing vector fields* on M , whose flows correspond to the isometries of M . See Sec. 7.

To obtain a generalized definition of objectivity, a crucial property of the diffeomorphism ϕ_g is that it enables us to use the corresponding

differential, or *pushforward*. See Fig. 7. The pushforward is a map

$$d\phi_g: TM \rightarrow TM, \tag{7}$$

where each $(d\phi_g)_x$ at a point $x \in M$ is a *linear map*

$$\begin{aligned} (d\phi_g)_x: T_x M &\rightarrow T_{\phi_g(x)} M, \\ \mathbf{v} &\mapsto (d\phi_g)_x(\mathbf{v}). \end{aligned} \tag{8}$$

The notation $(\cdot)_x$ means that the quantity in parentheses is located at $x \in M$, and $T_x M$ denotes the tangent space at x . We can simply imagine that the diffeomorphism ϕ_g transforms curves on M , and the differential $d\phi_g$ transforms their tangent vectors accordingly. See also App. U.

In components, the linear map $(d\phi_g)_x$ at any $x \in M$ can simply be given by a matrix, e.g., a 2×2 matrix for a two-manifold M . See Fig. 7.

Euclidean space. For example, when ϕ_g is an isometry of $M = \mathbb{R}^3$, the pushforward $d\phi_g$ is a globally constant proper orthogonal (rotation) tensor \mathbf{Q} , i.e., $(d\phi_g)_x = \mathbf{Q}$, at all points $x \in M$. See O’Neill [48, p.107].

Curved spaces. In general, however, the linear map $(d\phi_g)_x$ is *different* for different points $x \in M$. In components, each $(d\phi_g)_x$ can still be given by a matrix, but it will be a different matrix for each point $x \in M$.

5.2.2 Objective scalar fields

Being objective should mean invariant under transformation, which for scalar fields is trivial. We therefore define that a scalar field $f: M \rightarrow \mathbb{R}$ on a manifold M is objective when, under any diffeomorphism ϕ_g , given by the group action Φ of a symmetry group G , it transforms as

$$f^*(\phi_g(x)) = f(x). \tag{9}$$

Abbreviated, we could write $f^* = f$, but it is crucial to note that f^* is evaluated at the point $\phi_g(x)$, whereas f is evaluated at the point x .

5.2.3 Objective vector fields

We now define that an arbitrary vector field \mathbf{v} on a manifold M is objective (with respect to a given symmetry group G), if, under the corresponding group action Φ with any $g \in G$, it transforms as

$$(\mathbf{v}^*)_{\phi_g(x)} = (d\phi_g)_x(\mathbf{v}). \tag{10}$$

We emphasize that \mathbf{v}^* is an element of the tangent space $T_{\phi_g(x)} M$, whereas \mathbf{v} is an element of $T_x M$. Likewise, it is important to note that the differential $(d\phi_g)_x$ is a linear map defined on $T_x M$. We can say

Remark. A vector field is objective, if it is simply pushed forward by any diffeomorphism ϕ_g , defined according to the group action Φ . This definition of objectivity is valid for any smooth manifold where a notion of (smooth) symmetry is defined by a (smooth) symmetry group G .

Abbreviated transformation rule. For brevity, we can define the action Φ , with $g \in G$, on any vector field \mathbf{v} on M , by the differential in Eq. 8, and abbreviate the objectivity criterion of Eq. 10 simply as

$$\mathbf{v}^* = g\mathbf{v}. \tag{11}$$

However, it is crucial that the meaning of the transformation represented by $g \in G$ in this shorthand notation is given by Eq. 10. In general, g cannot be mapped to the same globally defined matrix, corresponding to the pushforward $d\phi_g$, even though this is possible in the Euclidean case. Nevertheless, this abbreviated form makes it easy to see the analogy with the definition of Truesdell and Noll. In Euclidean space, the two are equivalent. See App. D for more details. Our definition, however, gives a well-defined notion of objectivity for arbitrary manifolds M .

5.2.4 Objective tensor fields

The definition above for objective vectors generalizes to arbitrary tensor fields \mathbf{T} , in the way that is derived in full in App. A. Here, we derive the common special case of a second-order tensor field \mathbf{S} , interpreted as a

linear map of vectors, i.e., $\mathbf{S}: TM \rightarrow TM, \mathbf{v} \mapsto \mathbf{S}(\mathbf{v})$. Such a $\binom{1}{1}$ tensor field is objective, if under the group action Φ with g it transforms as

$$(\mathbf{S}^*)_{\phi_g(x)}(\mathbf{v}^*) = (\mathrm{d}\phi_g)_x(\mathbf{S}((\mathrm{d}\phi_{g^{-1}})_{\phi_g(x)}(\mathbf{v}^*))). \quad (12)$$

The pushforward $\mathrm{d}\phi_{g^{-1}}$ of the diffeomorphism $\phi_{g^{-1}}$ maps \mathbf{v}^* from $T_{\phi_g(x)}M$ to the corresponding \mathbf{v} in T_xM according to Eq. 10 (inverted), and then applies Eq. 10 again to $\mathbf{S}(\mathbf{v})$. $\phi_{g^{-1}}$ is the inverse of ϕ_g , meaning $\phi_{g^{-1}}(\phi_g(x)) = \phi_{g^{-1}g}(x) = \phi_e(x) = x$. As in Eq. 11, we can abbreviate

$$\mathbf{S}^* = g\mathbf{S}g^{-1}. \quad (13)$$

However, it is again crucial to note that the meaning of this rule is given by Eq. 12. Fig. 7 illustrates how the pushforwards of ϕ_g and $\phi_{g^{-1}}$ map tangent vectors between the tangent spaces $T_{\phi_g(x)}M$ and T_xM .

We note that to prove objectivity of the Killing operator K (Eq. 25), which we use for computing observer fields, the general definition of objectivity from App. A is required, since $K\mathbf{u}$ is a $\binom{0}{2}$ tensor (App. F.1).

5.3 Objectivity for Intrinsic Isometries

To realize the notion of observer transformations as isometric transformations of the underlying space in the general mathematical framework given above, we now consider the group $G := \text{Isom}(M)$, i.e., the *isometry group* of the manifold M . The group action Φ is an isometry, with respect to a given metric \mathbf{g} , i.e., a smoothly varying inner product $\langle \cdot, \cdot \rangle$, on M , if for all $g \in G$, and for any two vector fields \mathbf{v}, \mathbf{w} on M , we have

$$\langle (\mathrm{d}\phi_g)_x(\mathbf{v}), (\mathrm{d}\phi_g)_x(\mathbf{w}) \rangle_{\phi_g(x)} = \langle \mathbf{v}, \mathbf{w} \rangle_x. \quad (14)$$

As a direct consequence of Eq. 14 holding for the map ϕ_g , we get that for any vector field \mathbf{v} that is objective according to Eq. 10, we have

$$\|\mathbf{v}^*\|_{\phi_g(x)} = \|\mathbf{v}\|_x. \quad (15)$$

Infinitesimal isometries and Killing vector fields. Computationally, instead of working with isometries directly, we work with their derivatives, which are *infinitesimal isometries*, corresponding to the Killing fields on M . See Sec. 7. To describe Killing fields efficiently, we need the *intrinsic* velocity gradient tensor $\nabla\mathbf{v}$ of a velocity field \mathbf{v} on M .

6 GENERAL VELOCITY GRADIENT TENSORS

For any smooth vector field \mathbf{v} given on a manifold M , the *velocity gradient tensor* $\nabla\mathbf{v}$ is a second-order tensor field on M that allows computing directional derivatives of the vector field \mathbf{v} in any direction.

In Cartesian coordinates, $\nabla\mathbf{v}$ can simply be defined by partial derivatives $\partial_j v^i$. However, this definition cannot be used for intrinsic computations on curved manifolds, because $\partial_j v^i$ is in general *not a tensor*. For a curved (or flat) manifold M , the tensor $\nabla\mathbf{v}$ can, however, always be computed intrinsically as the *covariant derivative* of \mathbf{v} [21, p.241].

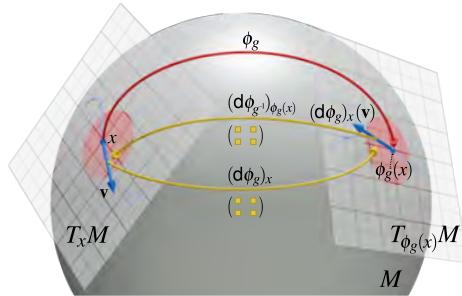


Fig. 7. **The pushforward** of a diffeomorphism ϕ_g is a linear map between the tangent spaces T_xM and $T_{\phi_g(x)}M$. That is, the pushforward $(\mathrm{d}\phi_g)_x$ maps a tangent vector $\mathbf{v} \in T_xM$ to the vector $(\mathrm{d}\phi_g)_x(\mathbf{v}) \in T_{\phi_g(x)}M$. The pushforward $(\mathrm{d}\phi_{g^{-1}})_{\phi_g(x)}$ maps a tangent vector in the opposite direction. In components, for a two-manifold M , each pushforward is a 2×2 matrix.

6.1 Intrinsic Covariant Derivatives

The covariant derivative $\nabla\mathbf{v}$ of a vector field \mathbf{v} on a manifold M is a second-order tensor field on M , which, at any $x \in M$, maps an arbitrary vector $\mathbf{x} \in T_xM$ to the derivative of the field \mathbf{v} in the direction \mathbf{x} , i.e.,

$$\begin{aligned} (\nabla\mathbf{v})_x : T_xM &\rightarrow T_xM, \\ \mathbf{x} \mapsto \nabla_{\mathbf{x}}\mathbf{v} &= (\nabla\mathbf{v})_x(\mathbf{x}). \end{aligned} \quad (16)$$

This notation means that the covariant derivative of a vector field \mathbf{v} , evaluated in the direction \mathbf{x} , is the vector $\nabla_{\mathbf{x}}\mathbf{v}$, which we get by computing $\nabla\mathbf{v}(\mathbf{x})$. We can refer $\nabla\mathbf{v}$ to a basis $\{\mathbf{e}_i \otimes \omega^j\}$, with \otimes the tensor product, and write the components as $n \times n$ matrix $\nabla_j v^i$, with row and column indices i, j , respectively, and n the dimension of M , i.e., for a curved surface $n = 2$. However, unless the manifold M is flat *and* Cartesian or affine coordinates are used, $\nabla_j v^i \neq \partial_j v^i$. The difference is a *connection* on M [21, p.242], corresponding to a notion of *parallel transport* [18, p.52] of vectors. The connection can be given in components via *Christoffel symbols* Γ_{jk}^i , and we get for $\nabla\mathbf{v}$ in components,

$$\nabla_j v^i := \partial_j v^i + \Gamma_{jk}^i v^k. \quad (17)$$

In flat space with Cartesian or affine coordinates, all Christoffel symbols vanish [23, p.72] ($\Gamma_{jk}^i \equiv 0$), but *only* in this special case is $\nabla_j v^i = \partial_j v^i$. The Γ_{jk}^i determine the *covariant derivatives of the basis vector fields* $\{\mathbf{e}_i\}$, at each point $x \in M$, referred back to the same basis $\{\mathbf{e}_i\}$ at x , i.e.,

$$\nabla_{\mathbf{e}_i} \mathbf{e}_k = \Gamma_{jk}^i \mathbf{e}_i, \quad \text{and therefore} \quad \Gamma_{jk}^i = \omega^i(\nabla_{\mathbf{e}_j} \mathbf{e}_k). \quad (18)$$

We note that although Γ_{jk}^i comprises n^3 components, it is *not* a (third-order) tensor [23, p.82]. In general, a connection needs to be *chosen* [18, p.50]. We employ the *Levi-Civita connection*,¹ which can be derived intrinsically from the metric of M [21, p.229]. See Eq. O.10, App. O. Even more easily, if we know an *immersion* of M into an ambient space \mathbb{R}^m , as for a curved surface immersed in \mathbb{R}^3 , we can obtain the Γ_{jk}^i as

$$\Gamma_{jk}^i = \tilde{\omega}^i(\partial_j \tilde{\mathbf{e}}_k). \quad (19)$$

The tildes denote ambient space (co)vectors, e.g., for M with $n = 2$ and ambient $m = 3$, $\{\partial_j \tilde{\mathbf{e}}_k\}$ comprises four 3D vectors, and $\{\tilde{\omega}^i\}$ comprises two 3D covectors. It is crucial that the $\{\tilde{\omega}^i\}$ are chosen such that they correspond to *orthogonal projection* from \mathbb{R}^m onto the tangent plane of the surface. See App. Q for a complete calculation for a simple atlas of charts for the sphere. The Levi-Civita connection given by this choice of Γ_{jk}^i corresponds to the metric on M that is *induced* by the immersion of M into the ambient space \mathbb{R}^m [18, p.39]. Crucially, however, the resulting Christoffel symbols are nevertheless *intrinsic*.

7 ISOMETRIES AND KILLING FIELDS IN CURVED SPACES

Our framework requires an efficient mathematical description of the intrinsic isometries of the manifold M . A powerful way to do this is to work with the derivatives of diffeomorphisms that are isometries, i.e., to consider *infinitesimal isometries*. These correspond to specific vector fields on the manifold M , which are called *Killing vector fields*.

A vector field \mathbf{u} is a Killing vector field on M , if the flow ϕ_t (App. T) that it generates preserves the metric \mathbf{g} on M . The most general way of stating this mathematically is to require that the *Lie derivative* (App. G) of the metric \mathbf{g} , with respect to \mathbf{u} , must vanish everywhere on M . I.e.,

$$\mathcal{L}_{\mathbf{u}} \mathbf{g} = 0. \quad (20)$$

This means that for all $x \in M$, we have $(\mathcal{L}_{\mathbf{u}} \mathbf{g})_x = 0$. See [45, p.250].

The set of all possible Killing vector fields on a given (Riemannian) manifold M constitutes a *Lie algebra*, which is, in particular, a vector space. This enables us to consider the *dimensionality* of the set of

¹This is the *unique* metric-compatible connection, corresponding to a given (Riemannian) metric \mathbf{g} on the manifold M , that has zero torsion [21, p.242].

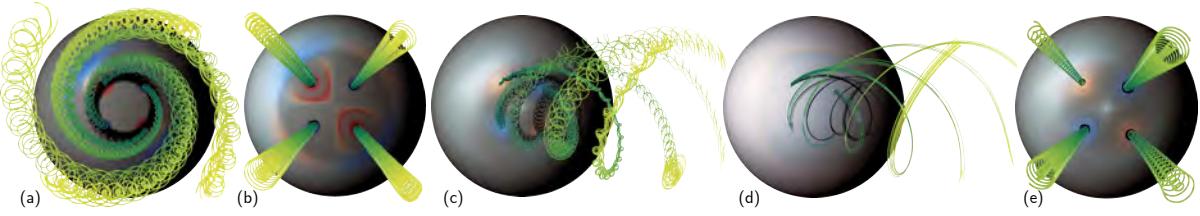


Fig. 8. Rotating four centers. (a) Four vortices rotating around each other. Path lines seeded close to the centers are shown in space-time, as in Fig. 3. (b) The flow observed by a steady observer field rotating clockwise with the vortices. The computed observer field corresponds to a rotation around the center point between the four vortices. (c) The rotating four centers as in (a), rotated around the vertical axis of the sphere for $\frac{\pi}{4}$ rotations. (d) The observer field is a non-trivial time-dependent superposition of two rotations. (e) Observed field of (c), relative to the observer field (d).

Killing fields on M , and to write any Killing field as a linear combination of *basis Killing fields*. Fig. 5 shows an example on the sphere. Apps. L and M describe the isometries of the two-sphere in detail, with full calculations. App. K gives more details on Killing vector fields on curved surfaces in general. However, for computing observer fields via optimization (Sec. 9), as the most important part of our framework, we directly evaluate *Killing's equation* on the unknown vector field \mathbf{u} .

7.1 Killing's Equation

A vector field $\mathbf{u} = u^i \mathbf{e}_i$ is a Killing field, if an equation known as Killing's equation holds at all $x \in M$ [21, p.529]. However, in its general form this equation is given for *covectors*, with components u_i , and not for vectors, with components u^i . The u_i can be computed as $u_i = g_{ij}u^j$ [23, p.88], using a metric \mathbf{g} , referred to components g_{ij} .

Killing's equation is then given as

$$\nabla_j u_i + \nabla_i u_j = 0. \quad (21)$$

This is the same as $\nabla \mathbf{u}$ being *anti-symmetric*, which is the same as $\langle \nabla \mathbf{u}(\mathbf{x}), \mathbf{x} \rangle = 0$ for all \mathbf{x} . However, in general it is not correct to simply check whether a component matrix of $\nabla \mathbf{u}$ is anti-symmetric, because $\nabla \mathbf{u}$ is a tensor of mixed type (contravariant and covariant indices). Therefore, Eq. 21 is given using the covariant derivative of a *covector field* [23, p.108]. In components, referred to a basis $\{\omega^i \otimes \omega^j\}$, this is

$$\nabla_j u_i := \partial_j u_i - \Gamma_{ij}^k u_k. \quad (22)$$

We can introduce the vector components u^i directly, and write Eq. 21 as

$$\nabla_j(g_{ik}u^k) + \nabla_i(g_{jk}u^k) = 0. \quad (23)$$

Applying the product rule for covariant derivatives [23, p.119], we get $\nabla_j(g_{ik}u^k) = (\nabla_j g_{ik})u^k + g_{ik}\nabla_j u^k = g_{ik}\nabla_j u^k$, because the covariant derivative of the metric vanishes.² Therefore, Eq. 21 is equivalent to

$$g_{ik}\nabla_j u^k + g_{jk}\nabla_i u^k = 0. \quad (24)$$

In matrix notation, we can therefore rewrite Killing's equation as the vanishing of the differential “Killing operator” K applied to the field \mathbf{u} ,

$$K\mathbf{u} := \mathbf{g}\nabla\mathbf{u} + (\mathbf{g}\nabla\mathbf{u})^T = 0. \quad (25)$$

The tensor $K\mathbf{u}$ is still a $\binom{0}{2}$ tensor, just like $(\nabla_j u_i + \nabla_i u_j)$. However, Eq. 25 implies that now we can simply check any matrix representation of $\mathbf{g}\nabla\mathbf{u}$ for anti-symmetry, to determine whether \mathbf{u} is a Killing field.

7.2 Killing Energy

If the expression $K\mathbf{u}$ is not exactly zero, in order to *quantify* how close to being Killing an arbitrary vector field \mathbf{u} is, we use the *tensor norm* of the tensor $K\mathbf{u}$ to define a scalar *Killing energy* (see also [8, Def. 5]). As for vectors, the tensor norm of a higher-order tensor \mathbf{T} is defined as

$$\|\mathbf{T}\|_g^2 := \langle \mathbf{T}, \mathbf{T} \rangle_g. \quad (26)$$

However, the *tensor inner product* required here, between two covariant second-order tensors \mathbf{T}, \mathbf{S} , is given by the double contraction [44, p.82]

$$\langle \mathbf{T}, \mathbf{S} \rangle_g := g^{ik}g^{jl}T_{kl}S_{ij} = T^{ij}S_{ij}. \quad (27)$$

²This $(\nabla \mathbf{g} = 0)$ is the *definition* of a metric connection [21, p.242].

This is similar to the Frobenius inner product $\text{tr}(\mathbf{T}^T \mathbf{S})$ of matrices, which determines the Frobenius norm $\text{tr}(\mathbf{T}^T \mathbf{T})^{1/2}$. The crucial difference here is that because pairs of covariant and contravariant indices are contracted, Eq. 27 is an invariant tensor expression, giving the same result in all coordinate systems. $\langle \mathbf{T}, \mathbf{T} \rangle_g = T^{ij}T_{ij}$ thus is the square of the norm. In matrix notation, we can write the inner product above as

$$\langle \mathbf{T}, \mathbf{S} \rangle_g = \text{tr}(\mathbf{g}^{-1} \mathbf{T}^T \mathbf{g}^{-1} \mathbf{S}). \quad (28)$$

We define the Killing energy $E\mathbf{u}$, of the field \mathbf{u} , at a point $x \in M$, as

$$E\mathbf{u} := \|\mathbf{K}\mathbf{u}\|_g^2 = \langle \mathbf{K}\mathbf{u}, \mathbf{K}\mathbf{u} \rangle_g. \quad (29)$$

Sec. 9.1 defines the Killing energy of M as the integral of $E\mathbf{u}$ over M . Eq. 29 is then the *density* of the Killing energy of M per unit area.

8 OBSERVED TIME DERIVATIVES IN CURVED SPACES

The *observed time derivative* defined in Eq. 1 measures the differential change of a tensor field with respect to the motion of a given observer field \mathbf{u} . We note that $\mathcal{D}/\mathcal{D}t$ is, in fact, the *time-dependent Lie derivative* [44, p.95] with respect to the flow of the field \mathbf{u} . The crucial difference here is the semantic meaning of an observer field \mathbf{u} , as an intrinsic part of our operator. Our most important use of $\mathcal{D}/\mathcal{D}t$ is applying it to the *relative velocity field* $(\mathbf{v} - \mathbf{u})$, which we minimize to compute the unknown observer field \mathbf{u} (Sec. 9). We get ($\mathcal{L}_{\mathbf{u}} \mathbf{u} = 0$),

$$\begin{aligned} \frac{\mathcal{D}}{\mathcal{D}t}(\mathbf{v} - \mathbf{u}) &= \frac{\partial \mathbf{v}}{\partial t} - \frac{\partial \mathbf{u}}{\partial t} + \mathcal{L}_{\mathbf{u}} \mathbf{v}, \\ &= \frac{\partial \mathbf{v}}{\partial t} - \frac{\partial \mathbf{u}}{\partial t} + \nabla \mathbf{v}(\mathbf{u}) - \nabla \mathbf{u}(\mathbf{v}). \end{aligned} \quad (30)$$

Lie derivatives in curved spaces. A basic property of Lie derivatives is that they are independent of the metric \mathbf{g} defined on the manifold M , and likewise independent of the connection on M [44, p.96]. That is,

$$\begin{aligned} \mathcal{L}_{\mathbf{u}} \mathbf{v} &= \nabla \mathbf{v}(\mathbf{u}) - \nabla \mathbf{u}(\mathbf{v}), \\ &= (\partial_j v^i u^j - \partial_j u^i v^j) \mathbf{e}_i. \end{aligned} \quad (31)$$

The expansion in components in the second row above comprises only partial derivatives. The full derivation is given in App. H. Thus, the observed time derivative can be derived solely from partial derivatives, even when the manifold M is curved, including the $\mathcal{L}_{\mathbf{u}}$ term in Eq. 30.

9 COMPUTING OBSERVER FIELDS

We compute a global observer field \mathbf{u} on a two-manifold M with multiple charts using a *single* linear least-squares optimization step. We enable the solution of a single sparse linear system by baking the transition maps (Jacobian matrices) between charts into the system matrix.

9.1 Objective Function

Our optimization is conceptually very similar to the approach of Hadwiger et al. [29] for Euclidean space. However, all differential operators have to be substituted by their generalized version for arbitrary manifolds, i.e., covariant derivatives, Lie derivatives, and tensor norms.

We formulate a global optimization problem to find an observer field \mathbf{u} , from the space of all possible fields in some function space \mathcal{V} of vector fields (e.g., C^n fields) on the manifold M , as the minimizer of

$$\min_{\mathbf{u} \in \mathcal{V}} \frac{1}{2} \int_{t_{\min}}^{t_{\max}} \int_M (E + \lambda D + \mu R)(\mathbf{u}, x, t) dA dt. \quad (32)$$

Here, dA is an infinitesimal area element of M at a point $x \in M$, and the constants $\lambda, \mu \in \mathbb{R}$ are relative weights. The individual terms are,

$$E(\mathbf{u}, x, t) := \|K\mathbf{u}(x, t)\|_{\mathbf{g}}^2, \quad (33)$$

$$D(\mathbf{u}, x, t) := \left\| \frac{\mathcal{D}}{\mathcal{D}t} (\mathbf{v} - \mathbf{u})(x, t) \right\|_{\mathbf{g}}^2, \quad (34)$$

$$R(\mathbf{u}, x, t) := \|(\mathbf{v} - \mathbf{u})(x, t)\|_{\mathbf{g}}^2. \quad (35)$$

The integral of E over the domain gives the *Killing energy* $\int_M E\mathbf{u}$ of the observer field \mathbf{u} on M . $\|\cdot\|_{\mathbf{g}}$ denotes the tensor and vector norms, respectively, depending on the argument, with respect to the metric \mathbf{g} .

Discretization. To solve Eq. 32, we discretize the manifold M with a triangle mesh. Vector components are stored at the vertices, and we label the corresponding points on the manifold M with x_1, x_2, \dots, x_N .

The Killing energy term E (Eq. 33) is computed by discretizing the Killing operator K (Eq. 25), to compute the covariant derivative terms of $\nabla\mathbf{u}$ in Eq. 25. The weights for all differential operators to evaluate $K\mathbf{u}$ are stored in a matrix \mathbf{K} . We compute partial derivatives via triangle vertex 1-ring neighborhood stencils, computing weighted averages of function values at mesh vertices. The weights are independent of the actual function values. They can therefore be pre-computed for a fixed mesh geometry, and be used for all functions. See App. R for details.

The observed time derivative term D (Eq. 34) is computed from partial derivatives alone (Eqs. 30 and 31). We compute time derivatives via central differences (in t) on each vertex. The weights for all differential operators to compute D are stored in a matrix \mathbf{D} . The regularization term R (Eq. 35) corresponds to a matrix \mathbf{N} , which is simply an identity matrix multiplied by -1 , to compute $-\mathbf{u}$. All terms involving the (known) input field \mathbf{v} are pre-computed and stored in the corresponding locations of the matrices \mathbf{K}, \mathbf{D} , and the right-hand side \mathbf{b} (see below).

9.2 Global Least-Squares Solution

To solve Eq. 32, we are looking for the minimizer of the least-squares problem $\min_{\mathbf{u}} \frac{1}{2} \|\mathbf{Au} - \mathbf{b}\|_{\mathbf{g}}^2$, with matrix \mathbf{A} and vector \mathbf{b} as given below. The least-squares solution, weighted with the metric \mathbf{g} , is the solution of the normal equations of the weighted least-squares problem given by the square, positive-definite system $\mathbf{A}^T \mathbf{W} \mathbf{A} \mathbf{u} = \mathbf{A}^T \mathbf{W} \mathbf{b}$, with

$$\mathbf{A} := \begin{pmatrix} \mathbf{K} \\ \mathbf{D} \\ \mathbf{N} \end{pmatrix}, \quad \mathbf{W} := \mathbf{M} \cdot \begin{pmatrix} \bar{\mathbf{G}} & 0 & 0 \\ 0 & \lambda \cdot \mathbf{G} & 0 \\ 0 & 0 & \mu \cdot \mathbf{G} \end{pmatrix}, \quad \mathbf{b} := - \begin{pmatrix} 0 \\ \frac{\partial \mathbf{v}}{\partial t} \\ \mathbf{v} \end{pmatrix}. \quad (36)$$

The matrix \mathbf{M} is diagonal, with weights giving the area on M assigned to each vertex element x_i , discretizing integration over area elements dA . The other blocks of the matrix \mathbf{W} encode \mathbf{g} on M , given by the blocks

$$\bar{\mathbf{G}} := \begin{pmatrix} \bar{\mathbf{g}}_{x_1}^{-1} & 0 & \dots & 0 \\ 0 & \bar{\mathbf{g}}_{x_2}^{-1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \bar{\mathbf{g}}_{x_N}^{-1} \end{pmatrix}, \quad \mathbf{G} := \begin{pmatrix} \mathbf{g}_{x_1} & 0 & \dots & 0 \\ 0 & \mathbf{g}_{x_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{g}_{x_N} \end{pmatrix}. \quad (37)$$

Here, \mathbf{g}_{x_i} simply refers to the metric tensor in components g_{ij} , at the vertex $x_i \in M$. For efficient computation of the tensor norm $\|\mathbf{T}\|_{\mathbf{g}}$, we use the following approach to convert the tensor inner product in Eq. 27 to a simple contraction of two first-order tensors. We introduce new \bar{n} -dimensional first-order tensors $\bar{\mathbf{T}}, \bar{\mathbf{S}}$, and a symmetric \bar{n} -dimensional second-order tensor $\bar{\mathbf{g}}^{-1}$, with $\bar{n} := n^2$, i.e., $\bar{n} = 4$ for $n = 2$. We denote the components of $\bar{\mathbf{g}}^{-1}$ by \bar{g}^{st} , and compute them from g^{ij} as

$$\bar{g}^{st} := g^{ik} g^{jl}, \quad s, t \in [1, \bar{n}], \quad (i, j) = \sigma^{-1}(s), (k, l) = \sigma^{-1}(t). \quad (38)$$

The indices i, j, k, l are given by the inverse of the indexing function

$$\sigma(i, j) := n \cdot (i - 1) + j, \quad \text{with } i, j \in [1, n]. \quad (39)$$

We can use this to rearrange the tensor inner product, to obtain

$$\begin{aligned} \bar{T}_t &:= T_{kl}, \quad \text{with } t = \sigma(k, l), \\ \bar{S}_s &:= S_{ij}, \quad \text{with } s = \sigma(i, j), \\ \bar{T}^s &= \bar{g}^{st} \bar{T}_t, \quad \text{and therefore,} \\ \bar{T}^s \bar{S}_s &= \bar{g}^{st} \bar{T}_t \bar{S}_s = g^{ik} g^{jl} T_{kl} S_{ij} = T^{ij} S_{ij}. \end{aligned} \quad (40)$$

Thus, the tensor inner product $\langle \mathbf{T}, \mathbf{S} \rangle_{\mathbf{g}}$ (Eq. 27) can simply be computed as the contraction $\bar{T}^s \bar{S}_s$. Using this approach, the tensor norm of the Killing energy in the least-squares problem above is computed by simply inserting the components \bar{g}^{st} of $\bar{\mathbf{g}}^{-1}$ into the matrix $\bar{\mathbf{G}}$ above.

9.3 Multi-Chart Optimization

At each vertex $x_i \in M$, there is only one vector for the field \mathbf{u} . We store vector components with respect to one selected chart for each x_i . However, finite differences to estimate derivatives (App. R) have to be computed in the same chart, even when adjacent vertices are in different charts. We enable this by “baking” all required chart transition maps, i.e., the 2×2 Jacobian matrices transforming from the coordinate basis of one chart to that of another chart, into the matrices \mathbf{K} and \mathbf{D} . The elements of these matrices are inserted such that all finite differences are computed using vector components referencing the same chart. More details for implementing this approach are explained in App. S.

9.4 Objectivity of Intrinsic Observer Fields

Remark. *The observer field \mathbf{u} that is obtained as the minimizer of Eq. 32 is objective with respect to the isometry group $G = \text{Isom}(M)$ of any given manifold M . We give full proofs for this claim in App. F.*

Except point (4) below, the reasons for this fact are intuitively simple and can be summarized as follows: (1) The norms used in Eq. 32 are invariant under isometries, because by definition (Eq. 14), isometries do not change them. (2) The Killing energy $\int_M E\mathbf{u}$ measures how much the field \mathbf{u} differs from an infinitesimal isometry. Since observer transformations are exact isometries of M , they do not change the Killing energy of \mathbf{u} on M . (3) A time-dependent observer transformation $t \mapsto g(t)$, which we define to be an isometry of M , changes velocity fields on M by point-wise addition of the Killing field corresponding to the derivative $X(t)$ of the isometry. Thus, the Killing field that is added due to the observer transformation cancels out in the relative velocity field $(\mathbf{v} - \mathbf{u})$. (4) The objectivity of the observed time derivative is far from obvious. However, Marsden and Hughes [44, p.101] prove that the time-dependent Lie derivative of an objective tensor field with respect to the flow of a (non-objective) vector field is objective. We also give a full proof in App. F. In summary, the norm of all terms in Eq. 32 does not change under the isometry ϕ_g of any observer transformation $g(t)$. Because the minimizer of Eq. 32 is unique (as in [29]), it is thus objective with respect to the group action Φ of the isometry group of M .

10 RESULTS

We have implemented a general framework for flow visualization on curved surfaces in C++ and OpenGL for rendering. For the sphere, we use six very simple charts, as described in detail in App. Q. We emphasize that the only 3D computations that we use are for initially obtaining the metric of the sphere as a 2D tensor field in each of the charts, and for visualization purposes in OpenGL. All our vector fields are given via 2D components, and all computations required for solving Eq. 32 are fully intrinsic in 2D. We solve the least-squares system given in Sec. 9.2 by using the conjugate gradient solvers provided by the C++ Eigen library, obtaining the minimizer \mathbf{u} in multiple iterations. We have also tried direct methods using matrix factorization of the symmetric positive definite matrix $\mathbf{A}^T \mathbf{W} \mathbf{A}$. Due to the size of our grids, conjugate gradients obtain the minimum faster.

Data sets. Table 1 gives an overview of the data sets that we have used in this paper and the video. We ran experiments with synthetic data

Table 1. Data sets with grid/mesh resolution.

Data set	# Vert.	# Tri.	Time	Time steps
SYNTHETIC VORTEX	10,242	20,480	$0\dots 2\pi$	100
SYNTHETIC FOUR CENTER	10,242	20,480	$0\dots 2\pi$	100
VORTEX STREET	40,962	81,920	$0\dots \frac{\pi}{2}$	300
EARTH FLOW	163,842	327,680	8 days	32
EARTH FLOW (SUBDOMAIN)	32,400	64,796	8 days	32
EARTH FLOW (ADAPT. RES.)	62,412	124,820	8 days	32

sets and with simulated data on the sphere. The optimizations are global, however in the case of the earth flow data set we experimented with two approaches for region of interest optimization to reduce computation time. Table 2 gives computation times for the conjugate gradient method to optimize observer fields \mathbf{u} . All computations were run on two Intel Xeon E5-2600 processors with 16 cores running at $2.3GHz$.

Observer-aware visualization. We have adapted the visualization methods described by Hadwiger et al. [29] from Euclidean space to curved spaces. Since we use coordinate charts for the optimization, we can also use them as texture parametrization for computing LIC in texture space on the GPU. Corresponding to the observer field \mathbf{u} , we compute LIC images of *observed stream lines* [29]. Likewise we also compute the geometry of integral curves for observed stream lines and *observed path lines*. To visualize the time component of path lines, we increase their radial distance from the sphere according to the time parameter, scaled by a user-specified factor. This is helpful to see the time evolution of path lines. We draw shadows of path lines directly on the sphere to ensure that their position in space can be correctly interpreted. Further, we visualize time-dependent observer-relative scalar fields using simple color coding, as well as steady observer-relative scalar fields such as the earth texture. We can also dynamically switch the frame of reference for any flow field and directly visualize the corresponding observed fields. Figs. 1, 6, and 9 show observer-aware scalar field visualizations (planet textures, and vorticity), as well as observed LIC. Figs. 3, 6, and 8 show observer-aware path lines.

10.1 Synthetic Data Sets

Synthetic Vortex. We transform a steady field of a vortex into time-dependent fields by rotating it around the vertical axis of the sphere. The rotations with different angular velocities correspond to Killing fields of different speeds (Fig. 3).

From these transformed fields, we compute one observer field each. In all cases the steady Killing fields (of different speeds) that are the minimizers of Eq. 32 are computed, which allow us to recover the original steady vortex in Fig. 3 (d). The different unsteady fields become steady again, as observed by their respective observer fields.

Synthetic Four Centers. Fig. 8 shows the superposition of multiple steady observer fields that together form a time-dependent Killing field. Fig. 8 (a) shows four vortices that rotate around each other. Fig. 8 (b) shows the steady flow field of four vortices. We recover this field relative to a steady observer field that rotates clockwise. Another rotation around the vertical axis is superimposed on the field (Fig. 8

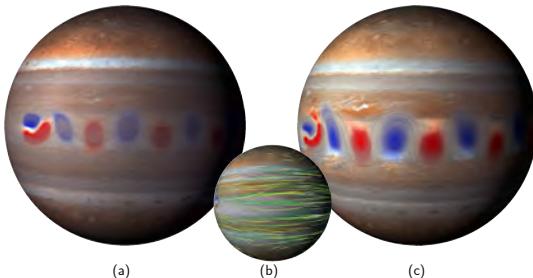


Fig. 9. **Planet-scale observed flow patterns.** Vorticity is shown color-coded on a red-blue CCW/CW scale. LIC is used to visualize instantaneous velocities. (a) A vortex street at the equator. LIC cannot show the vortices. (b) The globally-optimized observer field that transforms the flow field (a) into a near-steady flow field (c), which shows the vortices in place while the planet is moving underneath instead. Note how LIC of the instantaneous velocity field now clearly shows the vortices.

Table 2. Computation times for the observer field \mathbf{u} .

Data set	λ	μ	CG Iter.	Comp.-time
SYNTHETIC VORTEX	0.1	0	1,000	6s
SYNTHETIC FOUR CENTER	1	0	10,000	78s
VORTEX STREET	10^3	0	5,000	12min
EARTH FLOW (SUBDOMAIN)	10^{13}	0	10,000	3min
EARTH FLOW (ADAPT. RES.)	10^{13}	0	10,000	6min

(a)) resulting in the flow field shown in Fig. 8 (c). Fig. 8 (d) shows the computed observer field for the flow-field shown in Fig. 8 (c). This observer is a superposition of two steady rotational observers which results in an unsteady field. The shown observer field (Fig. 8 (d)) is the minimizer of Eq. 32. It is used to recover the steady field (Fig. 8 (e)).

10.2 Global Planet-Scale Optimization

Hurricane Isabel. For planet-scale experiments we use open source flow data from the European Copernicus project. We use global wind data to show that large phenomena like hurricane Isabel are automatically tracked by the solution of our optimization problem, without performing explicit feature tracking. Figs. 1 and 6 show the input flow data and the observed flow fields. We are able to compute an observed flow field that is nearly steady. To get an undistorted visualization that is centered on the hurricane, we can pick one time-dependent Killing field as the observer that travels with the hurricane. The resulting visualization applies the inverse observer motion to the earth texture.

Vortex Street. We use a 2D von Kármán vortex street for testing, mapped to a sphere directly via a latitude-longitude grid. See Fig. 9.

10.3 Region of Interest Optimization

Our approach optimizes for the best observer field $\mathbf{u}(x, t)$ with a global optimization. However, often an optimization of the whole domain at the same resolution level is not necessary, since the user might only be interested in a smaller subdomain. We have tested two methods for these cases. We either select a subdomain and run the optimization only in this subdomain, or adaptively remesh to have different mesh resolutions at different parts of the sphere and run the optimization over the whole domain. The computation times can be greatly reduced either way (Table 2).

11 DISCUSSION AND CONCLUSIONS

We have presented the first objective, optimization-based framework for computing time-dependent reference frames in curved spaces with the goal of turning unsteady input flow fields into (nearly) steady fields relative to a global observer field. A major motivation for this approach is to enable *feature-relative* visualization and analysis of the time evolution of features, such as scalar and vector field attributes of hurricane simulations, “in place.” A powerful example of this is shown in Fig. 1 (top left), where the path of hurricane Isabel is simply a path line of the automatically computed observer field \mathbf{u} . We have seeded this path line manually, but this could also be done algorithmically. Moreover, path lines of \mathbf{u} anywhere inside the core region of the hurricane follow the general time-dependent diffeomorphism that constitutes the warping domain of the hurricane. We believe that our mathematical framework has great potential in applications such as studying features of interest in climatology and oceanography in a reference frame-independent way. Our method is objective, meaning that it is invariant with respect to any isometric observer transformation of the input flow field. This enables it to automatically follow the motion of features of interest, and makes it independent of super-imposed motions, including those of moving measurement devices capturing real-world geophysical flow phenomena. Finally, we believe that the fully intrinsic approach that we have taken, and the methodology from Riemannian geometry that we have used in our framework, could provide significant advantages and insight to other areas and applications of flow visualization as well.

ACKNOWLEDGMENTS

We thank Anna Frühstück for the illustrations and for help with the figures and the video. This work was supported by King Abdullah University of Science and Technology (KAUST), and the KAUST Office of Sponsored Research (OSR) award OSR-2015-CCF-2533-01. This research used resources of the Core Labs of KAUST.

REFERENCES

- [1] M. Alexa, D. Cohen-Or, and D. Levin. As-rigid-as-possible shape interpolation. In *Proceedings of SIGGRAPH 2000*, pp. 157–164, 2000.
- [2] R. Aris. *Vectors, Tensors and the Basic Equations of Fluid Mechanics*. Dover Publications, Inc., 1990.
- [3] V. I. Arnold. *Mathematical Methods of Classical Mechanics*. Springer-Verlag, 2nd ed., 1989.
- [4] O. Azencot, M. Ben-Chen, F. Chazal, and M. Ovsjanikov. An operator approach to tangent vector field processing. *Computer Graphics Forum*, 32(5):73–82, 2013.
- [5] O. Azencot, M. Ovsjanikov, F. Chazal, and M. Ben-Chen. Discrete derivatives of vector fields on surfaces—an operator approach. *ACM Transactions on Graphics*, 34(3):Article No. 29, 2015.
- [6] S. Bachthaler, M. Strengert, D. Weiskopf, and T. Ertl. Parallel texture-based vector field visualization on curved surfaces using GPU cluster computers. In *Proceedings of Eurographics Symposium on Parallel Graphics and Visualization (EGPGV)*, pp. 75–82, 2006.
- [7] H. Battke, D. Stalling, and H.-C. Hege. Fast line integral convolution for arbitrary surfaces in 3d. In *Visualization and Mathematics: Experiments, Simulations, and Environments*, pp. 181–195. Springer, 1997.
- [8] M. Ben-Chen, A. Butscher, J. Solomon, and L. Guibas. On discrete Killing vector fields and patterns on surfaces. In *Proceedings of Eurographics Symposium on Geometry Processing*, pp. 1701–1711, 2010.
- [9] H. Bhatia, G. Norgard, V. Pascucci, and P.-T. Bremer. The Helmholtz-Hodge decomposition—a survey. *IEEE Transactions on Visualization and Computer Graphics*, 19(8):1386–1404, 2013.
- [10] H. Bhatia, V. Pascucci, and P.-T. Bremer. The natural Helmholtz-Hodge decomposition for open-boundary flow analysis. *IEEE Transactions on Visualization and Computer Graphics*, 20(11):1566–1578, 2014.
- [11] H. Bhatia, V. Pascucci, R. M. Kirby, and P.-T. Bremer. Extracting features from time-dependent vector fields using internal reference frames. *Computer Graphics Forum*, 33(3):21–30, 2014.
- [12] R. Bujack, M. Hlawitschka, and K. I. Joy. Topology-inspired Galilean invariant vector field analysis. In *Proceedings of IEEE Pacific Visualization 2016*, pp. 72–79, 2016.
- [13] B. Cabral and L. C. Leedom. Imaging vector fields using line integral convolution. In *Proceedings of SIGGRAPH '93*, pp. 263–270, 1993.
- [14] G. Chen, V. Kwatra, L.-Y. Wei, C. D. Hansen, and E. Zhang. Design of 2d time-varying vector fields. *IEEE Transactions on Visualization and Computer Graphics*, 18(10):1717–1730, 2012.
- [15] G. Chen, K. Mischaikow, R. S. Laramee, P. Pilarczyk, and E. Zhang. Vector field editing and periodic orbit extraction using morse decomposition. *IEEE Transactions on Visualization and Computer Graphics*, 13(4):769–785, 2007.
- [16] F. de Goes, M. Desbrun, and Y. Tong. Vector Field Processing on Triangle Meshes. In *SIGGRAPH Asia 2015 Courses*, pp. 17:1–17:48, 2015.
- [17] F. de Goes, B. Liu, M. Budninskiy, Y. Tong, and M. Desbrun. Discrete 2-tensor fields on triangulations. *Computer Graphics Forum*, 33(5):13–24, 2014.
- [18] M. P. do Carmo. *Riemannian Geometry*. Birkhäuser, 1992.
- [19] M. Edmunds, R. S. Laramee, G. Chen, N. Max, E. Zhang, and C. Ware. Surface-based flow visualization. *Computers & Graphics*, 36(8):974–990, 2012.
- [20] L. K. Forssell and S. D. Cohen. Using line integral convolution for flow visualization: curvilinear grids, variable-speed animation, and unsteady flows. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):133–141, 1995.
- [21] T. Frankel. *The Geometry of Physics: An Introduction*. Cambridge University Press, 3rd ed., 2011.
- [22] A. Frishman and C. Herbert. Turbulence statistics in a two-dimensional vortex condensate. *Phys. Rev. Lett.*, 120(20):204505, 2018.
- [23] P. Grinfeld. *Introduction to Tensor Analysis and the Calculus of Moving Surfaces*. Springer-Verlag, 2013.
- [24] T. Günther, M. Gross, and H. Theisel. Generic objective vortices for flow visualization. *ACM Transactions on Graphics*, 36(4):Article No. 141, 2017.
- [25] T. Günther, M. Schulze, and H. Theisel. Rotation invariant vortices for flow visualization. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):817–826, 2016.
- [26] T. Günther and H. Theisel. Hyper-objective vortices. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):1–1, 2018.
- [27] T. Günther and H. Theisel. The State of the Art in Vortex Extraction. *Computer Graphics Forum*, 37(6):149–173, 2018.
- [28] A. Hadjighasem and G. Haller. Geodesic transport barriers in Jupiter’s atmosphere: A video-based analysis. *SIAM Review*, 58(1):69–89, 2016.
- [29] M. Hadwiger, M. Mlejnek, T. Theußl, and P. Rautek. Time-dependent flow seen through approximate observer Killing fields. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):1257–1266, 2019.
- [30] G. Haller. An objective definition of a vortex. *Journal of Fluid Mechanics*, 525:1–26, 2005.
- [31] G. Haller, A. Hadjighasem, M. Farazmand, and F. Huhn. Defining coherent vortices objectively from the vorticity. *Journal of Fluid Mechanics*, 795:136–173, 2016.
- [32] D. D. Holm, T. Schmah, and C. Stoica. *Geometric Mechanics and Symmetry: From Finite to Infinite Dimensions*. Oxford University Press, 2009.
- [33] G. A. Holzapfel. *Nonlinear Solid Mechanics: A Continuum Approach for Engineering*. Wiley, 2000.
- [34] B. L. Hua and P. Klein. An exact criterion for the stirring properties of nearly two-dimensional turbulence. *Physica D: Nonlinear Phenomena*, 113(1):98–110, 1998.
- [35] J. Huang, W. Pei, C. Wen, G. Chen, W. Chen, and H. Bao. Output-coherent image-space LIC for surface flow visualization. In *Proceedings of IEEE Pacific Visualization*, pp. 137–144, 2012.
- [36] B. Jobard, G. Erlebacher, and M. Y. Hussaini. Lagrangian-Eulerian advection of noise and dye textures for unsteady flow visualization. *IEEE Transactions on Visualization and Computer Graphics*, 8(3):211–222, 2002.
- [37] T. Langer, A. Belyaev, and H.-P. Seidel. Spherical barycentric coordinates. In *Proceedings of Eurographics Symposium on Geometry Processing*, pp. 81–88, 2006.
- [38] R. S. Laramee, B. Jobard, and H. Hauser. Image space based visualization of unsteady flow on surfaces. In *Proceedings of IEEE Visualization 2003*, pp. 123–130, 2003.
- [39] J. Laurie, G. Boffetta, G. Falkovich, I. Kolokolov, and V. Lebedev. Universal profile of the vortex condensate in two-dimensional turbulence. *Phys. Rev. Lett.*, 113(25):254503, 2014.
- [40] J. M. Lee. *Introduction to Smooth Manifolds*. Springer-Verlag, 2nd ed., 2012.
- [41] J. M. Lee. *Introduction to Riemannian Manifolds*. Springer-Verlag, 2nd ed., 2018.
- [42] G.-S. Li, X. Tricoche, and C. Hansen. GPUFLIC: Interactive and accurate dense visualization of unsteady flows. In *Proceedings of Eurovis 2006*, pp. 29–34, 2006.
- [43] G.-S. Li, X. Tricoche, D. Weiskopf, and C. Hansen. Flow charts: Visualization of vector fields on arbitrary surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 14(5):1067–1080, 2008.
- [44] J. E. Marsden and T. J. Hughes. *Mathematical Foundations of Elasticity*. Dover Publications, Inc., 1994.
- [45] A. McInerney. *First Steps in Differential Geometry*. Springer-Verlag, 2013.
- [46] S. B. Myers. Isometries of 2-dimensional riemannian manifolds into themselves. *Proc. National Academy of Sciences of the USA*, 22(5):297–300, 1936.
- [47] R. W. Ogden. *Non-Linear Elastic Deformations*. Dover Publications, Inc., 1997.
- [48] B. O’Neill. *Elementary Differential Geometry*. Academic Press, revised 2nd ed., 2006.
- [49] J. Palacios and E. Zhang. Rotational symmetry field design on surfaces. *ACM Transactions on Graphics*, 26(3):Article No. 55, 2007.
- [50] J. Pedlosky. *Geophysical Fluid Dynamics*. Springer-Verlag, 2nd ed., 1987.
- [51] Z. Peng and R. S. Laramee. Higher dimensional vector field visualization: A survey. In *Proceedings of EG UK Theory and Practice of Computer Graphics*, pp. 149–163, 2009.
- [52] P. Petersen. *Riemannian Geometry*. Springer-Verlag, 3rd ed., 2016.
- [53] I. B. Rojo and T. Günther. Vector field topology of time-dependent flows in a steady reference frame. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):280–290, 2020.
- [54] P. G. Saffman. *Vortex Dynamics*. Cambridge University Press, 1995.
- [55] H.-W. Shen and D. L. Kao. UFLIC: A line integral convolution algorithm for visualizing unsteady flows. In *Proceedings of IEEE Visualization ’97*, pp. 317–322, 1997.
- [56] J. Sokol. Flattened fluids help scientists understand oceans and atmospheres, July 2018. [Online: www.wired.com/story/flattened-fluids-help-scientists-understand-oceans-and-atmospheres/; posted 27-June-2018].

- [57] J. Solomon. Discrete killing fields for pattern synthesis and symmetry detection. Undergraduate honors thesis, Stanford University, Palo Alto, CA, 2010.
- [58] M. Spivak. *A Comprehensive Introduction to Differential Geometry (5 volumes)*. Publish or Perish Press, 3rd ed., 1999.
- [59] D. Stalling. LIC on surfaces. Technical report, Konrad Zuse Institute Berlin (ZIB), Berlin, Germany, 1997.
- [60] D. Stalling and H.-C. Hege. Fast and resolution independent line integral convolution. In *Proceedings of SIGGRAPH '95*, pp. 249–256, 1995.
- [61] D. Sujudi and R. Haimes. Identification of swirling flow in 3-d vector fields. In *Proceedings of the 12th Computational Fluid Dynamics Conference*, pp. 792–799, 1995.
- [62] C. Teitzel, R. Gross, and T. Ertl. Line integral convolution on triangulated surfaces. In *Proceedings of WSCG*, pp. 572–581, 1997.
- [63] K. S. Thorne and R. D. Blandford. *Modern Classical Physics: Optics, Fluids, Plasmas, Elasticity, Relativity, and Statistical Physics*. Princeton University Press, 2017.
- [64] Y. Tong, S. Lombeyda, A. N. Hirani, and M. Desbrun. Discrete multiscale vector field decomposition. In *Proceedings of SIGGRAPH*, pp. 445–452, 2003.
- [65] C. Truesdell and W. Noll. *The Nonlinear Field Theories of Mechanics*. Springer-Verlag, 1965.
- [66] L. W. Tu. *Differential Geometry: Connections, Curvature, and Characteristic Classes*. Springer-Verlag, 2017.
- [67] J. J. van Wijk. Image based flow visualization. *ACM Transactions on Graphics*, 21(3):745–754, 2002.
- [68] J. J. van Wijk. Image based flow visualization for curved surfaces. In *Proceedings of IEEE Visualization 2003*, pp. 123–130, 2003.
- [69] A. Vaxman, M. Campen, O. Diamanti, D. Panozzo, D. Bommes, K. Hildebrandt, and M. Ben-Chen. Directional Field Synthesis, Design, and Processing. *Computer Graphics Forum*, 35(2):545–572, 2016.
- [70] T. Weinkauf, J. Sahner, H. Theisel, and H.-C. Hege. Cores of swirling particle motion in unsteady flows. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1759–1766, 2007.
- [71] E. Zhang, K. Mischaikow, and G. Turk. Vector field design on surfaces. *ACM Transactions on Graphics*, 25(4):1294–1326, 2006.