Smoke Rings from Smoke

Steffen Weißmann (TU Berlin) Ulrich Pinkall (TU Berlin) Peter Schröder (Caltech)

Abstract

We give an algorithm which extracts vortex filaments ("smoke rings") from a given 3D velocity field. Given a filament strength $h > 0$, an optimal number of vortex filaments, together with their extent and placement, is given by the zero set of a complex valued function over the domain. This function is the global minimizer of a quadratic energy based on a Schrödinger operator. Computationally this amounts to finding the eigenvector belonging to the smallest eigenvalue of a Laplacian type sparse matrix.

Turning traditional vector field representations of flows, for example, on a regular grid, into a corresponding set of vortex filaments is useful for visualization, analysis of measured flows, hybrid simulation methods, and sparse representations. To demonstrate our method we give examples from each of these.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

Keywords: Vortex filaments, DEC, Schrödinger operator, vortex reconnection, hairpin removal, hybrid methods, fluid simulation

Links: DL PDF VIDEO CODE

1 Introduction

Many velocity fields common in nature can be encoded by *vortex filaments* in an extraordinarily sparse and efficient way. Such collections of directed curves (closed or beginning and ending on the boundary) arise because vorticity always originates in 2D sheets on interfaces, *e.g.*, on obstacles or between hot and cold fluid, and then rapidly rolls up into 1D structures. Simulation methods based on vortex filaments have been described in the CFD literature [Chorin 1990; Chorin 1993; Bernard 2006; Bernard 2009] as well as in Computer Graphics [Angelidis and Neyret 2005; Weißmann and Pinkall 2010], and offer a number of advantages over grid-based methods:

- no gridding of the domain is required, facilitating simulations in unbounded domains;
- energy and momentum conservation are achieved easily and problems such as vorticity diffusion avoided;
- filaments gracefully handle phenomena, such as leapfrogging vortex rings, which are difficult to reproduce with grid-based methods;
- animators appreciate the intuitive control of fluid flow through manipulating sets of curves.

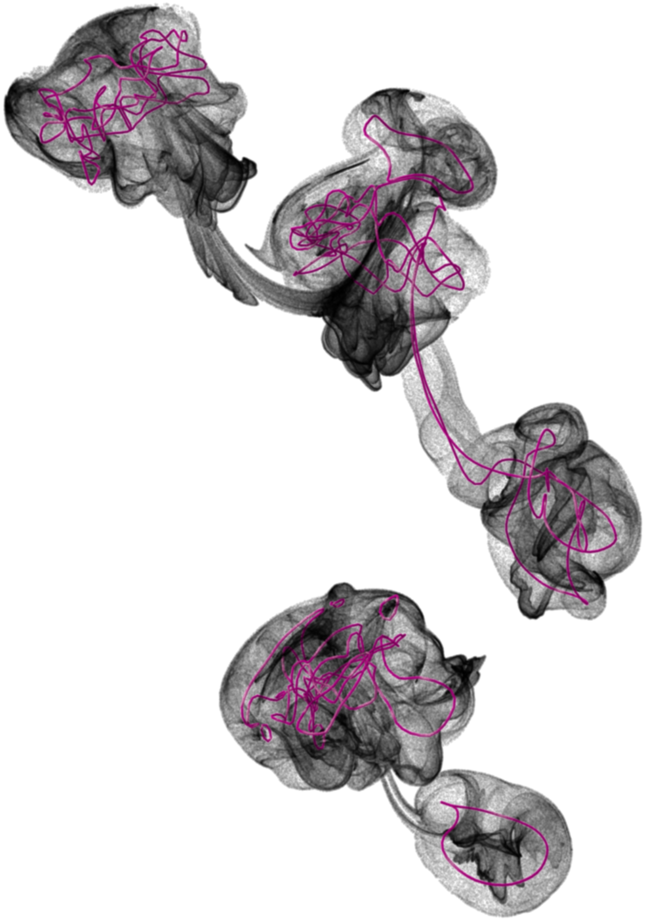


Figure 1: *Using a frame from a standard velocity simulation, we convert it to vortex filaments and then evolve these. This results in finer detail than the grid simulation alone could have resolved.*

For these reasons, filaments have been used in production and are now part of the Houdini™ FX animation software [Side Effects Software Inc. 2013]. What is missing so far is a method to *extract* a vortex filament representation from a given velocity field. Such a method

1. provides a tool to analyse and visualize measured and computed flows;
2. enables simplification and level-of-detail for filament representations;
3. allows for new hybrid solvers (*e.g.*, see Fig. 1).

We provide the first such method to extract vortex filaments from velocity fields. It is based on a novel criterion for the comparison of singular vortex filaments and continuous vorticity. In practice, our algorithm amounts to setting up a data dependent sparse Laplacian matrix and using a standard linear algebra package to find the eigenvector corresponding to its smallest eigenvalue. The filaments are then extracted through 1D contour tracing.

ACM Reference Format
Weißmann, S., Pinkall, U., Schröder, P. 2014. Smoke Rings from Smoke. ACM Trans. Graph. 33, 4, Article 140 (July 2014), 8 pages. DOI = 10.1145/2601097.2601171 http://doi.acm.org/10.1145/2601097.2601171.

DOI: http://doi.acm.org/10.1145/2601097.2601171

1.1 Related Work

While there is no antecedent for the ability to extract vortex filaments from velocity fields, detection, visualization, and simulation of vortex primitives has been pursued and we review work from these areas briefly here.

Vortex *detection* is essential to the analysis of flow fields. Jiang and co-workers [2005] provide a taxonomy of available methods, in terms of which our approach finds directed structure, *i.e.*, *lines* rather than just vortex *regions*; works *globally* rather than *locally*; and is *Galilei invariant*. The latter is a requirement for identifying moving vortices. Since our method is *gauge invariant* (see App. A.1) it is in particular Galilei invariant.

Vortex *visualization* is often done through volumetric color coding or level sets of the norm of vorticity, *e.g.*, [Chatelain et al. 2008; Troolin and Longmire 2010; Le et al. 2011]. Iso-surfaces in particular can be quite misleading, as they do not necessarily reveal the topology of the vorticity field lines. Extracting vortex filaments, as we do, produces more reliable and geometrically meaningful results, in particular for measured data with its high noise (see Sec. 4.3).

Vortex *simulation* methods also use sheets and particles [Stock et al. 2008; Kim et al. 2009; Brochu et al. 2012; Pfaff et al. 2012; Golas et al. 2012], often for *hybrid* approaches. Pfaff and coworkers [2012], for example, use a high-detail vortex sheet on the boundary of a buoyant region while a grid-based solver handles bulk motion. Golas and co-workers [2012] use vortex particles to deal with unbounded domains and grid-based solvers for boundary handling. To match a given velocity they seed vortex particles iteratively in a local, greedy fashion, while we solve globally for vortex filaments without fixing the location or number of such filaments.

To summarize, an efficient method for the extraction of vortex filaments from velocity fields has broad applicability across a wide spectrum of applications and is so far missing.

2 Ansatz

Problem Statement: Given a velocity field $\mathbf{v}$ over some domain $U \subset \mathbb{R}^3$ and a strength parameter $h > 0$, find a set of oriented curves $\Gamma = \{\gamma_1, \ldots\}$ that, when interpreted as vortex filaments of strength h, provide a parsimonious approximation of the vorticity field $\boldsymbol{\omega} = \operatorname{curl} \mathbf{v}$.

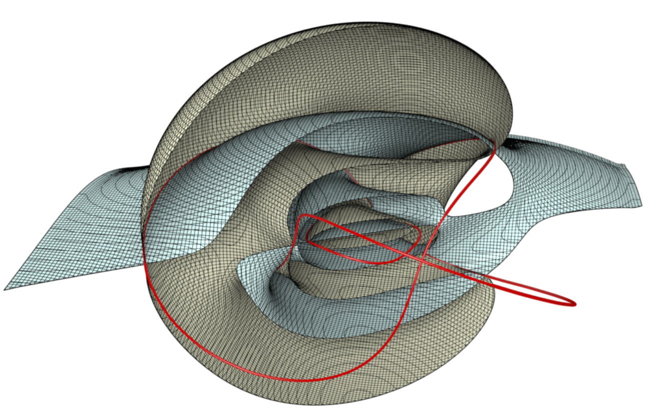


Figure 2: *Filaments (red) are given by the intersection of the zero contours of* $\mathrm{Re}(\psi)$ *(blue) and* $\mathrm{Im}(\psi)$ *(yellow). (Compare Fig. 5.)*

To formulate an optimality criterion for this task we need to choose (i) a representation for Γ and (ii) a measure of the quality of the approximation. To avoid a difficult combinatorial (how many?) optimization problem we represent Γ implicitly as the zero set of a complex function $\psi : U \to \mathbb{C}$. The zero set corresponds to intersections of the zero contours of $\mathrm{Re}(\psi)$ and $\mathrm{Im}(\psi)$, which are, generically, transversally intersecting smooth surfaces. Hence their intersection generically consists of 1D curves which are either closed or begin and end on the boundary of U (Fig. 2 & movie).

Since vortex filaments are singular we cannot compare them directly with a continuous vector field $\boldsymbol{\omega}$. Instead we ask that the vorticity $\boldsymbol{\omega}$ of the original field, integrated over a test surface, be close to the (signed) sum of vortex filaments crossing this test surface (see Fig. 3). This is analogous to judging a halftoned image by comparing the number of black and white pixels (signed crossings) in a given region (test surface) with its gray content (integrated vorticity).

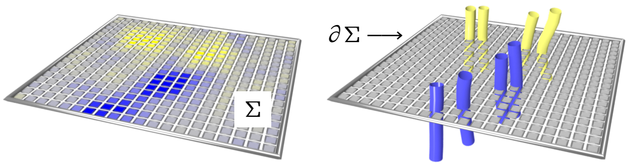


Figure 3: *Left: a test surface* Σ *colored by vorticity. Right: vortex filaments crossing this test surface. The (signed) sum of filaments "threading" through the boundary* $\partial\Sigma$*, multiplied by their strength* $h > 0$*, should approximate the integral of vorticity over* Σ *well.*

To turn this quality criterion into a computationally feasible convex optimization problem we need to study the relationship between ψ and measures of vorticity. We express our derivations in the language of *exterior calculus*. For readers new to the subject we recommend the course notes [Crane et al. 2013] for a basic introduction. Note that we use *complex valued* differential forms, which are not fundamentally different from the standard case of real valued forms. Readers whose primary interest is in the implementation may choose to skip ahead to Sec. 3.

First note that for a general $\psi = re^{i\alpha}$ its differential is $d\psi = dre^{i\alpha} + i\psi d\alpha$ and thus

$$d\alpha = d\arg\psi = \frac{\langle d\psi, i\psi\rangle}{|\psi|^2},$$

using the real inner product $\langle u, v\rangle = \mathrm{Re}(\bar{u}v)$ for $u, v \in \mathbb{C}$. Knowing $d\arg\psi$ we can compute the *winding number* $n \in \mathbb{Z}$ of ψ (with respect to the origin) around the boundary of a surface Σ as

$$n = \frac{1}{2\pi}\int_{\partial\Sigma} d\arg\psi = \frac{1}{2\pi}\int_{\partial\Sigma} \frac{\langle d\psi, i\psi\rangle}{|\psi|^2} \tag{1}$$

(here we assumed $\psi \neq 0$ on $\partial\Sigma$). This integral counts the number of full turns the vector ψ makes along $\partial\Sigma$ [Wikipedia 2014b], which in turn reveals the (signed) sum of zeros of ψ crossing Σ (as a consequence of Stokes' theorem). Fig. 4 demonstrates an example showing the complex numbers $\psi(p)$ displayed as vectors (left) resp. normalized vectors (right) for $p \in \Sigma$. One can verify here that the number of full turns performed by ψ along the boundary loop equals the (signed) sum of zeros of ψ (resp. singularities of the normalized field) on the inside.

We will now compare this with the vorticity of the given velocity field $\mathbf{v}$ as measured over Σ. Define the 1-form $\eta := \langle \mathbf{v}, .\rangle$ and integrate the vorticity 2-form $d\eta$ over Σ

$$\int_{\Sigma} d\eta = \int_{\partial\Sigma} \eta = \int_{\partial\Sigma} \frac{\langle i\eta\psi, i\psi\rangle}{|\psi|^2}, \tag{2}$$

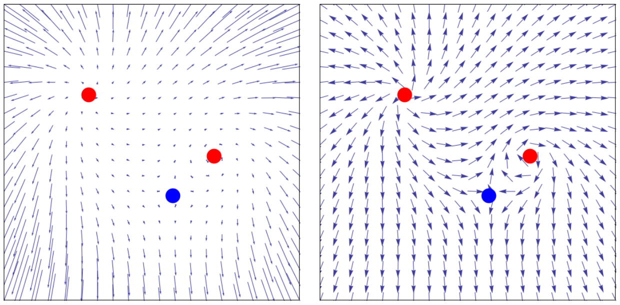


Figure 4: *A signed (red: $+1$; blue: -1) count of zeros of ψ on Σ is given by its winding number around $\partial\Sigma$. This also corresponds to a signed count of the singularities of $\psi/|\psi|$ within Σ.*

where the first equality follows from Stokes' theorem and the second equality from $1 = \langle i\psi, i\psi\rangle/|\psi|^2$.

Our approximation will be good if $hn \approx \int_{\partial\Sigma}\eta$. Using the shorthand $\hbar = \frac{h}{2\pi}$ and Eqs. (1) and (2) this amounts to asking that

$$\int_{\partial\Sigma} \frac{\hbar\langle d\psi - i\hat{\eta}\psi, i\psi\rangle}{|\psi|^2} \tag{3}$$

be small in magnitude for $\hat{\eta} = \frac{\eta}{\hbar}$. If $|\psi|$ is bounded away from zero, *i.e.*, $\partial\Sigma$ maintains some minimal distance to the zero set of ψ, the integral in Eq. (3) will certainly be small if the 1-form $d\psi - i\hat{\eta}\psi$ is small.

As a function of ψ this expression encodes a particular differential which we will denote d^∇ and henceforth write $d^\nabla\psi := (d - i\hat{\eta})\psi = d\psi - i\hat{\eta}\psi$.

To ensure that the 1-form $d^\nabla\psi$ is small in magnitude we seek ψ which minimizes the quadratic, convex, Dirichlet like energy

$$E(\psi) = \frac{\hbar^2}{2}\|d^\nabla\psi\|^2 = \frac{\hbar^2}{2}\langle\langle d^\nabla\psi, d^\nabla\psi\rangle\rangle, \tag{4}$$

subject to a unit L^2 norm constraint, *i.e.*, $\|\psi\|^2 = 1$ (note that this is not a pointwise constraint on ψ) and suitable boundary conditions (more on those in App. A). Here $\langle\langle\alpha,\beta\rangle\rangle = \int_U \bar{\alpha}\wedge *\beta$ denotes the standard L^2 Hermitian product on complex valued k-forms α and β.

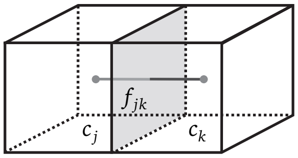
Summary In our derivation we used the concept of test surfaces and measurements across these, namely that the number of filaments (zeros of ψ) crossing such a test surface, multiplied by their strength ($h > 0$), should be close to the vorticity of the input vector field $\mathbf{v}$ over such test surfaces. We showed that this approximation is indeed a good one if the L^2 norm of $d^\nabla\psi$ is small *over all of U*. In particular the energy to be minimized (Eq. (4)) *no longer depends on test surfaces*. That such a minimizing solution for ψ indeed yields small approximation error when measured across particular test surfaces, is the subject of Sec. 4.2.

In summary then, **the sought after smoke ring set Γ (of strength h) is the zero set of the minimizer ψ of Eq.** (4).

In App. A we show that Eq. (4) is the Hamiltonian of a quantum mechanical particle with unit mass and unit charge moving in a magnetic field $\mathbf{B} = \operatorname{curl}\mathbf{v}$. Thus the minimizer of Eq. (4) is the eigenfunction belonging to the smallest eigenvalue of the *magnetic Schrödinger* operator. Importantly we can use numerical methods from computational physics to find it (see Sec. 3).

3 Implementation

Numerical methods for the minimization of Eq. (4) on regular Cartesian grids were given by [Governale and Ungarelli 1998; Halvorsen and Kvaal 2012]. Since this is the most common case in practice we use it as a basis of our implementation (see also the elementary Matlab code provided as supplemental material). A more general derivation applicable to non-uniform grids as well as simplicial meshes can be found in App. B.

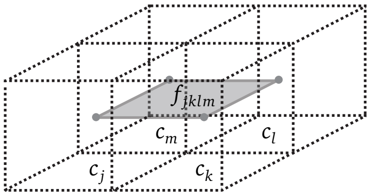


Let the input velocity field be given on a staggered Cartesian grid with spacing δ (uniform along x-, y-, and z-axes) and let $\mathbf{v}_{jk}$ denote the velocity component stored on the facet f_{jk} between grid cells c_j and c_k. Given a filament strength $h > 0$ define the data dependent matrix E^δ

$$E^\delta_{jk} = -e^{-i\hat{\eta}_{jk}}, \qquad E^\delta_{kj} = \bar{E}^\delta_{jk}, \qquad E^\delta_{jj} = d, \qquad \hat{\eta}_{jk} = \frac{\delta}{\hbar}\mathbf{v}_{jk},$$

where d is the number of cell neighbors of c_j. For input in the form of 3D velocity samples on cell centers, set $\mathbf{v}_{jk}$ to the corresponding component of the three vector $(\mathbf{v}_j + \mathbf{v}_k)/2$.

With E^δ in hand, find the eigenvector ψ belonging to its smallest eigenvalue. We used the iterative eigensolver PRIMME [Stathopoulos and McCombs 2010] with ILUT preconditioning [Saad 2005] (direct methods are not practical due to the size of E^δ). If the chosen eigensolver supports only real symmetric matrices, convert E^δ by replacing all complex numbers $a + ib$ with 2×2 block matrices $\left(\begin{smallmatrix} a & -b \\ b & a\end{smallmatrix}\right)$.



Given the minimizer ψ_j we extract the filament set Γ as the zero set of a tri-linear interpolation of ψ_j. Here we make the genericity assumption that this tri-linear interpolation has zeros only in the open interior of any facet f_{jklm} (formed by 4 consecutive centers $\{c_j, c_k, c_l, c_m\}$). The associated values $\psi_{j,k,l,m}$ form a quadrilateral in the complex plane whose edges do not pass through the origin with winding number

$$n_{jklm} = \frac{1}{2\pi}\left(\arg(\tfrac{\psi_k}{\psi_j}) + \arg(\tfrac{\psi_l}{\psi_k}) + \arg(\tfrac{\psi_m}{\psi_l}) + \arg(\tfrac{\psi_j}{\psi_m})\right).$$

Since all four summands lie in the open interval between $(-\pi,\pi)$

$$n_{jklm} \in \{-1, 0, 1\}.$$

The curves γ in Γ only intersect faces with $n_{jklm} = \pm 1$. We locate the position of these intersection points using bi-linear interpolation

$$0 = (1-v)((1-u)\psi_j + u\psi_k) + v((1-u)\psi_m + u\psi_l). \tag{5}$$

This equation implies the vanishing of

$$\det((1-u)\psi_j + u\psi_k, (1-u)\psi_m + u\psi_l),$$

where $\det(z,w) = -i\,\mathrm{Im}(\bar{z}w)$ is the real determinant of complex numbers. The solutions to this quadratic equation in u give a corresponding v via Eq. (5). Because the winding number is ± 1 we know that exactly one of the resulting (u,v)-pairs lies in $(0,1)\times(0,1)$.

In this way we determine intersection points for any face f_{jklm} with a non-zero winding number. For a given cube of 6 faces we know that the indices must sum to zero because the 24 angles involved cancel in pairs. Hence we have one, two or three "incoming faces" and an equal number of "outgoing faces" for any cube crossed by the zero curves.

Producing Γ It remains to determine how to join entry and exit points in the interior of a cube. In case there is more than one entry/exit pair for a given cube, we connect them to minimize the total length of the joining edges. This choice is consistent with the variational principle underlying reconnection, which favors shorter curves [Weißmann and Pinkall 2010]. The whole set Γ is produced by tracing. Starting with boundary faces of non-zero winding number follow the (directed) curve until it exits another boundary face (curves that start on the boundary must end on the boundary). Once those are exhausted we may choose any remaining face with a non-zero winding number and trace the corresponding closed curve until all such faces are exhausted.

In our implementation we use OpenVDB [Museth 2013] as a sparse, infinite grid data structure, storing in it only faces with non-zero winding number.

4 Numerical Experiments

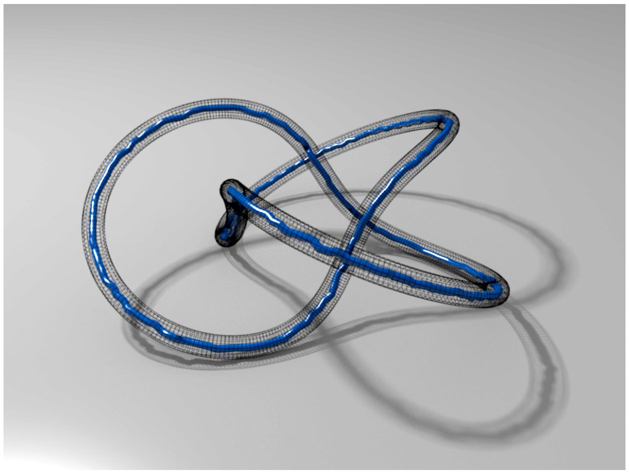
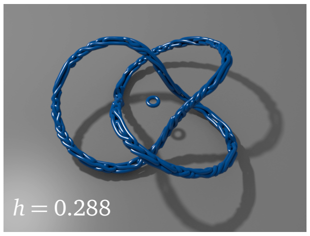


Figure 5: *For a sharp vortex ring with strength $h = 1$ we find a single filament within a narrow tubular neighborhood (no smoothing was performed on the extracted piecewise linear curve).*

4.1 Influence of h

We begin our experiments with velocity fields arising from a unit strength vortex filament. Taking a trefoil knot (studied experimentally in [Kleckner and Irvine 2013]), we sample its velocity via the Biot-Savart law on the facets of an $80 \times 60 \times 80$ grid. For $h = 1$ our method recovers the input trefoil within discretization error (Fig. 5). Increasing h above 1 results in coarser approximations (Fig. 6 & movie).



Letting h be a fraction of the strength of the input vortex ring produces "ropes." Here h is slightly less than 1/3 resulting in three vortex rings with an additional small ring in the center. As h continues to decrease, the small center ring grows and eventually joins the "rope" and so on (see movie). In general, small rings appear whenever necessary in order to optimally approximate the velocity field with given filament strength h. This can result in undesired effects when visualization is the primary objective, making it necessary to fine-tune the choice of h.

To understand the appearance of ropes consider a test surface Σ intersecting the trefoil. For $h \approx 1/n$ the winding number of ψ should be n, *i.e.*, we expect n filament intersections with Σ. Since filaments do not overlap and are resolved at the grid size (*i.e.*, any grid facet can intersect with a single filament only), "ropes" result. Details of their braiding have little impact on the velocity field and the rope as a whole carries the correct vorticity. In contrast, for smoother vorticity fields (*e.g.*, from measurements or grid based simulations), the occurrence of "ropes" corresponds to aliasing effects due to undersampling of the velocity field relative to h and indicates that h was chosen too small for the given data (see below).

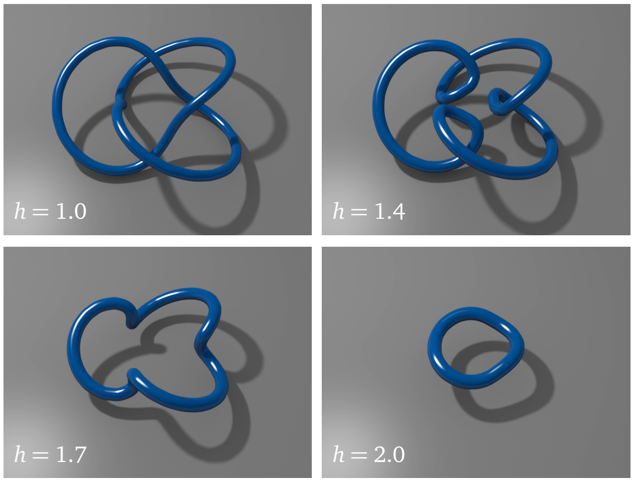


Figure 6: *As h increases ever more coarse approximations result (filament cross sections are proportional to h).*

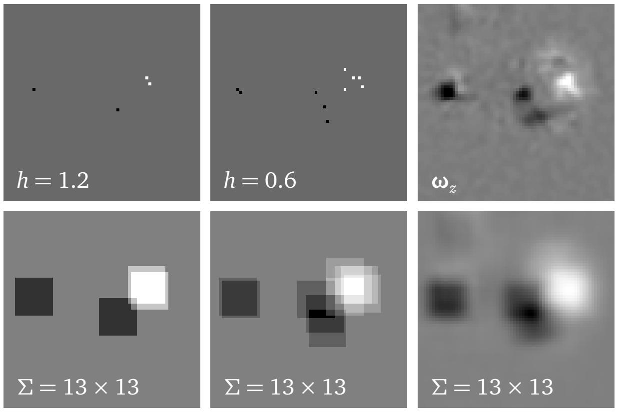
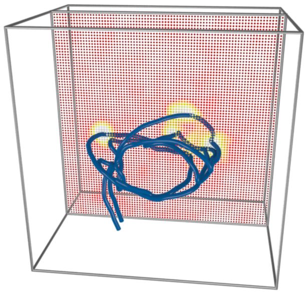


Figure 7: *Slice through a volume showing filament intersections (top left; middle) and vorticity (top right). Underneath the corresponding Σ integrals. Note excellent agreement between bottom middle and right.*

4.2 Agreement on Test Surfaces



Our approach is predicated on (h-times) the number of vortex filament intersections in a test surface Σ approximating vorticity flux over Σ. As an example we take a slice orthogonal to the z-axis of a time frame of the measurements in [Troolin and Longmire 2010] (the inset Fig. shows vorticity on such a slice with an extracted vortex filament). Fig. 7 shows the results on this slice for differing h and the corresponding test surface integrals. The excellent agreement between the bottom middle and right image supports our claim: the vortex filaments exhibit vorticity on test surfaces similar to that of the input field.

Taking the difference between filament and vorticity flux through Σ, Fig. 8 shows the mean error as a function of h for three different test surface sizes: 7×7 (blue), 15×15 (purple), and 21×21 (red). On the left synthetic data (a blurred version of the velocity field in Sec. 4.1) and on the right measured data (see Sec. 4.3). We observe a reduction in error as h becomes smaller with a rate independent of the test surface size though depending on the type of input data.

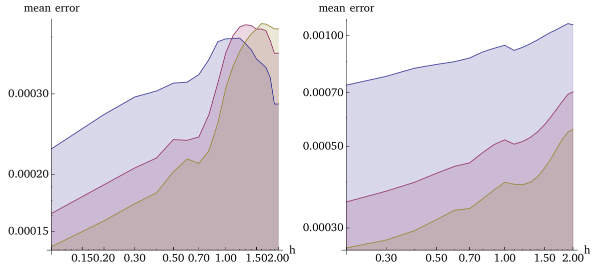


Figure 8: *Mean error of our method as a function of h for different Σ plotted on a log/log scale. Left: synthetic data; right: measured data.*

Choice of h The strength parameter h controls level-of-detail in the resulting filament representation, and its choice depends on the application at hand. Due to discretization artifacts (see discussion on "ropes" in Sec. 4.1) it is not possible to choose h arbitrarily small. In fact there is a data-dependent *lower bound* below which aliasing effects from undersampling occur. This bound depends on both grid resolution and velocity data, and so far we have no automatic method to determine it. In practice we found that running the minimization with a couple of different h values quickly reveals a good value of h.

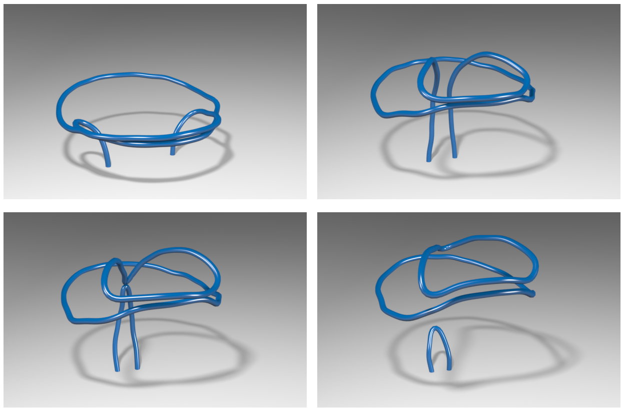
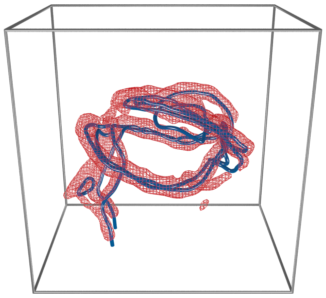


Figure 9: *Result (4 time frames) of applying our method to the measured data set of Troolin and co-workers (see movie).*

4.3 Extracting Features from Measured Data

We now apply our method to measured velocity data of a jet emanating from an inclined nozzle [Troolin and Longmire 2010]. The velocity of $\approx 35{,}000$ tracer particles was tracked and interpolated onto a regular $80 \times 80 \times 60$ grid. Compare also the numerical simulation of the same setup in [Le et al. 2011].

As is evident from [Troolin and Longmire 2010], interpreting the resulting vortex structure in terms of several interacting vortex rings is crucial for understanding the underlying dynamics. The original visualization was based on level sets of the vorticity norm with superimposed velocity vectors (the movie shows such a level set without additional velocity vectors). Our method produces the vortex filaments directly and faithfully records the topology changes that occur during the time evolution of the flow (Fig. 9).



The filament strength in Fig. 9 matches the strength of the dominant vortex rings. With 1/2 the filament strength more of the fine level structure of the flow is resolved (shown here together with a level surface of vorticity magnitude).

4.4 Reconnection

In practice, filament-based simulations must deal with the exponential growth in filament detail, much of which contributes almost nothing to the velocity field. This is achieved by changing the topology through *reconnection* and in particular the removal of *hairpins* [Chorin 1990; Chorin 1993; Weißmann and Pinkall 2010]. These approaches are local. Instead we can proceed globally by converting the current set of filaments to velocities on a grid (using, *e.g.*, Biot-Savart or a Poisson solver) followed immediately by extracting filaments of the same strength. Fig. 10 shows the result on the right when starting with the filaments on the left. Here the total length of filaments was reduced by 40%.

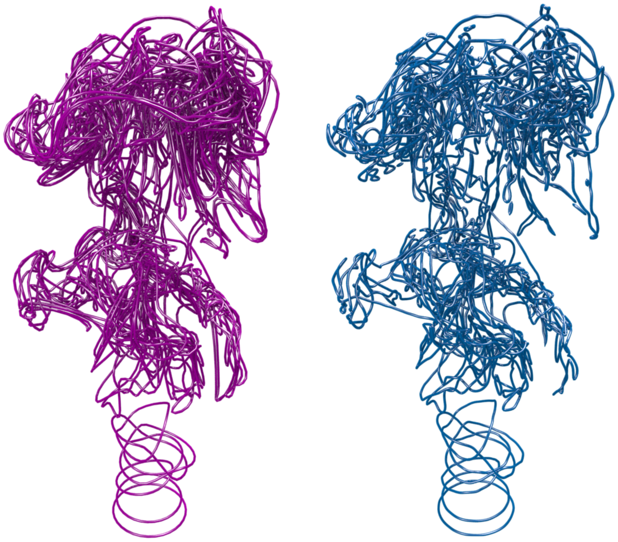


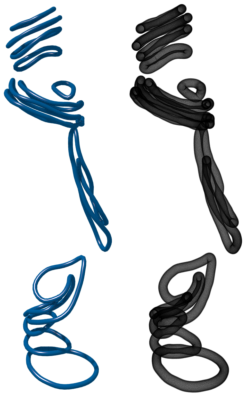
Figure 10: *Left: A filament jet that has evolved without reconnection. Right: The result of decomposing the velocity field of the filaments on the left into filaments of the same strength (see movie).*

This method can also be used to create level-of-detail approximations by performing the extraction with a larger h. Such *re-quantization* is shown in Fig. 11 (and the movie) and can be useful for time constrained simulations.

Figure 11: *Re-quantization of the same jet as in Fig. 10 with filaments whose strength equals two, three or four times the original filament strength.*

4.5 Increasing Detail in Velocity Fields

When animating fluids there often is a lack of fine level detail, due to, *e.g.*, insufficient grid resolution or excessive numerical diffusion. We can improve this situation with a process we call "velocity stippling": given a velocity field we extract vortex filaments and immediately convert them back to a velocity field. This process tends to give regions of high average vorticity fine level detail in the form of numerous filaments while keeping the overall appearance of the flow. This is so because the method interprets the vorticity seen in a certain region as an averaged version of a vorticity that has already rolled up into filaments and then reconstructs those filaments which in turn give rise to a highly detailed velocity field in those regions.



To see velocity stippling at work consider first a time frame of a simulation produced with a standard grid solver. The inset Fig. shows vortex filaments extracted from such a frame (left). To track different evolutions we use the same seed particles (right) for all our experiments. Now we consider three different experiments starting from this frame. In the first one we keep all the original velocity data and simply advect the particles seeded near the filaments (Fig. 12, left). This is the simulation that lacks detail which we wish to improve. In the next experiment we apply our velocity stippling algorithm. We turn the vortex filaments back into a velocity field and use it as initial condition for the standard grid solver (Fig. 12, middle). This single conversion to filaments and back to velocity, while still using the standard grid solver, already shows more detail. Finally, we can also use the extracted filaments and evolve them with a filament solver. This of course yields the most detail (Fig. 1 & Fig. 12, right). In the last case the particles are still advected on a grid by advecting them in the velocity field generated from the filament simulation at each step (using Biot-Savart or a Poisson solver).

Performance Most of the computation time is spent on solving for the smallest eigenvector ψ. Initializing the sparse matrix E^{δ} and tracing the filaments is negligible in all experiments. The trefoil example and the measured velocity data (grid size for both $80 \times 60 \times 80$) take about 15 sec. to compute. Finding ψ for the filament jet (grid size $80 \times 160 \times 80$) takes less than 60 sec., tracing the 335 polylines (with 20,052 vertices) takes 680 ms.

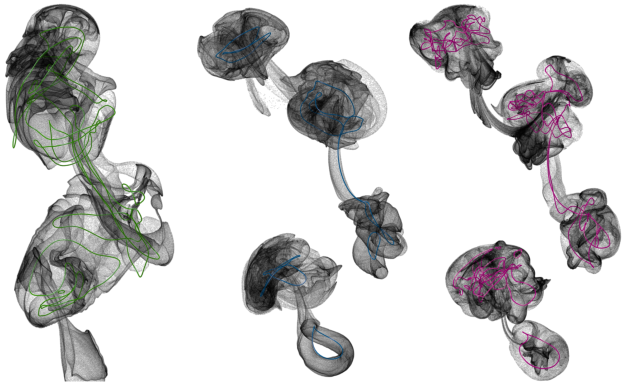


Figure 12: *The baseline simulation (left) shows advection of particles in a grid based solver velocity field. Evolving the stippled velocity field with the grid based solver yields more detail (middle), while the highest detail results from a filament solver (right). The superimposed filaments also show the different degrees of detail (also see movie).*

All simulations were computed on a Macbook Pro with 2.7 GHz Intel Core i7 and 16 GB RAM. A sequence of frames (coherent either in time or in h) typically requires only 1/2 the time when using a previous frame ψ as initial guess for the eigensolver at the next frame.

5 Conclusion

We demonstrated that a rather simple computational procedure (Sec. 3) can reliably extract vortex filaments from given velocity data. While the usefulness of this algorithm is clear for flow visualization and analysis, we see a large unexplored territory concerning possible further applications within the work flow of fluid modelling and simulation. Our example of velocity stippling (Sec. 4.5) used stippling only once and already achieved a marked increase in detail. The algorithm is fast enough that one can imagine integrating it into a fluid solver on a per-time-frame basis. We believe that we have only scratched the surface of the possibilities in this area.

While we demonstrated empirically that the quality of the filament representation improves with smaller h, the relationship between h, grid resolution and velocity input deserves a more detailed analysis.

Our algorithm is also applicable in other physical scenarios where filaments play a role. These include electromagnetism and the simulation and visualization of solar flares in magnetohydrodynamics.

Much work also remains to be done in the underlying mathematical theory. Efforts in this direction seem all the more worthwhile since the same type of mathematical machinery (related to Lattice Gauge Theory) also occurred in the seemingly unrelated context of placing direction field singularities on surfaces [Knöppel et al. 2013].

Acknowledgments

Dan Troolin (TSI Inc.) kindly provided the velocity data used in Sec. 4.3. We thank Matthias Durynek for his support with the water vapor photograph and Albert Chern for the Matlab implementation. This work was supported in part by DFG Research Center Matheon, SFB/Transregio 109 "Discretization in Geometry and Dynamics," and Side Effects Software.

References

ANGELIDIS, A., AND NEYRET, F. 2005. Simulation of Smoke based on Vortex Filament Primitives. In *Proc. Symp. Comp. Anim.*, 87–96.

BERNARD, P. S. 2006. Turbulent Flow Properties of Large-scale Vortex Systems. *PNAS 103*, 27, 10174–10179.

BERNARD, P. S. 2009. Vortex Filament Simulation of the Turbulent Coflowing Jet. *Phys. Fluids 21*, 2.

BROCHU, T., KEELER, T., AND BRIDSON, R. 2012. Linear-Time Smoke Animation with Vortex Sheet Meshes. In *Proc. Symp. Comp. Anim.*, 87–95.

CHATELAIN, P., CURIONI, A., BERGDORF, M., ROSSINELLI, D., ANDREONI, W., AND KOUMOUTSAKOS, P. 2008. Billion Vortex Particle Direct Numerical Simulations of Aircraft Wakes. *Comp. Meth. Appl. Mech. & Eng. 197*, 13–16, 1296–1304.

CHORIN, A. J. 1990. Hairpin Removal in Vortex Interactions. *J. Comput. Phys. 91*, 1, 1–21.

CHORIN, A. J. 1993. Hairpin Removal in Vortex Interactions II. *J. Comput. Phys. 107*, 1, 1–9.

CHRISTIANSEN, S. H., AND HALVORSEN, T. G. 2011. A Gauge Invariant Discretization on Simplicial Grids of the Schrödinger Eigenvalue Problem in an Electromagnetic Field. *SIAM J. Numer. Anal. 49*, 1, 331–345.

CRANE, K., DE GOES, F., DESBRUN, M., AND SCHRÖDER, P. 2013. Digital Geometry Processing with Discrete Exterior Calculus. In *ACM SIGGRAPH 2013 Courses*, 7:1–7:126.

DESBRUN, M., KANSO, E., AND TONG, Y. 2008. Discrete Differential Forms for Computational Modeling. In *Discrete Differential Geometry*, A. I. Bobenko, P. Schröder, J. M. Sullivan, and G. M. Ziegler, Eds., Vol. 38 of *Oberwolfach Seminars*. Birkhäuser Verlag, 287–324.

GOLAS, A., NARAIN, R., SEWALL, J., KRAJCEVSKI, P., DUBEY, P., AND LIN, M. 2012. Large-Scale Fluid Simulation using Velocity-Vorticity Domain Decomposition. *ACM Trans. Graph. 31*, 6.

GOVERNALE, M., AND UNGARELLI, C. 1998. Gauge Invariant Grid Discretization of the Schrödinger Equation. *Phys. Rev. B 58*, 12, 7816–7821.

HALVORSEN, T. G., AND KVAAL, S. 2012. Manifestly Gauge Invariant Discretizations of the Schrödinger Operator. *Phys. Lett. A 376*, 12–13, 1107–1114.

JIANG, M., MACHIRAJU, R., AND THOMPSON, D. 2005. Detection and Visualization of Vortices. In *The Visualization Handbook*, C. D. Hansen and C. R. Johnson, Eds. Elsevier, 295–309.

KIM, D., YOUNG SONG, O., AND KO, H.-S. 2009. Stretching and Wiggling Liquids. *ACM Trans. Graph. 28*, 5.

KLECKNER, D., AND IRVINE, W. T. M. 2013. Creation and Dynamics of Knotted Vortices. *Nat. Phys. 9*, 4, 253–258.

KNÖPPEL, F., CRANE, K., PINKALL, U., AND SCHRÖDER, P. 2013. Globally Optimal Direction Fields. *ACM Trans. Graph. 32*, 4.

LE, T. B., BORAZJANI, I., KANG, S., AND SOTIROPOULOS, F. 2011. On the Structure of Vortex Rings from Inclined Nozzles. *J. Fluid Mech. 686*, 451–483.

MUSETH, K. 2013. VDB: High-resolution Sparse Volumes with Dynamic Topology. *ACM Trans. Graph. 32*, 3, 27:1–27:22.

PFAFF, T., THUEREY, N., AND GROSS, M. 2012. Lagrangian Vortex Sheets for Animating Fluids. *ACM Trans. Graph. 31*, 4.

SAAD, Y. 2005. ILUT: A Dual Threshold Incomplete LU Factorization. *Num. Lin. Alg. Appl. 1*, 4, 387–402.

SIDE EFFECTS SOFTWARE INC., 2013. Houdini$^{\text{TM}}$ FX.

STATHOPOULOS, A., AND MCCOMBS, J. R. 2010. PRIMME: PReconditioned Iterative MultiMethod Eigensolver: Methods and Software Description. *ACM Trans. Math. Softw. 37*, 2, 21:1–21:30.

STOCK, M. J., DAHM, W. J. A., AND TRYGGVASON, G. 2008. Impact of a Vortex Ring on a Density Interface using a Regularized Inviscid Vortex Sheet Method. *J. Comput. Phys. 227*, 21, 9021–9043.

TROOLIN, D. R., AND LONGMIRE, E. K. 2010. Volumetric Velocity Measurements of Vortex Rings from Inclined Exits. *Exp. Fluids 48*, 3, 409–420.

WEIßMANN, S., AND PINKALL, U. 2010. Filament-based Smoke with Vortex Shedding and Variational Reconnection. *ACM Trans. Graph. 29*, 4.

WIKIPEDIA, 2014. Hamiltonian (quantum mechanics) — Wikipedia, The Free Encyclopedia.

WIKIPEDIA, 2014. Winding Number — Wikipedia, The Free Encyclopedia.

WILSON, K. G. 1974. Confinement of Quarks. *Phys. Rev. D 10*, 8, 2445–2459.

A Quantum Mechanical Analogue

The energy in Eq. (4) has a physical interpretation. It is twice the expectation value of the energy for a quantum mechanical particle with unit mass and unit charge moving in a magnetic field $\mathbf{B} = \operatorname{curl} \mathbf{v}$. According to the standard rules of quantum mechanics [Wikipedia 2014a] this expectation value is given as

$$E_{\mathrm{QM}}(\psi) = \langle\!\langle \psi, H\psi \rangle\!\rangle \tag{6}$$

with the Schrödinger operator

$$H = -\tfrac{\hbar^2}{2} * d^{\nabla} * d^{\nabla} \tag{7}$$

and the 1-form $\hat{\eta} = \langle \frac{\mathbf{v}}{\hbar}, . \rangle$ arising from the vector potential of the magnetic field.

More precisely, using the divergence theorem we obtain

$$E_{\mathrm{QM}}(\psi) = E(\psi) - \frac{\hbar^2}{2} \int_{\partial U} \bar{\psi} * d^{\nabla} \psi.$$

Hence, if we impose either zero Dirichlet boundary conditions $\psi|_{\partial U} = 0$, or zero Neumann boundary conditions $*d^{\nabla}\psi|_{\partial U} = 0$, we obtain equality of $E_{\mathrm{QM}}(\psi)$ and $E(\psi)$. In all our experiments we used zero Neumann boundary conditions.

With either of these boundary conditions our optimization will be solved by the eigenfunction ψ belonging to the smallest eigenvalue of the elliptic self-adjoint operator H

$$H\psi = \lambda\psi \tag{8}$$

Physically, this ψ represents the *ground state* of the charged particle in the given magnetic field.

A.1 Gauge Invariance

Our goal is to find an approximation of the vorticity $\boldsymbol{\omega} = \operatorname{curl} \mathbf{v}$ through vortex filaments, yet we define an energy which depends on $\mathbf{v}$ not $\boldsymbol{\omega}$. Adding the gradient of a function to the velocity field $\tilde{\mathbf{v}} = \mathbf{v} + \operatorname{grad} f$, leaves the curl unchanged $\tilde{\boldsymbol{\omega}} = \boldsymbol{\omega}$. What happens to the minimizer of Eq. (4)? Let us assume that the domain U is simply connected and incorporate $\operatorname{grad} \hat{f} = \operatorname{grad} \frac{f}{\hbar}$ into the definition of $\tilde{d}^{\nabla} = d - i(\hat{\eta} + d\hat{f})$ and apply it to $\tilde{\psi} = e^{i\hat{f}}\psi$

$$\begin{aligned}\tilde{d}^{\nabla}\tilde{\psi} &= i d\hat{f}\,\tilde{\psi} + e^{i\hat{f}} d\psi - i\hat{\eta} e^{i\hat{f}}\psi - i d\hat{f}\,\tilde{\psi} \\ &= e^{i\hat{f}}(d - i\hat{\eta})\psi = e^{i\hat{f}} d^{\nabla}\psi\end{aligned} \tag{9}$$

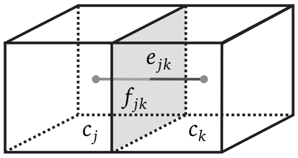
Thus if ψ is a minimizer of Eq. (4) then $\tilde{\psi}$ is a minimizer of the energy derived from $\tilde{\mathbf{v}}$ and achieves the same minimum value. In particular $\tilde{\psi}$ has the same zeros as ψ and so our vortex filaments only depend on $\boldsymbol{\omega}$ not $\mathbf{v}$.

Importance of Gauge Invariance In physics the transition from ψ to $\tilde{\psi}$ (accompanied by replacing $\mathbf{v}$ with $\tilde{\mathbf{v}}$) is called a *gauge transformation* and our vortex filament solutions are gauge invariant. For this reason alone it is important that our numerical solutions are discretely gauge invariant. Lack of computational gauge invariance can also lead to severe numerical errors [Governale and Ungarelli 1998]. In App. B we will see that the discretization we use is gauge invariant.

B Discretization

In this section we give a derivation of a discrete version of Eq. (4) and the corresponding eigenproblem Eq. (8) using the machinery of Discrete Exterior Calculus (DEC) [Desbrun et al. 2008]. For concreteness we assume a topologically regular Cartesian grid, which may be geometrically non-uniform. The final expressions are valid even for simplicial meshes.

Given the definition of our energy (4) we only need the discrete version of d^{∇} acting on discrete 0-forms ψ_j and the inner product on discrete 1-forms, *i.e.*, the (diagonal) Hodge-$*$. Since the definition of d^{∇} involves the input velocity field we first discuss the setup of $\hat{\eta}$.



Velocity Field Input For concreteness we assume a staggered grid representation as input, *i.e.*, fluxes $\varphi_{jk} = \mathbf{v}_{jk}|f_{jk}|$ on the facets of a (topologically) regular 3D Cartesian grid. Here f_{jk} refers to the facet incident on both cell c_j and c_k. Define the discrete 1-form

$$\hat{\eta}_{jk} := \tfrac{1}{\hbar} *_1^{-1} \varphi_{jk} = \tfrac{1}{\hbar}\tfrac{|e_{jk}|}{|f_{jk}|}\varphi_{jk}$$

living on (dual) edges e_{jk} connecting cell centers. The discrete version of the 1-form Hodge star $*_1$ is the ratio of facet area $|f_{jk}|$ to edge length $|e_{jk}|$ (as is standard).

Discrete d^{∇} The discrete version of the differential d^{∇} acting on a discrete 0-form (function) ψ_j is given by

$$d^{\nabla}\psi_{jk} := \bar{r}_{jk}\psi_k - r_{jk}\psi_j, \tag{10}$$

for $r_{jk} = e^{i\hat{\eta}_{jk}/2}$. In words, the values from the end points are rotated to the midpoint of the edge before their difference is taken. To see how this expression comes about consider a function $\phi = e^{i\int\hat{\eta}}$. (Such a function can always be constructed locally.) Clearly $d^{\nabla}\phi = 0$ and it is natural to ask that the discrete differential d^{∇} of the discrete version of ϕ vanish as well. With ϕ_k being different from ϕ_j by a multiplicative factor of $e^{i\hat{\eta}_{jk}}$ the discrete definition of Eq. (10) follows from the requirement that a 1-form change sign when the orientation of the underlying edge is reversed, *i.e.*, $d^{\nabla}\psi_{jk} = -d^{\nabla}\psi_{kj}$.

The Discrete Energy Putting it all together the discrete version of Eq. (4) is a sum over all edges

$$E(\psi) = \tfrac{\hbar^2}{2}\sum_{e_{jk}} \tfrac{|f_{jk}|}{|e_{jk}|}|\bar{r}_{jk}\psi_k - r_{jk}\psi_j|^2. \tag{11}$$

Note that this equation is also correct for tetrahedral meshes with dual edges e_{jk} and primal triangles f_{jk}. Defining the matrix

$$E_{jk} = \begin{cases} -\dfrac{\hbar^2}{2}\dfrac{|f_{jk}|}{|e_{jk}|}\bar{r}_{jk}^2 & k \text{ neighbor of } j, \\ \dfrac{\hbar^2}{2}\displaystyle\sum_{e_{jl}\ni j}\dfrac{|f_{jl}|}{|e_{jl}|} & \text{for } j = k, \\ 0 & \text{otherwise,} \end{cases} \tag{12}$$

the discrete energy is succinctly expressed as

$$E(\psi) = \bar{\psi}_j E_{jk}\psi_k. \tag{13}$$

As we saw earlier we have $E(\psi) = E_{\mathrm{QM}}(\psi)$. In the discrete setting this implies that we also have a discretization of the Schrödinger operator

$$H_{jk} := |c_j|^{-1}E_{jk}.$$

Discrete Optimization The minimizer of the discrete energy (13) under the unit norm constraint $\|\psi\| = 1$ is the eigenvector belonging to the smallest eigenvalue of

$$E_{jk}\psi_k = \lambda|c_j|\psi_j,$$

(which is equivalent to $H_{jk}\psi_k = \lambda\psi_j$, the discrete version of Eq. (8)). Why use the former and not the latter? H_{jk} is not Hermitian with respect to the Euclidean inner product while E_{jk} is. Consequently faster eigensolvers are available for E_{jk}. The appearance of $|c_j|$, *i.e.*, the cell volumes, on the right hand side makes this a generalized eigenproblem. To arrive at a standard eigenproblem. Let $\tilde{E}_{jk} = E_{jk}/\sqrt{|c_j||c_k|}$ and $\psi_j = \tilde{\psi}_j/\sqrt{|c_j|}$ in

$$\tilde{E}_{jk}\tilde{\psi}_k = \lambda\tilde{\psi}_j.$$

In the case of geometrically uniform grids with $|c_j|$ a constant independent of j one may of course omit the factor $|c_j|$ to begin with and similarly ignore the global factor $\frac{\hbar^2}{2}\frac{|f_{jk}|}{|e_{jk}|}$. If furthermore the grid spacing is equal in all dimensions, the simplified matrix E^{δ} (Sec. 3) results.

Convergence and Gauge Invariance In App. A.1 we showed that the zero set of ψ is gauge invariant in the smooth setting. Our approach is discretely gauge invariant, which follows from repeating the calculation of Eq. (9) using the discrete d^{∇} (Eq. (10)) and the fact that $f_k = f_j + df_{jk}$.

The first gauge invariant lattice discretization goes back to Wilson [1974, Eq. 3.5]. For a comparison with standard finite difference methods see [Governale and Ungarelli 1998; Halvorsen and Kvaal 2012]. Convergence to the smooth energy, using an exterior calculus finite element framework, is examined in [Christiansen and Halvorsen 2011], who use the same discretization as we do (Eq. (11)) albeit on a simplicial complex.