Randomized methods to characterize large-scale vortical flow networks

Zhe Bai[iD][1]*, N. Benjamin Erichson[2], Muralikrishnan Gopalakrishnan Meena[3], Kunihiko Taira[3], Steven L. Brunton[4]*

1 Computational Research Division, Lawrence Berkeley National Laboratory, Berkeley, CA, United States of America, 2 Department of Applied Mathematics, University of Washington, Seattle, WA, United States of America, 3 Department of Mechanical and Aerospace Engineering, University of California, Los Angeles, CA, United States of America, 4 Department of Mechanical Engineering, University of Washington, Seattle, WA, United States of America

* zhebai@lbl.gov (ZB); sbrunton@uw.edu (SLB)

Abstract

We demonstrate the effective use of randomized methods for linear algebra to perform network-based analysis of complex vortical flows. Network theoretic approaches can reveal the connectivity structures among a set of vortical elements and analyze their collective dynamics. These approaches have recently been generalized to analyze high-dimensional turbulent flows, for which network computations can become prohibitively expensive. In this work, we propose efficient methods to approximate network quantities, such as the leading eigen-decomposition of the adjacency matrix, using randomized methods. Specifically, we use the Nyström method to approximate the leading eigenvalues and eigenvectors, achieving significant computational savings and reduced memory requirements. The effectiveness of the proposed technique is demonstrated on two high-dimensional flow fields: two-dimensional flow past an airfoil and two-dimensional turbulence. We find that quasi-uniform column sampling outperforms uniform column sampling, while both feature the same computational complexity.

Introduction

Fluid dynamics is a rich and challenging field at the intersection of physics and engineering. There is still a tremendous amount that we do not understand about turbulence, yet working fluids are at the heart of nearly every major industry, including health, defense, transportation, and energy. One cannot overestimate the significance and impact that a better understanding of fluid flows would have in our ability to predict and manipulate their behavior in the real world. The challenge of modeling and controlling fluids stems from the fact that they are non-linear with complex multi-scale interactions over a large range of spatial and temporal scales. Moreover, data from experiments and simulations are generally represented as exceedingly high-dimensional measurements that may obscure underlying patterns. With advances in simulation capabilities and measurement techniques, the volume and quality of such data are rapidly increasing.

Despite the complexity of the governing equations and the overwhelming volume of data, it is often observed that flows evolve on a low-dimensional attractor defined by a few dominant coherent structures [1]. There are a wealth of modal decomposition techniques to mine and characterize these structures from experimental data and numerical solvers [2, 3]. The majority of techniques, such as proper orthogonal decomposition (POD) [1, 4] and dynamic mode decomposition [5–7] are fundamentally linear, as are many of the standard techniques in control theory [8]. Linear control has been widely applied for flow control [9–11], for example to stabilize boundary layers. However, many flows are fundamentally nonlinear, limiting the broad use of linear control theory. Low-order nonlinear models may be obtained by Galerkin projection of the Navier Stokes equations onto POD modes, and these models have had great success for uncovering underling mechanisms that drive flows [4]. Regardless, there are issues with Galerkin models, such as instability and mode deformation with changing parameters and boundary conditions [12–14], limiting the success of POD-Galerkin modeling for turbulence.

Recently, network theoretic approaches have been increasingly leveraged to analyze complex, fluid flow systems [15–21]. Network science [22] characterizes the structure and dynamics on a graph consisting of nodes and the edges connecting them. It is possible to cast a fluid flow in this context, where each grid cell of vorticity is a node, and the induced velocity from the Biot-Savart interaction between each node establishes the edge connections [15]. In this way, individual packets of vorticity interact and evolve according to physics that is encoded in the graph. Network analysis provides a complementary perspective to classical techniques in fluid dynamics, especially for flow control. There have been many powerful advances in network control theory [22, 23] surrounding multi-agent systems [23, 24] in the past two decades. Multi-agent control is designed to be local, efficient to operate, and scalable to extremely large systems, such as the internet [25, 26] or the electric grid [27]. The multi-agent control and underlying network dynamics may also be strongly nonlinear, and these controllers are built to handle time-delays and communication failures, which are typical limiting factors for robust performance in classical control [8]. In particular, networks are often characterized by a large collection of elements, represented by nodes on a graph, that each execute their own set of local protocols in response to external stimulus. This analogy holds quite well for a number of large networked dynamical systems, including animals flocking [28, 29], multi-robotic cooperative control systems [30], sensor networks [31, 32], biological regulatory networks [33, 34], and the internet [25, 26], to name a few. A long-term goal of characterizing vortical networks is the eventual application of multi-agent control to *school* turbulence into a beneficial configuration for an engineering advantage. In contrast, past closed-loop flow control efforts have commonly involved applying linear control techniques to a suitable reduced-order model of the fluid [10].

Motivation

The application of network theory in fluid dynamics faces the challenge of an exceedingly large number of degrees of freedom (nodes) for a fully turbulent flow, leading to an even larger adjacency matrix of edges. Indeed, the adjacency matrix for a vortical flow network scales as the square of the number of fluid grid points, quickly becoming intractable to store and analyze using classical techniques from linear algebra. Recent approaches have been developed to manage this complexity, including building a graph on the *modes* of a flow [35], given by dominant fluid coherent structures from POD, rather than the *nodes*, given by fluid grid elements. However, there is still interest in developing scalable methods to directly analyze the vortical network in the original ambient measurement space, where nodes are grid elements.

Randomized methods for linear algebra provide an emerging alternative to efficiently compute an approximate eigendecomposition of large-scale matrices. These methods work with a reduced representation, a so-called *sketch*, of the input matrix that captures the essential spectral information. This sketch is obtained either through *random sampling* or by random *projections*. The *sketch* can then be used to efficiently compute a desired low-rank approximation, such as the singular value decomposition. For a comprehensive survey and implementation details see [36–40].

Contributions

In this work, we use randomized methods to analyze large networks arising in fluid mechanics. Specifically we are interested in using randomized methods to obtain a qualitatively correct view of the dominant graph structures, which can then be used for the downstream tasks of sensor and actuator placement, optimization, and control. This application domain is the subject of the present work, where we explore the use of random sampling and randomized linear algebra to generate qualitatively accurate estimates of dominant flow structures for cases where the adjacency matrix is exceedingly large. We compare both uniform and quasi-uniform sampling, and demonstrate this approach on two high-dimensional vortical flow networks. A high-level principle sketch is provided in Fig 1.

Vortical interaction network

Recently, network analysis has been employed to represent and analyze vortical interactions in flows [15, 17, 19, 35, 41]. In Nair and Taira [15], spectral sparsification was used to identify a low-dimensional skeleton underlying the high-dimensional, nonlinear fluid dynamics. Taira et al. [17] later demonstrated scale-free behavior of the vortex interaction network for two-dimensional isotropic turbulence. Network analysis has also been used by Meena et al. [19] for community detection and force prediction in an unsteady wake. More recently, interaction networks based on the energy transfer between *modes* has been used for modeling and control [35, 41]. These studies provide compelling evidence that the emerging network perspective may complement well-established flow modeling techniques.

A graph (network) $\mathcal{G}$ consists of three components: a set of vertices (nodes) $\mathcal{V}$, a set of edges $\mathcal{E}$ that connect these vertices, and the associated edge weights $\mathcal{W}$. In the present work, the vortical elements in a flow field represent the nodes of the vortical flow network, following the formulation of Nair and Taira [15]. The interaction among vortical elements can be quantified by the induced velocity. Following the Biot-Savart law, the induced velocity from element i to j in a two-dimensional flow is given by

$$u_{i \to j} = \frac{\Gamma_i}{2\pi d_{ij}} \tag{1}$$

where $\Gamma_i = \omega_i \Delta x \Delta y$ is the strength of vortical element i with vorticity ω_i and grid area $\Delta x \Delta y$, and d_{ij} is the Euclidean distance between the vortical elements. This induced velocity represents the edge weights, and may be used to construct an adjacency matrix

$$\mathbf{A}_{ij} = \begin{cases} \frac{1}{2} \left(|u_{i \to j}| + |u_{j \to i}| \right) & \text{if} \quad i \neq j \\ 0 & \text{if} \quad i = j, \end{cases} \tag{2}$$

which mathematically represents the interactions in the vortical network. Since a vortex cannot induce motion upon itself, there are no diagonal entries for the adjacency matrix. Here we
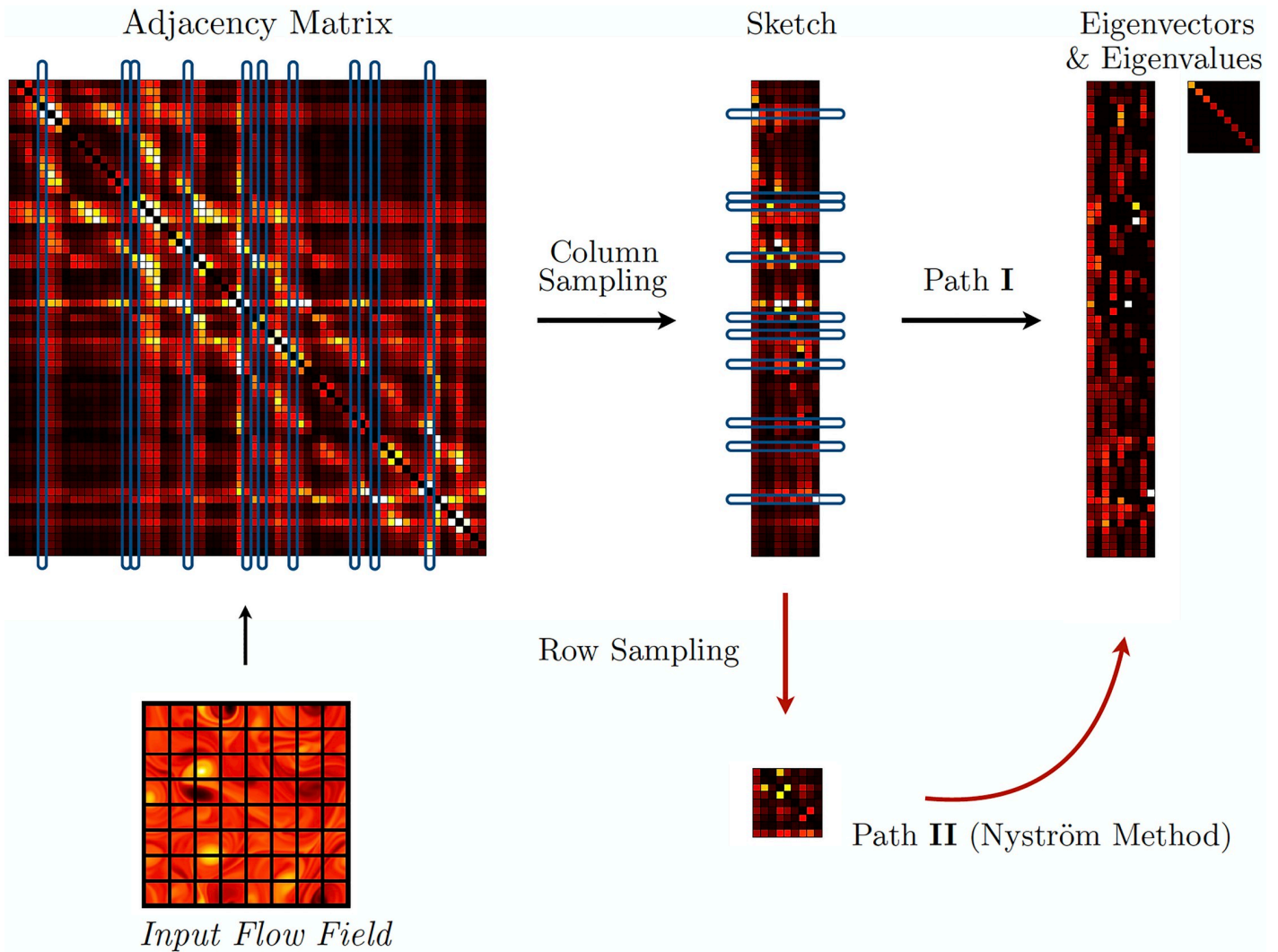
Fig 1. Identification of dominant eigenvalues and eigenvectors of the adjacency matrix obtained from a fluid flow field using randomized methods. Path **I** randomly samples columns of the adjacency matrix to approximate the leading eigenvalues and eigenvectors; path **II** uses both column and row sampling to form a sketch. Note, it is not required to explicitly construct the full adjacency matrix.

have an undirected network with a symmetric adjacency matrix **A** to enable the use of the algorithm described later. However, it is possible to define an asymmetric adjacency matrix for a directed network.

Given the vorticity field of a flow, the corresponding vortical interaction network can be formulated as above. This also suggests that the size of the adjacency matrix scales as $\mathbf{A} \in \mathbb{R}^{n \times n}$ where n is the number of grid points or vortical elements in the flow field. For a high-fidelity simulation or experimental measurements of a flow field, n can exceed $\mathcal{O}(10^5 - 10^6)$, making the computation and analysis of the vortical network computationally expensive. Below, we discuss a few network measures that are particularly useful for understanding the interaction-based characteristics of the flow. The current work focuses on computing these measures for large, dense vortical networks in a computationally tractable manner.

Network measures and computations

The adjacency matrix is the basis for a number of useful network measures, such as eigenvector centrality, Katz centrality, and PageRank [22, 42]. The centrality of a network describes the most important or central elements based on measures that quantify the connection strength of the nodes. One of the most basic centrality measures is the node strength, given by $s_i = \Sigma_j A_{ij}$, which measures the overall interaction strength of a node. The eigenvector centrality is another fundamental network property, which also considers the strength of the neighboring elements of a node and is used to identify the most influential nodes in the network [43]. It is computed as the eigenvector of $\mathbf{A}$ corresponding to the largest eigenvalue λ:

$$\mathbf{Au} = \lambda\mathbf{u}. \tag{3}$$

Entries with large absolute value in the vector $\mathbf{u}$ correspond to influential nodes. The Katz centrality and PageRank are related and also based on computing the dominant eigenvector of the adjacency matrix.

Network analysis is also useful for graph partitioning and clustering [22], where the nodes in the network are grouped into clusters based on their close connectivity. Spectral partitioning or clustering is a common approach based on the dominant eigenvector of the adjacency matrix. Nodes can be divided and sub-divided into groups based on the sign of the corresponding elements in the dominant eigenvectors of $\mathbf{A}$. Spectral partitioning can also be used in community detection algorithms to identify groups of closely interacting nodes, called communities [44]. These algorithms perform a hierarchical grouping of nodes into communities based on maximizing the overall modular measure of the network. This hierarchical grouping can be performed using the dominant eigenvectors of $\mathbf{A}$ [45]. In vortical flows, these network-based measures are useful for identifying the most influential vortical elements and clustering them into coherent structures or communities.

Due to the importance of the spectral information of the adjacency matrix, considerable effort has gone into developing accurate and efficient algorithms to compute dominant eigenvalues for large adjacency matrices. Provided that the largest eigenvalue is real and distinct, the dominant eigenvector can be efficiently computed using the power method [46]. A symmetric adjacency matrix with nonnegative entries will have purely real eigenvalues. For instance, the power method is commonly used to compute the dominant eigenvector of an adjacency matrix [47], and this method is particularly efficient for large sparse adjacency matrices. The computational costs of the power method for computing the dominant eigenvector scales as $\mathcal{O}(n^2)$.

Random sampling-based matrix approximation

As stated above, we are interested in computing the eigendecomposition of a real symmetric matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ that takes the form

$$\mathbf{A} = \mathbf{UDU}^\top, \tag{4}$$

where the columns of the orthogonal matrix $\mathbf{U} \in \mathbb{R}^{n \times n}$ are eigenvectors and the entries of the diagonal matrix $\mathbf{D} \in \mathbb{R}^{n \times n}$ are the corresponding eigenvalues $\lambda_1 \geq \cdots \geq \lambda_n$. More concretely, we are interested in computing only the dominant k eigenvectors and eigenvalues (where $k \ll n$) that yield the rank-k approximation

$$\mathbf{A}_k = \mathbf{U}_k\mathbf{D}_k\mathbf{U}_k^\top, \tag{5}$$

where $\mathbf{U}_k \in \mathbb{R}^{n \times k}$ contains only the k columns of $\mathbf{U}$ that correspond to the k largest eigenvalues.

Here, we use tools from the field of *randomized numerical linear algebra* [36, 38–40, 48–54] to compute such a low-rank approximation efficiently. Indeed, randomized methods are widely used for computing low-rank approximations [55–61].

Scalability is achieved by forming a sketched representation of the input (adjacency) matrix which extracts the inherent spectral information. A sketch $\mathbf{Y} \in \mathbb{R}^{n \times k}$ can be constructed by post-multiplying the adjacency matrix with a an arbitrary 'sketching' matrix $\mathbf{S} \in \mathbb{R}^{n \times k}$. The random matrix $\mathbf{S}$ represents either a random projection or a random sampling process.

- *Projection-based methods* construct a 'sketch' by forming a set of k randomly weighted linear combinations of the columns of the input (adjacency) matrix [39, 48, 49, 62]. This approach can substantially reduce the computational demands when computing a low-rank approximation. However, projection-based methods generally require at least a single pass over the entire data matrix.

- *Sampling-based methods* aim to approximate the low-rank structure of the input (adjacency) matrix from a small random subsample of actual columns or rows of the matrix [63–65]. The sampling process is described in more detail below. In practice, sampling-based methods may bypass the construction of the full adjacency matrix, while sacrificing some accuracy for improved scalability.

In what follows, we generate qualitatively accurate estimates of dominant flow structures for cases where the adjacency matrix is exceedingly large. In this problem context, we require a qualitatively correct view of the dominant graph structures, which can be used for the downstream tasks of sensor and actuator placement, optimization, and control. Therefore, we investigate the use of column sampling to compute a sketched singular value decomposition and the Nyström Method to approximate the leading k eigenvalues and eigenvectors. Both techniques have been successfully used to approximate large-scale matrices in other applications [66]. Further, sampling-based methods provide tunable error bounds that are based on the number of sampled columns/rows, making them viable as an alternative to well-established deterministic algorithms. It is also important to contrast randomized sampling approaches to uniform undersampling, which is the most obvious approach to reduce computations. However, uniform sampling has well-known limitations, including systematic aliasing of high-frequency spatial structures. Randomized methods avoid this aliasing, providing both efficiency and accuracy.

### Sketched singular value decomposition

Column sampling for low-rank matrix approximation dates back to the pioneering work of Frieze et al. [62]. Instead of computing the full eigendecomposition of $\mathbf{A}$, it is possible to form a rank-k approximation by first sampling $l \geq k$ columns from the input matrix

$$\mathbf{C} = \mathbf{AS} \tag{6}$$

where the matrix $\mathbf{C} \in \mathbb{R}^{n \times l}$ consists of a subset of l columns of $\mathbf{A}$ and $\mathbf{S}$ describes the corresponding random sampling process. In practice, it is only necessary to form a list J comprised of l column indices

$$\mathbf{C} := \mathbf{A}(:, J). \tag{7}$$

There are several sampling strategies to choose *good* columns, and the approximation error depends on the both the sampling process and the number of columns sampled.
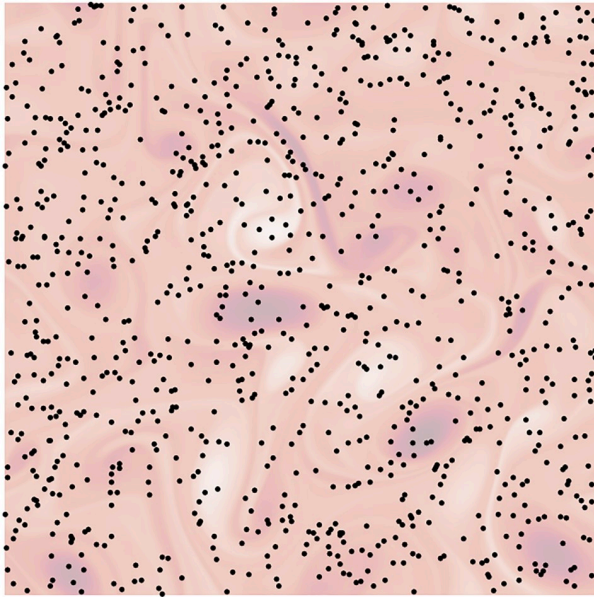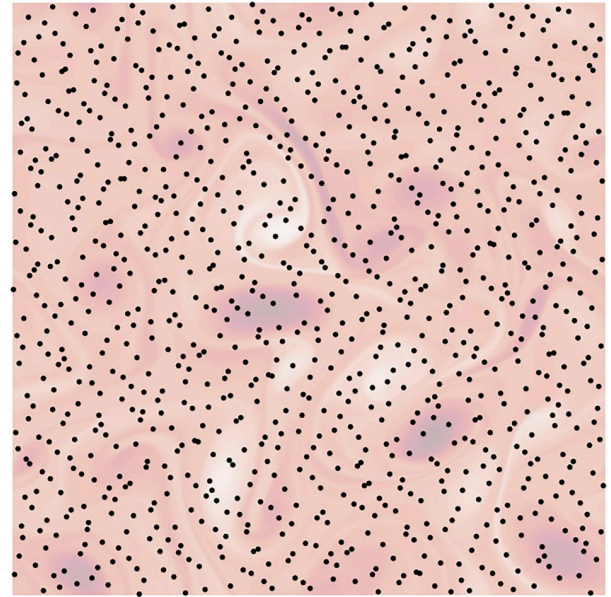
(a) Uniform random sampling.   **(b)** Halton random sampling.

Fig 2. Examples of two different random sampling approaches based on uniform sampling and Halton random sampling. Halton sampling provides a better coverage of the domain than uniform sampling does.

In our problem setup, each column of **A** corresponds to the network interaction of a single grid element in the original fluid flow field with every other grid element. Here, we compare uniform random sampling against (quasi-random) Halton sampling [67, 68]. Halton sampling provides better coverage of the domain, as illustrated in Fig 2, and our empirical results show that this translates into faster convergence of the approximation error with respect to the number of columns sampled, as discussed in the next section.

If the columns of **C** are carefully chosen, then **C** provides a basis for the column space of the adjacency matrix **A**. Note that here we assume that **A** is a symmetric matrix. Thus, the dominant k left singular vectors of $\mathbf{C} = \mathbf{U}_\mathbf{C}\boldsymbol{\Sigma}_\mathbf{C}\mathbf{V}_\mathbf{C}^*$, provide an approximation for the dominant k eigenvectors of **A**, i.e., we have $\tilde{\mathbf{U}}_k \approx \mathbf{U}_\mathbf{C}[:, 1:k]$. The corresponding eigenvalues are approximated by scaling the singular values of **C**:

$$\tilde{\mathbf{D}}_k = \sqrt{\frac{n}{l}}\boldsymbol{\Sigma}_\mathbf{C}[1:k, 1:k]. \tag{8}$$

The cost of constructing the sketch matrix using uniform or Halton sampling is $\mathcal{O}(nl)$. Then, the cost of computing the SVD on **C** is $\mathcal{O}(nl^2)$. For parallel computations, it is possible to compute $\mathbf{U}_\mathbf{C}$ and $\boldsymbol{\Sigma}_\mathbf{C}$ by taking the SVD of $\mathbf{C}^T\mathbf{C}$, which requires $\mathcal{O}(nl^2)$ to be generated and $\mathcal{O}(l^3)$ for the corresponding SVD.

Because the SVD is the computational engine for snapshot POD, randomized techniques have been used to accelerate this computation as well.

The Nyström method

The Nyström method provides an efficient approach for low-rank matrix approximations in large-scale learning applications. It was initially introduced as a quadrature method for

numerical integration to approximate eigenfunction solutions. Recent work has streamlined these computations and provided theoretical foundations for the use of various sampling schemes [69–72].

In addition to column sampling, the Nyström method further constructs a square matrix $\mathbf{W} \in \mathbb{R}^{l \times l}$ by selecting the J rows and columns of $\mathbf{A}$:

$$\mathbf{W} := \mathbf{C}(J, :) = \mathbf{A}(J, J). \tag{9}$$

When $\mathbf{A}$ is symmetric and positive-semidefinite (SPSD), $\mathbf{W}$ is also SPSD. The small matrix $\mathbf{W}$ can be used to efficiently compute the dominant eigenvectors and eigenvalues of $\mathbf{A}$. Following [73], we first compute the eigendecomposition:

$$\mathbf{W} = \mathbf{U_W} \mathbf{D_W} \mathbf{U_W}^\top. \tag{10}$$

Then, we reconstruct the dominant eigenvalues as

$$\tilde{\mathbf{D}}_k = \frac{n}{l} \mathbf{D_W}, \tag{11}$$

and the corresponding eigenvectors

$$\tilde{\mathbf{U}}_k = \sqrt{\frac{l}{n}} \mathbf{C} \mathbf{U_W} \mathbf{D_W}^\dagger. \tag{12}$$

The Nyström method has a computational complexity of $\mathcal{O}(l^3)$.

Results

We now demonstrate the use of sampling-based randomized decomposition techniques to characterize the vortical interaction networks for two example flows: the laminar wake flow past a NACA 0012 airfoil with a Gurney flap attached to the trailing edge [74], and a two-dimensional homogeneous decaying isotropic turbulence [17].

Numerical simulations

The first example is the flow past a NACA 0012 airfoil with a chord-based Reynolds number of $Re = 1,000$, at an angle of attack of 20°, and with a Gurney flap of 10% chord length (c). This setup generates an unsteady 2P wake [74, 75], with periodic shedding of two pairs of positive and negative vortices. This flow is solved using direct numerical simulations (DNS) via the immersed boundary projection method [76, 77], following the computational setup of Gopalakrishnan Meena *et al.* [74]. The computational domain consists of five nested levels of multi-domains with the finest level of size $(x/c, y/c) \in [−1, 1] \times [−1, 1]$ and the largest being $(x/c, y/c) \in [−16, 16] \times [−16, 16]$ in size. All the domains have a grid resolution of $m_x \times m_y = 360 \times 360$. The time step is limited to a maximum Courant-Friedrichs-Lewy (CFL) number of 0.3. Fig 3(a) shows the vorticity field in a sub-domain with a grid resolution of 1261×1261 and size 7×7, nondimensionalized by the chord length of the airfoil.

The second example is two-dimensional decaying homogeneous isotropic turbulence, which is a canonical high-dimensional, mutiscale turbulent flow that exhibits complex nonlinear interaction of vortices over a wide range of length scales [78]. The vorticity field is obtained from a two-dimensional incompressible DNS, solving the vorticity transport equation without forcing [17]. The simulation, based on the Fourier spectral method and the fourth-order Runge-Kutta time integration scheme, is performed on a square biperiodic computational domain with grid resolution of $m_x \times m_y = 1024 \times 1024$. The vorticity field is initialized with 100 vortices with random strengths, core sizes, and centers such that the kinetic energy spectra
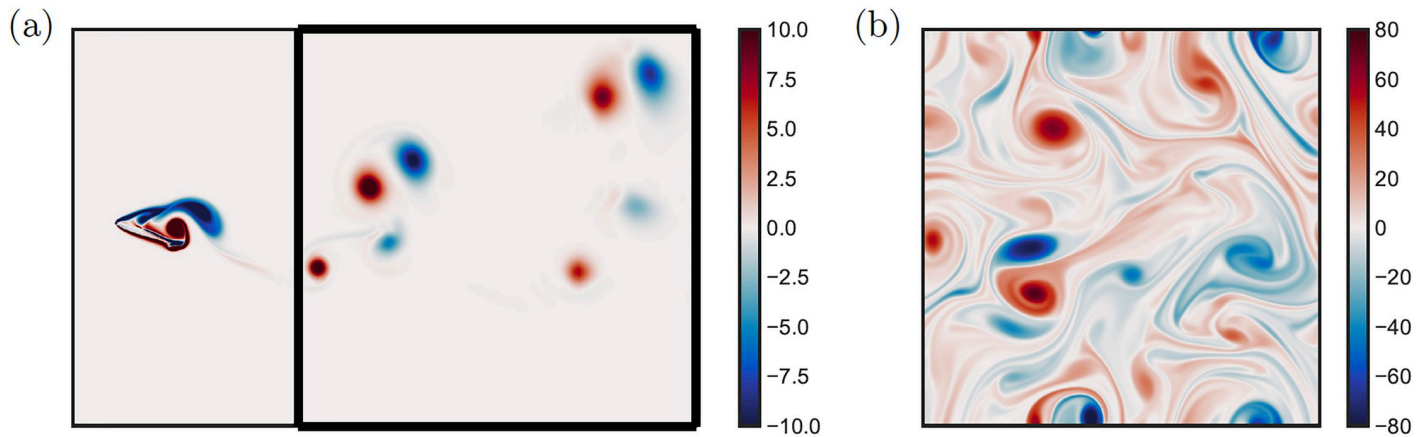
Fig 3. Example flow fields. (a) Vorticity field of two-dimensional DNS of the flow over a NACA 0012 airfoil with Gurney flap at an angle of attack of 20° and flap height of 0.1 chord length at Re = 1000; the full illustration can be found in [74]. A subdomain of the original vorticity field is used in this example. (b) Vorticity field of two-dimensional decaying homogeneous isotropic turbulence, initialized by an integral length-scale based Reynolds number of $Re(t_0)$ = 814.

https://doi.org/10.1371/journal.pone.0225265.g003

satisfies $E(k) \propto k \exp(-k^2/k_0^2)$ with $k_0 = 26.5$ [79]. For this study, the flow field is initialized by an integral length-scale based Reynolds number of $Re(t_0) = 814$, shown in Fig 3(b).

The vorticity snapshots of both flows are used to construct the adjacency matrices of the corresponding vortical networks using Eq 2. The size of the adjacency matrices are of the order $\mathcal{O}(10^8)$ and $\mathcal{O}(10^{12})$ for the airfoil and turbulent flow, respectively, emphasizing the need for dimensionality reduction and computationally tractable analysis tools.
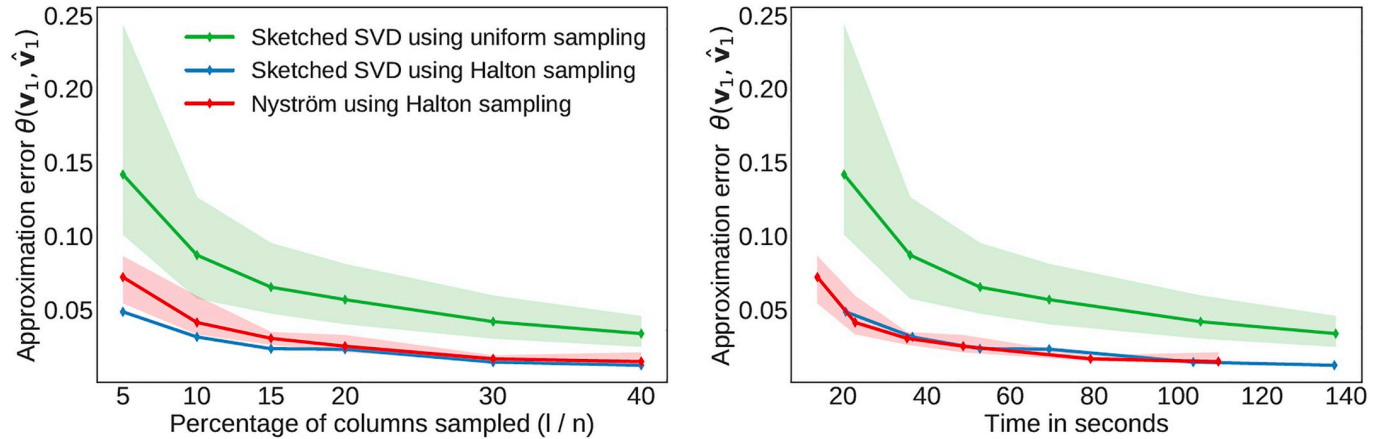
Computational performance

A naive application of network theory in fluid dynamics can be challenging, because the adjacency matrix is often too large to be stored or analyzed using classical techniques from linear algebra. There are two primary challenges associated with large vortical networks:

- The size of the adjacency matrix, which describes the network edges, scales as the square of the number of fluid grid points;

- Unlike social graphs, the vortical adjacency matrix is dense.

Here, we use randomized methods to accelerate computations based on the vortical adjacency matrix. We sample a small number of columns from the adjacency matrix to compute a sketched SVD or Nyström approximation. Thus, the approximation quality can be controlled via the number of samples used to compute the low-rank approximation. In the context of networks arising in fluid mechanics, it turns out that sampling a small number of columns is sufficient for computing an approximation that is qualitatively useful. This leads to tremendous memory and computational savings, since the costs are proportional to the number of columns that we sample. We run all of our experiments using Amazon AWS, working on a memory optimized instance 'x1.16xlarge' that is powered by two Intel Xeon E7 8880 v3 (Haswell) processors with 64 virtual cores and 976 GB fast memory. Our algorithms are implemented in Python powered by MKL accelerated linear algebra routines, and C extensions for constructing the elements of the adjacency matrix.

We show in Fig 4 the approximation error as a function of the percentage of columns sampled and the corresponding computational time required.

Here, we consider two cropped spatial grids of different dimensions from which we construct the adjacency matrix. The error is reported as the acute angle between the true leading

(a) Results for an $n \times n$ adjacency matrix with $n = 316^2$.
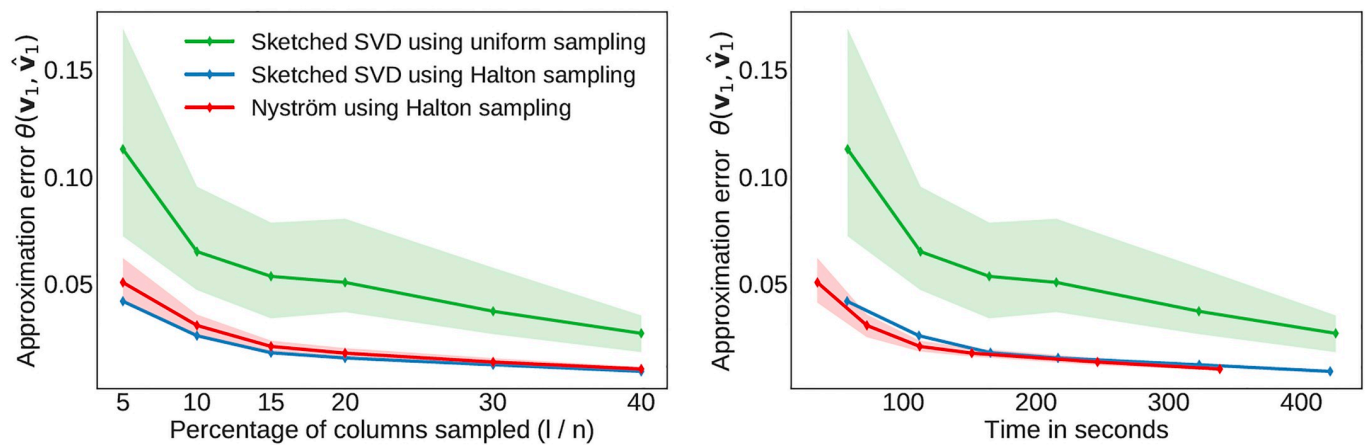
(b) Results for an $n \times n$ adjacency matrix with $n = 421^2$.

Fig 4. Computational performance for the flow over an airfoil using two different spatial resolutions. The results are averaged over 20 runs using different random seeds. The shaded region indicates the variance of approximation error in multiple runs.

eigenvector **v** and the approximate eigenvector $\hat{\mathbf{v}}$, which provides a useful summary of how closely the two high-dimensional eigenvectors align. Since the algorithms are fundamentally based on random sampling, we average the results over 20 initializations and report the distribution of errors.

Halton sampling results in considerably lower error than uniform sampling at the same percentage of columns sampled, which is consistent with the observation that it is sampling the flow domain more efficiently. The results based on Halton sampling also have considerably lower variance, indicating improved numerical robustness. Fig 4a and 4b illustrate how substantially the computational time increases when the spatial resolution of the flow field is increased only slightly. In the 421×421 case, only about 5% of the columns need to be sampled for 5% error compared between the approximate and true eigenvectors. In other words, we use only about 5% of the information to compute the low-rank approximation, and this reduces the memory requirements for the example in Fig 4b from about 215GB to roughly 10.5GB. At the same time, the computational time is reduced since it is not necessary to construct the full adjacency matrix. The computational time is reduced further by using the Nyström method, while only sacrificing a small amount of accuracy. For comparison, Table 1

Table 1. Summary of computational results for constructing the full adjacency matrix and computing the dominant eigenvector using the deterministic power method. Here the computational bottleneck is the memory required to construct the adjacency matrix. Note that the power method does not generally require that the full adjacency matrix is constructed explicitly; however, the power method is not efficient because the adjacency matrices we consider are dense.

	Dimensions of adjacency matrix	Time to construct full adjacency matrix	Storage required	Time to compute deterministic eigenvector
airfoil	$316^2 \times 316^2$	197.76 (sec)	74.29 (GB)	60.84 (sec)
airfoil	$421^2 \times 421^2$	715.78 (sec)	214.67 (GB)	197.87 (sec)
turbulent	$256^2 \times 256^2$	158.44 (sec)	32.00 (GB)	27.51 (sec)
turbulent	$342^2 \times 342^2$	272.09 (sec)	101.92 (GB)	85.01 (sec)

https://doi.org/10.1371/journal.pone.0225265.t001

summarizes the computational demands and performance for the deterministic power iteration method. Fig 5 shows similar results for the turbulent flow. Again, we see that Halton sampling is favorable and that only a small fraction of columns is required to yield a good approximation of the leading eigenvector.
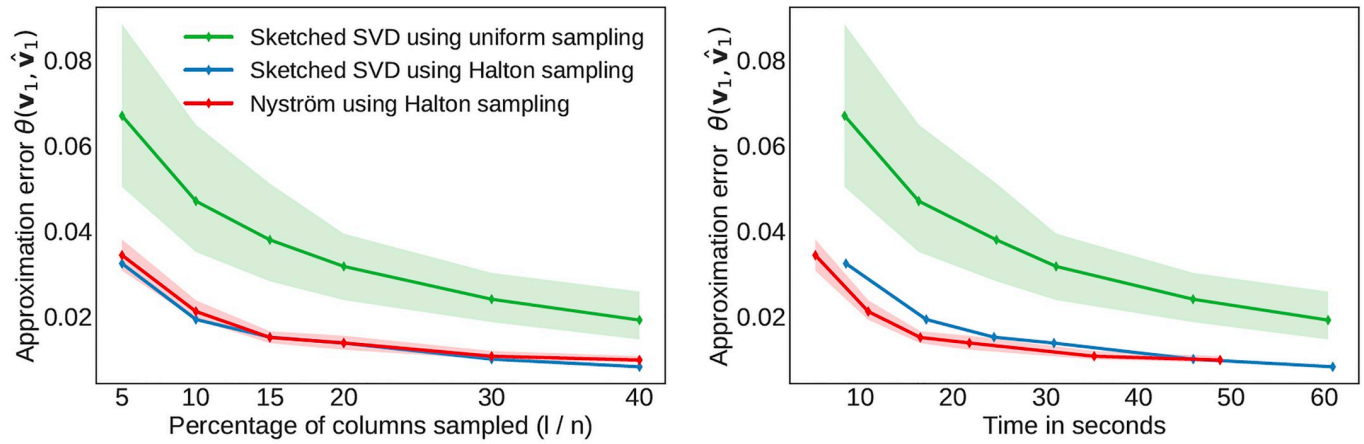
(a) For an $n \times n$ adjacency matrix with $n = 256^2$.
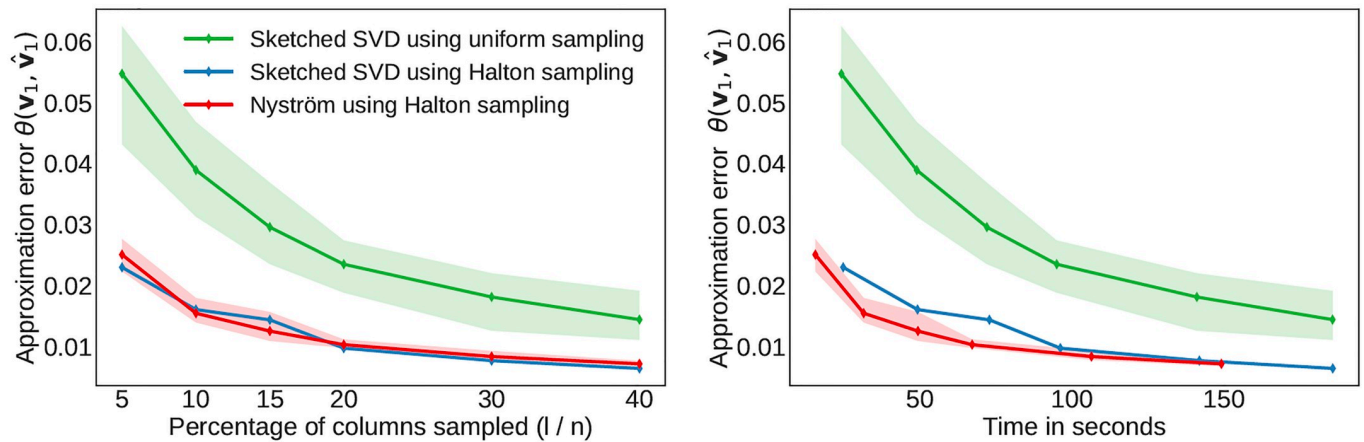
(b) For an $n \times n$ adjacency matrix with $n = 342^2$.

Fig 5. Computational performance for the isotropic turbulent flow using two different spatial resolutions. The results are averaged over 20 runs using different random seeds. The shaded region indicates the variance of approximation error in multiple runs.

https://doi.org/10.1371/journal.pone.0225265.g005

(a) Snapshots　　**(b)** Dominant eigenvectors　　**(c)** Approximate eigenvectors
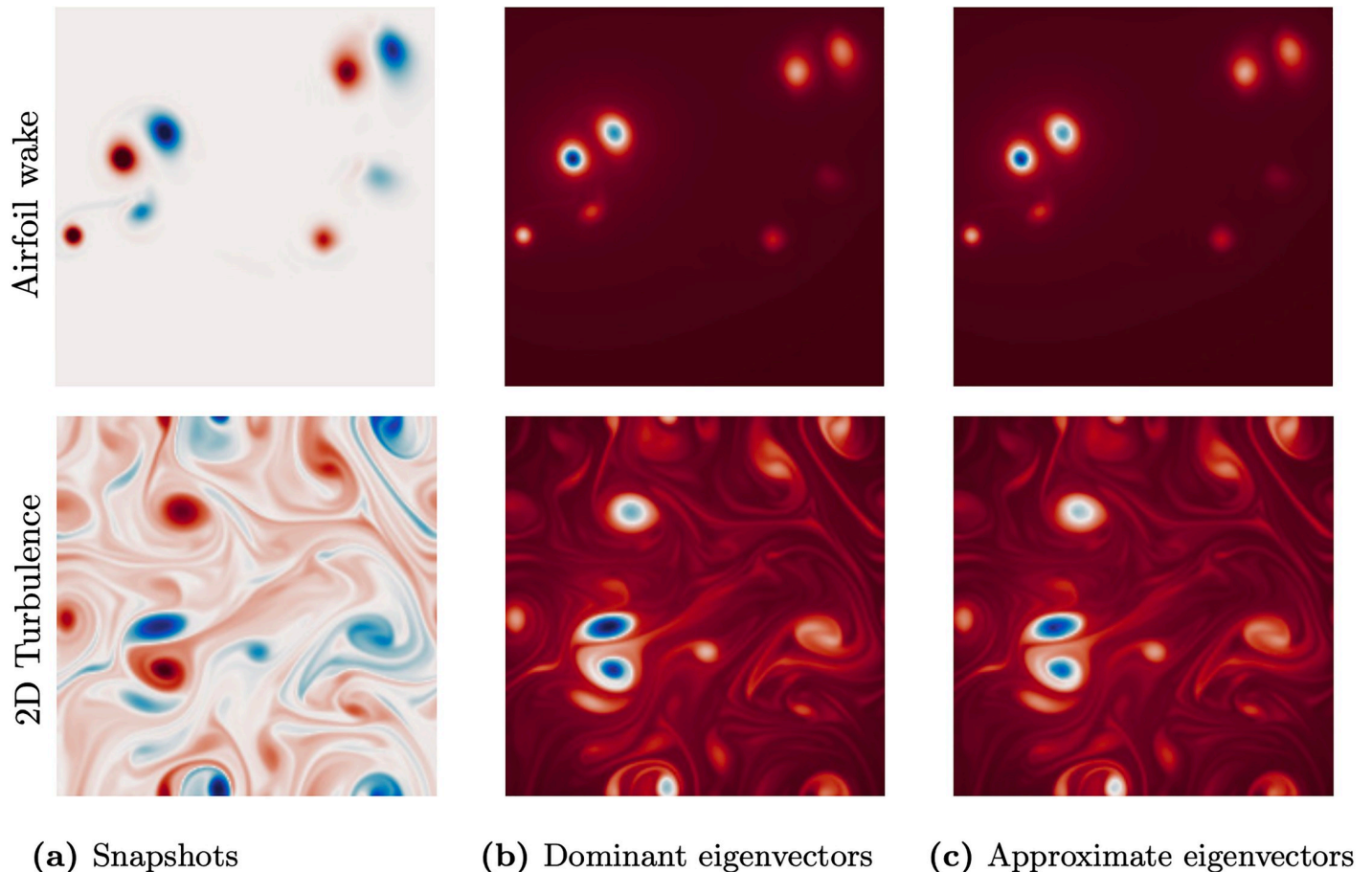
Fig 6. Qualitative comparison of the deterministic (b) and approximate (c) dominant eigenvector of the adjacency matrix generated from the flow field in (a).
The top row shows the results for the airfoil wake and the bottom shows the results for the two-dimensional isotropic turbulence. The approximation in (c) uses the Nyström method with Halton sampling with 10% of the columns of the adjacency matrix.

https://doi.org/10.1371/journal.pone.0225265.g006

Approximate eigenvectors and spectral clustering

Fig 6 shows the leading eigenvectors of the adjacency matrix for the two flow examples, computed using deterministic and randomized methods. By visual inspection, the Nyström method with quasi-random Halton sampling provides an excellent approximation of the leading eigenvector using only 10% of the columns of the adjacency matrix. The resulting eigenvector centrality of the matrix **A** highlights strong vortex cores in both flows, as shown in Fig 6(b). Vortices with lower strength have less centrality, emphasizing their decreased role in influencing the flow field. The 2P wake structure is clearly visible in the flow past an airfoil, and some shear layer structures are present in the 2D turbulent flow. The irrotational regions in both flows have the lowest centrality measure. These observations complement the traditional fluid dynamics literature regarding highly influential structures in a vortical flow. The approximate eigenvector centrality measures capture both the highly influential vortex cores as well as the smaller, less influential vortices, as seen in both flows. Moreover, the network-based measures take into account the spatial arrangements of the vortices to assess their influence. The error between the approximate and true eigenvectors are shown in Fig 7.
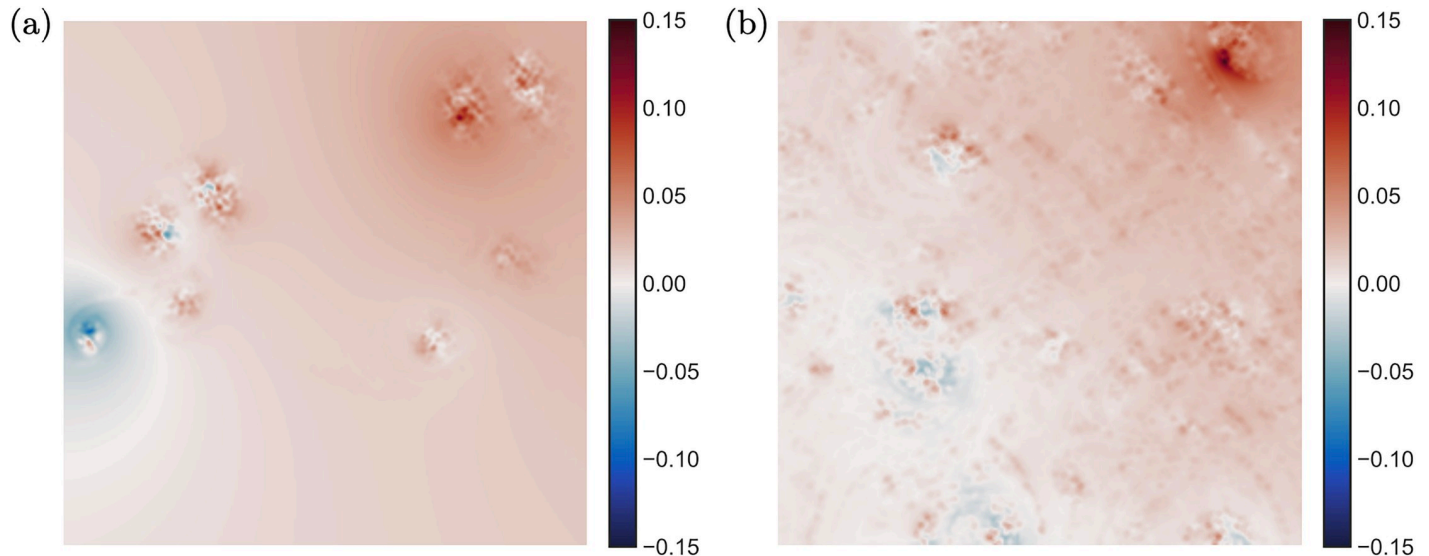
Fig 7. Relative approximation error between the deterministic and approximate dominant eigenvector as in Fig 6, distributed in the spatial domain: (a) airfoil wake flow, (b) two-dimensional isotropic turbulent flow.

In the computational comparison above, we only considered relatively low resolution flow fields because it was computationally prohibitive to compute the ground truth eigenvectors using deterministic methods.

However, with randomized techniques it is possible to compute the leading eigenvector of the adjacency matrix for a 512×512 resolution grid and a 1024×1024 resolution grid, respectively, as shown in Fig 8. The adjacency matrix for the 1024×1024 grid would require about 8 TB of storage, if explicitly constructed. Instead, we sample only about 10% of the columns to compute the Nyström approximation. While we cannot quantify the approximation error, we can see by visual inspection that the eigenvector captures dominant flow structures.

Finally, we investigate spectral clustering, which is one of the common uses of the leading eigenvectors of the adjacency matrix, providing a principled approach to cluster nodes into common communities. Fig 9 shows the results of spectral clustering based on the three leading eigenvectors obtained from the adjacency matrix. In both flow fields, the approximate clusters obtained using randomized methods closely resemble the clusters computed via deterministic eigendecomposition. The right panels in Fig 9 further shows the data plotted in these first three eigenvector coordinates to visualize the clusters and subspaces. Typically, these leading eigenvectors are quite useful for determining relevant community structures within a network. These communities have an important physical interpretation in vortical flow networks, hierarchically identifying and organizing vortex cores within the flow. Because these communities are often determined from the few dominant eigenvectors, the randomized approach here can provide considerable computational advantages. Further, since high-vorticity nodes have a large influence on the network interactions, it may be possible to improve convergence by preferential sampling using these community structures.

Discussion

In this work, we have demonstrated the effective use of scalable algorithms from randomized linear algebra to accelerate network computations for large-scale fluid flows, as demonstrated on the wake behind an airfoil and two-dimensional isotropic turbulent flow. In particular, we
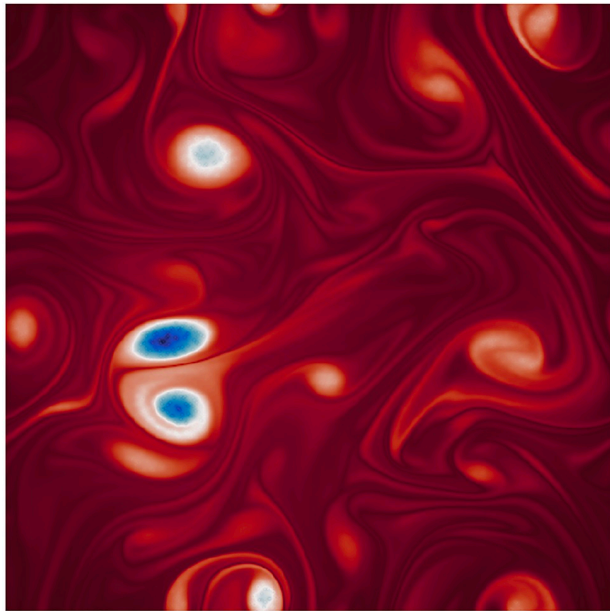
(a) Eigenvector of dimension 512^2.
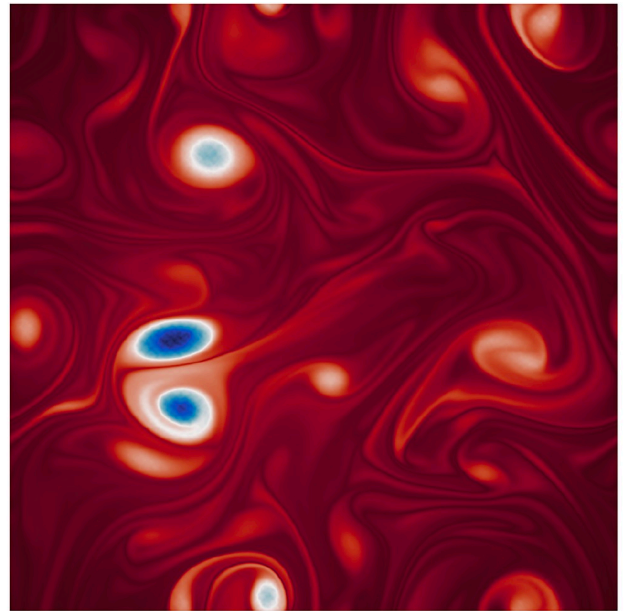

**(b)** Eigenvector of dimension 1024^2.

Fig 8. Approximated leading eigenvectors of the adjacency matrices for higher-resolution isotropic flow fields. Here, we are using the Nyström method with Halton sampling. By visual inspection we can see that sampling only about 10% of the columns of the full adjacency matrix is sufficient for computing an approximated eigenvector that captures the dominant graph structure.
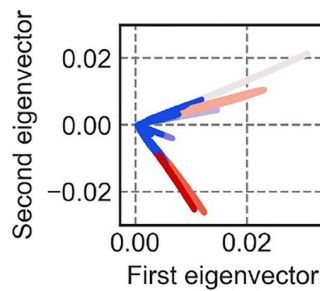
have used sampling-based methods to approximate the leading eigenvectors of the adjacency matrix of the vortical interaction network for cases where the data is so large that even single-pass algorithms are prohibitively expensive. Combining importance sampling, based on the probability distribution of detected communities, and the Nyström method, the leading eigenvector can be accurately computed at a fraction of the computational cost and memory requirement, bypassing the need to construct or query the full **A** matrix. We also showed that quasi-random Halton sampling techniques outperform uniform sampling in both examples as they provide more comprehensive sampling coverage of the spatial domain.
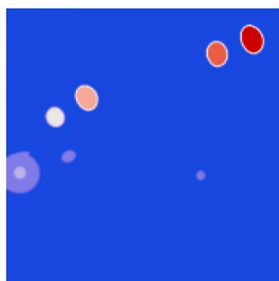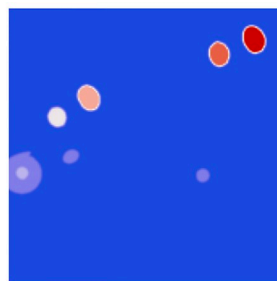
(a) Exact clusters.   **(b)** Approximation.   **(c)** Scores (1 vs. 2). **(d)** Scores (1 vs. 3).
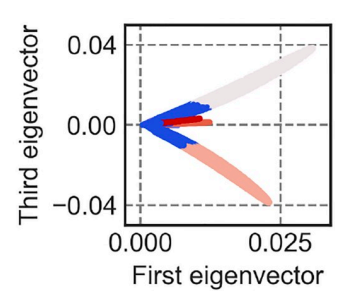
Fig 9. Spectral clustering using the top three eigenvectors of the adjacency matrix for the airfoil wake flow. In (a) we show the clusters that we found using the exact eigenvectors. It can be seen that the approximate eigenvectors allow us to reveal the same seven clusters, as shown in (b). In (c) and (d) we show the first three approximated eigenvector coordinates to visualize the clusters and subspaces.

There are a number of interesting future directions based on this work. Further study is required to apply these randomized network based analysis techniques to more complex three-dimensional turbulent flows. There are a two main aspects to consider: the high-dimensionality of 3D flows, and the broadband spectrum of turbulent flow fields. The high-dimensionality of 3D flows does not present a challenge for randomized methods, and in fact, would present an even bigger opportunity for computational savings. However, if the increase of dimensionality is accompanied by an increase in the number of energetic modes in the SVD, then this may present a challenge for randomized methods, which rely on the existence of low-rank structure and a gap between these singular values and those characterized by turbulent fluctuations. However, even in fully turbulent flows with broadband frequency content, there often exist dominant coherent structures, and it is likely that these structures will still be identified with randomized algorithms.

In addition, there are similar scaling issues in the related fields of almost invariant sets and set-oriented methods [80–84], where the state-transition operator may be viewed as a graph on the high-dimensional state-space. In the transfer operator case, machine learning has been used to identify a data-driven discretization of the state-space into clusters, which serve as nodes that scale with the intrinsic rank of the attractor, rather than the high-dimensional measurement dimension [85]; similar clustering has been used to identify subspaces for POD–Galerkin models [86]. These methods have been used to determine eigenvectors of the Perron–Frobenius operator, the dual of the Koopman operator, establishing a strong connection between sensitivity and coherence in fluid flows. It will be interesting to mathematically connect those approaches with the randomized methods considered in the present study. Further, since the adjacency matrix depends on the instantaneous flow field, it will be interesting to extend the structures derived in this analysis to time-varying flows. Previously, this would have been prohibitively expensive, but the randomized methods presented here may make this more tractable. There is also an opportunity to incorporate regularization in the Biot-Savart computations, as vorticity elements that are very close may cause numerical instability.

Supporting information

S1 File. Data set of the example flow fields used in the manuscript, as shown in Fig 3. (ZIP)

Acknowledgments

Author Contributions

Conceptualization: Zhe Bai, Steven L. Brunton.

Data curation: Muralikrishnan Gopalakrishnan Meena.

Formal analysis: Zhe Bai.

Funding acquisition: Kunihiko Taira, Steven L. Brunton.

Investigation: Zhe Bai, N. Benjamin Erichson.

Methodology: Zhe Bai, N. Benjamin Erichson.

Project administration: Kunihiko Taira, Steven L. Brunton.

Supervision: Steven L. Brunton.

Visualization: Zhe Bai, N. Benjamin Erichson.

Writing – original draft: Zhe Bai.

Writing – review & editing: Zhe Bai, N. Benjamin Erichson, Muralikrishnan Gopalakrishnan Meena, Kunihiko Taira, Steven L. Brunton.

References

1. Holmes PJ, Lumley JL, Berkooz G, Rowley CW. Turbulence, coherent structures, dynamical systems and symmetry. 2nd ed. Cambridge Monographs in Mechanics. Cambridge, England: Cambridge University Press; 2012.

2. Taira K, Brunton SL, Dawson S, Rowley CW, Colonius T, McKeon BJ, et al. Modal analysis of fluid flows: An overview. AIAA Journal. 2017; 55(12):4013–4041. https://doi.org/10.2514/1.J056060

3. Taira K, Hemati MS, Brunton SL, Sun Y, Duraisamy K, Bagheri S, et al. Modal Analysis of Fluid Flows: Applications and Outlook. accepted, AIAA Journal. 2019. https://doi.org/10.2514/1.J058462

4. Noack BR, Afanasiev K, Morzynski M, Tadmor G, Thiele F. A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. Journal of Fluid Mechanics. 2003; 497:335–363. https://doi.org/10.1017/S0022112003006694

5. Schmid PJ. Dynamic Mode Decomposition of Numerical and Experimental Data. Journal of Fluid Mechanics. 2010; 656:5–28. https://doi.org/10.1017/S0022112010001217

6. Rowley CW, Mezić I, Bagheri S, Schlatter P, Henningson DS. Spectral analysis of nonlinear flows. Journal of Fluid Mechanics. 2009; 645:115–127. https://doi.org/10.1017/S0022112009992059

7. Kutz JN, Brunton SL, Brunton BW, Proctor JL. Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems. SIAM; 2016.

8. Dullerud GE, Paganini F. A course in robust control theory: A convex approach. Texts in Applied Mathematics. Berlin, Heidelberg: Springer; 2000.

9. Bagheri S, Hoepffner J, Schmid PJ, Henningson DS. Input-output analysis and control design applied to a linear model of spatially developing flows. Applied Mechanics Reviews. 2009; 62(2):020803–1– 020803–27. https://doi.org/10.1115/1.3077635

10. Brunton SL, Noack BR. Closed-loop turbulence control: Progress and challenges. Applied Mechanics Reviews. 2015; 67:050801–1–050801–48. https://doi.org/10.1115/1.4031175

11. Sipp D, Schmid PJ. Linear Closed-Loop Control of Fluid Instabilities and Noise-Induced Perturbations: A Review of Approaches and Tools. Applied Mechanics Reviews. 2016; 68(2):020801. https://doi.org/10.1115/1.4033345

12. Carlberg K, Barone M, Antil H. Galerkin v. least-squares Petrov–Galerkin projection in nonlinear model reduction. Journal of Computational Physics. 2017; 330:693–734. https://doi.org/10.1016/j.jcp.2016.10.033

13. Loiseau JC, Brunton SL. Constrained Sparse Galerkin Regression. Journal of Fluid Mechanics. 2018; 838:42–67. https://doi.org/10.1017/jfm.2017.823

14. Loiseau JC, Noack BR, Brunton SL. Sparse reduced-order modeling: sensor-based dynamics to full-state estimation. Journal of Fluid Mechanics. 2018; 844:459–490. https://doi.org/10.1017/jfm.2018.147

15. Nair AG, Taira K. Network-theoretic approach to sparsified discrete vortex dynamics. Journal of Fluid Mechanics. 2015; 768:549–571. https://doi.org/10.1017/jfm.2015.97

16. Murugesan M, Sujith R. Combustion noise is scale-free: transition from scale-free to order at the onset of thermoacoustic instability. Journal of Fluid Mechanics. 2015; 772:225–245. https://doi.org/10.1017/jfm.2015.215

17. Taira K, Nair AG, Brunton SL. Network structure of two-dimensional decaying isotropic turbulence. Journal of Fluid Mechanics. 2016; 795. https://doi.org/10.1017/jfm.2016.235

18. Schlueter-Kuck KL, Dabiri JO. Coherent structure colouring: identification of coherent structures from sparse data using graph theory. Journal of Fluid Mechanics. 2017; 811:468–486. https://doi.org/10.1017/jfm.2016.755

19. Gopalakrishnan Meena M, Nair AG, Taira K. Network community-based model reduction for vortical flows. Physical Review E. 2018; 97(6):063103. https://doi.org/10.1103/PhysRevE.97.063103 PMID: 30011542

20. Murayama S, Kinugawa H, Tokuda IT, Gotoda H. Characterization and detection of thermoacoustic combustion oscillations based on statistical complexity and complex-network theory. Physical Review E. 2018; 97(2):022223. https://doi.org/10.1103/PhysRevE.97.022223 PMID: 29548163

21. Iacobello G, Scarsoglio S, Kuerten J, Ridolfi L. Lagrangian network analysis of turbulent mixing. Journal of Fluid Mechanics. 2019; 865:546–562. https://doi.org/10.1017/jfm.2019.79

22. Newman MEJ. Networks: an introduction. Oxford Univ. Press; 2010.

23. Mesbahi M, Egerstedt M. Graph theoretic methods in multiagent networks. Princeton University Press; 2010.

24. Rahmani A, Ji M, Mesbahi M, Egerstedt M. Controllability of multi-agent systems from a graph-theoretic perspective. SIAM J Control and Optimization. 2009; 48(1):162–186. https://doi.org/10.1137/060674909

25. Low SH, Paganini F, Doyle JC. Internet congestion control. Control Systems, IEEE. 2002; 22(1):28–43. https://doi.org/10.1109/37.980245

26. Doyle JC, Alderson DL, Li L, Low S, Roughan M, Shalunov S, et al. The "robust yet fragile" nature of the Internet. Proceedings of the National Academy of Sciences of the United States of America. 2005; 102(41):14497–14502. https://doi.org/10.1073/pnas.0501426102

27. Susuki Y, Mezić I, Hikihara T. Coherent swing instability of power grids. Journal of nonlinear science. 2011; 21(3):403–439. https://doi.org/10.1007/s00332-010-9087-5

28. Leonard NE, Fiorelli E. Virtual leaders, artificial potentials and coordinated control of groups. In: Decision and Control, 2001. Proceedings of the 40th IEEE Conference on. vol. 3. IEEE; 2001. p. 2968–2973.

29. Olfati-Saber R. Flocking for multi-agent dynamic systems: Algorithms and theory. Automatic Control, IEEE Transactions on. 2006; 51(3):401–420. https://doi.org/10.1109/TAC.2005.864190

30. Balch T, Arkin RC. Behavior-based formation control for multirobot teams. Robotics and Automation, IEEE Transactions on. 1998; 14(6):926–939. https://doi.org/10.1109/70.736776

31. Cortes J, Martinez S, Karatas T, Bullo F. Coverage control for mobile sensing networks. In: IEEE International Conference on Robotics and Automation (ICRA). vol. 2. IEEE; 2002. p. 1327–1332.

32. Leonard NE, Paley DA, Lekien F, Sepulchre R, Fratantoni DM, Davis RE. Collective motion, sensor networks, and ocean sampling. Proceedings of the IEEE. 2007; 95(1):48–74. https://doi.org/10.1109/JPROC.2006.887295

33. Milo R, Shen-Orr S, Itzkovitz S, Kashtan N, Chklovskii D, Alon U. Network motifs: simple building blocks of complex networks. Science. 2002; 298(5594):824–827. https://doi.org/10.1126/science.298.5594.824 PMID: 12399590

34. Luscombe NM, Babu MM, Yu H, Snyder M, Teichmann SA, Gerstein M. Genomic analysis of regulatory network dynamics reveals large topological changes. Nature. 2004; 431(7006):308–312. https://doi.org/10.1038/nature02782 PMID: 15372033

35. Nair AG, Taira K, Brunton SL. Networked oscillator based modeling and control of unsteady wake flows. Physical Review E. 2018; 97(063107).

36. Halko N, Martinsson PG, Tropp JA. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. SIAM Review. 2011; 53(2):217–288. https://doi.org/10.1137/090771806

37. Drineas P, Mahoney MW. RandNLA: Randomized Numerical Linear Algebra. Commun ACM. 2016; 59(6):80–90. https://doi.org/10.1145/2842602

38. Kannan R, Vempala S. Randomized algorithms in numerical linear algebra. Acta Numerica. 2017; 26:95–135. https://doi.org/10.1017/S0962492917000058

39. Woodruff DP, et al. Sketching as a tool for numerical linear algebra. Foundations and Trends® in Theoretical Computer Science. 2014; 10(1–2):1–157.

40. Erichson NB, Voronin S, Brunton SL, Kutz JN. Randomized Matrix Decompositions Using R. Journal of Statistical Software. 2019; 89(11):1–48. https://doi.org/10.18637/jss.v089.i11

41. Nair AG, Yeh CA, Kaiser E, Noack BR, Brunton SL, Taira K. Cluster-based feedback control of turbulent post-stall separated flows. Journal of Fluid Mechanics, accepted. 2019. https://doi.org/10.1017/jfm.2019.469

42. Page L, Brin S, Motwani R, Winograd T. The PageRank citation ranking: Bringing order to the web. Stanford InfoLab; 1999.

43. Kolaczyk ED, Csárdi G. Statistical analysis of network data with R. vol. 65. Springer; 2014.

44. Clauset A, Newman ME, Moore C. Finding community structure in very large networks. Physical review E. 2004; 70(6):066111. https://doi.org/10.1103/PhysRevE.70.066111

45. Leicht EA, Newman ME. Community structure in directed networks. Physical review letters. 2008; 100(11):118703. https://doi.org/10.1103/PhysRevLett.100.118703 PMID: 18517839

46. Trefethen LN, Bau D, III. Numerical Linear Algebra. vol. 50. SIAM; 1997.

47. Bryan K, Leise T. The $25,000,000,000 eigenvector: The linear algebra behind Google. SIAM Review. 2006; 48(3):569–581. https://doi.org/10.1137/050623280

48. Mahoney MW. Randomized Algorithms for Matrices and Data. Foundations and Trends in Machine Learning. 2011; 3:123–224.

49. Drineas P, Mahoney MW. RandNLA: Randomized Numerical Linear Algebra. Commun ACM. 2016; 59(6):80–90. https://doi.org/10.1145/2842602

50. Liberty E, Woolfe F, Martinsson PG, Rokhlin V, Tygert M. Randomized algorithms for the low-rank approximation of matrices. Proceedings of the National Academy of Sciences. 2007; 104(51):20167–20172. https://doi.org/10.1073/pnas.0709640104

51. Sarlos T. Improved Approximation Algorithms for Large Matrices via Random Projections. In: Foundations of Computer Science. 47th Annual IEEE Symposium on; 2006. p. 143–152.

52. Martinsson PG, Rokhlin V, Tygert M. A Randomized Algorithm for the Decomposition of Matrices. Applied and Computational Harmonic Analysis. 2011; 30:47–68. https://doi.org/10.1016/j.acha.2010.02.003

53. Erichson NB, Brunton SL, Kutz JN. Compressed singular value decomposition for image and video processing. In: Proceedings of the IEEE International Conference on Computer Vision; 2017. p. 1880–1888.

54. Woolfe F, Liberty E, Rokhlin V, Tygert M. A Fast Randomized Algorithm for the Approximation of Matrices. Journal of Applied and Computational Harmonic Analysis. 2008; 25:335–366. https://doi.org/10.1016/j.acha.2007.12.002

55. Liberty E. Simple and Deterministic Matrix Sketching. In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM; 2013. p. 581–588.

56. Erichson NB, Donovan C. Randomized low-rank dynamic mode decomposition for motion detection. Computer Vision and Image Understanding. 2016; 146:40–50. https://doi.org/10.1016/j.cviu.2016.02.005

57. Erichson NB, Manohar K, Brunton SL, Kutz JN. Randomized CP tensor decomposition. arXiv preprint arXiv:170309074. 2017.

58. Erichson NB, Mendible A, Wihlborn S, Kutz JN. Randomized nonnegative matrix factorization. Pattern Recognition Letters. 2018; 104:1–7. https://doi.org/10.1016/j.patrec.2018.01.007

59. Erichson NB, Brunton SL, Kutz JN. Compressed dynamic mode decomposition for background modeling. Journal of Real-Time Image Processing. 2016; p. 1–14.

60. Shabat G, Shmueli Y, Aizenbud Y, Averbuch A. Randomized LU decomposition. Applied and Computational Harmonic Analysis. 2018; 44(2):246–272. https://doi.org/10.1016/j.acha.2016.04.006

61. Rokhlin V, Szlam A, Tygert M. A Randomized Algorithm for Principal Component Analysis. SIAM Journal on Matrix Analysis and Applications. 2009; 31:1100–1124. https://doi.org/10.1137/080736417

62. Frieze A, Kannan R, Vempala S. Fast Monte-Carlo Algorithms for Finding Low-Rank Approximations. Journal of the ACM. 2004; 51(6):1025–1041. https://doi.org/10.1145/1039488.1039494

63. Ma P, Mahoney MW, Yu B. A statistical perspective on algorithmic leveraging. Journal of Machine Learning Research. 2015; 16(1):861–911.

64. Drineas P, Mahoney MW, Muthukrishnan S. Sampling algorithms for l 2 regression and applications. In: Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm. Society for Industrial and Applied Mathematics; 2006. p. 1127–1136.

65. Drineas P, Magdon-Ismail M, Mahoney MW, Woodruff DP. Fast approximation of matrix coherence and statistical leverage. Journal of Machine Learning Research. 2012; 13(Dec):3475–3506.

66. Talwalkar A, Kumar S, Mohri M, Rowley H. Large-scale SVD and manifold learning. Journal of Machine Learning Research. 2013; 14(1):3129–3152.

67. Niederreiter H. Random number generation and quasi-Monte Carlo methods. SIAM; 1992.

68. Halton JH. Algorithm 247: Radical-inverse Quasi-random Point Sequence. Commun ACM. 1964; 7(12):701–702. https://doi.org/10.1145/355588.365104

69. Williams CK, Seeger M. Using the Nyström method to speed up kernel machines. In: Advances in neural information processing systems; 2001. p. 682–688.

70. Drineas P, Mahoney MW. On the Nyström method for approximating a Gram matrix for improved kernel-based learning. journal of machine learning research. 2005; 6(Dec):2153–2175.

71. Zhang K, Tsang IW, Kwok JT. Improved Nyström low-rank approximation and error analysis. In: Proceedings of the 25th international conference on Machine learning. ACM; 2008. p. 1232–1239.

72. Gittens A, Mahoney MW. Revisiting the Nyström method for improved large-scale machine learning. The Journal of Machine Learning Research. 2016; 17(1):3977–4041.

73. Kumar S, Mohri M, Talwalkar A. Sampling methods for the Nyström method. Journal of Machine Learning Research. 2012; 13(Apr):981–1006.

74. Gopalakrishnan Meena M, Taira K, Asai K. Airfoil-Wake Modification with Gurney Flap at Low Reynolds Number. AIAA Journal. 2017; 56(4):1348–1359. https://doi.org/10.2514/1.J056260

75. Williamson CH, Roshko A. Vortex formation in the wake of an oscillating cylinder. Journal of Fluids and Structures. 1988; 2(4):355–381. https://doi.org/10.1016/S0889-9746(88)90058-8

76. Taira K, Colonius T. The immersed boundary method: a projection approach. Journal of Computational Physics. 2007; 225(2):2118–2137. https://doi.org/10.1016/j.jcp.2007.03.005

77. Colonius T, Taira K. A fast immersed boundary method using a nullspace approach and multi-domain far-field boundary conditions. Computer Methods in Applied Mechanics and Engineering. 2008; 197:2131–2146. https://doi.org/10.1016/j.cma.2007.08.014

78. Boffetta G, Ecke RE. Two-dimensional turbulence. Annual Review of Fluid Mechanics. 2012; 44:427–451. https://doi.org/10.1146/annurev-fluid-120710-101240

79. Kida S. Numerical simulation of two-dimensional turbulence with high-symmetry. Journal of the Physical Society of Japan. 1985; 54(8):2840–2854. https://doi.org/10.1143/JPSJ.54.2840

80. Dellnitz M, Froyland G, Junge O. The algorithms behind GAIO—Set oriented numerical methods for dynamical systems. In: Ergodic theory, analysis, and efficient simulation of dynamical systems. Springer; 2001. p. 145–174.

81. Dellnitz M, Junge O. Set oriented numerical methods for dynamical systems. Handbook of dynamical systems. 2002; 2:221–264.

82. Froyland G, Padberg K. Almost-invariant sets and invariant manifolds—Connecting probabilistic and geometric descriptions of coherent structures in flows. Physica D. 2009; 238:1507–1523. https://doi.org/10.1016/j.physd.2009.03.002

83. Froyland G, Santitissadeekorn N, Monahan A. Transport in time-dependent dynamical systems: Finite-time coherent sets. Chaos. 2010; 20(4):043116–1–043116–16. https://doi.org/10.1063/1.3502450

84. Tallapragada P, Ross SD. A set oriented definition of finite-time Lyapunov exponents and coherent sets. Communications in Nonlinear Science and Numerical Simulation. 2013; 18(5):1106–1126. https://doi.org/10.1016/j.cnsns.2012.09.017

85. Kaiser E, Noack BR, Cordier L, Spohn A, Segond M, Abel M, et al. Cluster-based reduced-order modelling of a mixing layer. J Fluid Mech. 2014; 754:365–414. https://doi.org/10.1017/jfm.2014.355

86. Amsallem D, Zahr MJ, Farhat C. Nonlinear model order reduction based on local reduced-order bases. International Journal for Numerical Methods in Engineering. 2012; 92(10):891–916. https://doi.org/10.1002/nme.4371