Discovering multiscale and self-similar structure with data-driven wavelets

Daniel Floryan[a] and Michael D. Graham[a,1]

[a]Department of Chemical and Biological Engineering, University of Wisconsin–Madison, Madison, WI 53706

Edited by David A. Weitz, Harvard University, Cambridge, MA, and approved November 21, 2020 (received for review October 12, 2020)

Many materials, processes, and structures in science and engineering have important features at multiple scales of time and/or space; examples include biological tissues, active matter, oceans, networks, and images. Explicitly extracting, describing, and defining such features are difficult tasks, at least in part because each system has a unique set of features. Here, we introduce an analysis method that, given a set of observations, discovers an energetic hierarchy of structures localized in scale and space. We call the resulting basis vectors a "data-driven wavelet decomposition." We show that this decomposition reflects the inherent structure of the dataset it acts on, whether it has no structure, structure dominated by a single scale, or structure on a hierarchy of scales. In particular, when applied to turbulence—a high-dimensional, nonlinear, multiscale process—the method reveals self-similar structure over a wide range of spatial scales, providing direct, model-free evidence for a century-old phenomenological picture of turbulence. This approach is a starting point for the characterization of localized hierarchical structures in multiscale systems, which we may think of as the building blocks of these systems.

wavelet | multiscale | data-driven decomposition | machine learning | turbulence

Many important processes are multiscale in nature, meaning that they exhibit structure at multiple scales of time and/or space. In nature, a prominent example is the dynamics of oceans and associated interactions with the atmosphere, which govern the planet's weather and climate systems (1); much effort is expended in capturing and understanding effects at multiple scales of time and space (2). In engineering, a prominent example is networks, specifically social media networks. Networks have multiscale structure by virtue of hierarchies of communities of nodes in the networks (3). Understanding the structure of hierarchical communities in social media networks is crucial to understanding the spread of disinformation (and censorship of information) in these networks (4). Broadly speaking, identifying and understanding the features present in multiscale processes are crucial to understanding and controlling these processes. Although the application we focus on here will be turbulent fluid flows, the ensuing discussion applies to any multiscale process for which the notions of energy (variance in the statistical context) and localization (a form of sparsity) are relevant.

Turbulence is a canonical multiscale process consisting of localized concentrations of vortex motion that are coherent in space and time and coexist at a wide range of scales. Theoretical arguments indicate that at intermediate scales and far from walls the structure of a turbulent flow should be self-similar (5, 6). This notion is qualitatively illustrated in Fig. 1, which illustrates a snapshot from a simulation of homogeneous isotropic turbulence (HIT) at several scales (7–10). As with other multiscale processes, a great challenge in fluid dynamics is to rationally identify and analyze coherent structures from a complex turbulent flow field. While it is often mathematically convenient to analyze signals in the Fourier domain, trigonometric functions are not localized in space, and what one observes at an instant in time in a turbulent flow rarely, if ever, looks sinusoidal. Alternately, conventional wavelet bases, which are localized and self-similar, can be used for analysis (11). In both the Fourier and wavelet approaches the bases for representing the flow are imposed a priori rather than emerging from data.

One of the primary methods of extracting structure from data is principal components analysis (PCA), which in fluid dynamics is typically denoted proper orthogonal decomposition (12) (see ref. 13 for other popular modal decomposition methods). Given an ensemble (often a time series) of data, PCA yields a data-driven orthogonal basis whose elements are optimally ordered by energy content. When applied to velocity field data for a fluid flow, the resulting basis elements may be thought of as the building blocks of that flow, and its application has yielded many structural and dynamical insights (12, 14). One limitation of PCA is that the basis elements tend not to be localized in space; indeed, for directions in which a field is statistically homogeneous, the PCA basis elements are Fourier modes (12). In this case, not only do the PCA modes have no localization in space but they also reveal no information about the flow beyond what Fourier decomposition would provide.

A well-known formalism that produces bases with spatially localized elements is that of wavelets. The name is quite descriptive: Wavelets are localized waves. In particular, wavelet decompositions provide an orthogonal basis whose elements are localized in both space and scale. Traditionally, the basis elements are translations and dilations of a single vector called the mother wavelet (15–19). *SI Appendix*, section 1 provides a concise summary of results relevant to the present work. Traditional wavelet methods (where the mother wavelet is prescribed a priori) have already found use in turbulence precisely because of the space-scale unfolding they produce (11, 20–27), giving hope that data-driven methods based on wavelets may lead to new insights into turbulence.

A myriad of data-driven methods of structure identification and extraction based on wavelets have been developed (e.g.,

Significance

Multiscale structure is all around us: in biological tissues, active matter, oceans, networks, and images. Identifying the multiscale features of these systems is crucial to our understanding and control of them. We introduce a method that rationally extracts localized multiscale features from data, which may be thought of as the building blocks of the underlying phenomena.

Author contributions: D.F. and M.D.G. designed research, performed research, analyzed data, and wrote the paper.

The authors declare no competing interest.

This article is a PNAS Direct Submission.

[1]To whom correspondence may be addressed. Email: mdgraham@wisc.edu.

This article contains supporting information online at https://www.pnas.org/lookup/suppl/doi:10.1073/pnas.2021299118/-/DCSupplemental.

Published December 21, 2020.

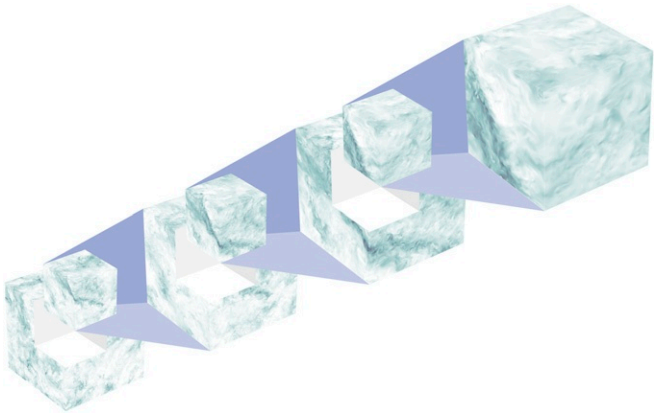

Fig. 1. Snapshot of HIT from the Johns Hopkins Turbulence Databases (7–10), showing the kinetic energy per unit mass, with darker color corresponding to greater energy.

refs. 28–38). Although these methods may yield localized structures, they are limited in that the construction of the resulting basis elements is prescribed in either scale or frequency, and many impose self-similarity on the basis, as is done with traditional wavelets (the "empirical wavelet transform" of ref. 28 does not have this feature but relies on the existence of local maxima in the power spectrum of a signal, making it ill-suited to phenomena like turbulence without such local maxima).

In the present work we develop a method that integrates the data- and energy-driven nature of PCA with the space and scale localization properties of wavelets. As our derivation and illustrative examples will reveal, we impose very little structure in our method, so any structure in the basis may be attributed to the underlying structure of the dataset under consideration. We call the resulting basis a "data-driven wavelet decomposition" (DDWD) and use it to gain insights into the structure of turbulence, though we emphasize that the method is general in its application.

Formulation

Before presenting the DDWD, it will be useful to introduce key features of PCA and conventional wavelet decompositions. Suppose we have a dataset $\{z_i\}_{i=1}^M \in \mathbb{R}^N$, each z_i being a sample data vector (e.g., one component of a velocity field uniformly sampled along a line through the flow). We can arrange the dataset into a matrix $Z \in \mathbb{R}^{N\times M}$ whose columns are the data vectors z_i, normalized so that $\mathrm{tr}\, ZZ^T = 1$ (the normalization does not change the results of PCA, but is done here because it parallels our formulation of DDWD later). PCA seeks an ordered orthonormal basis $\{\phi_i\}_{i=1}^N$ such that the energy of the dataset projected onto the first $K \leq N$ basis elements is maximized. One way to state this problem, which parallels our later description of data-driven wavelets, is as follows. We determine the first basis element ϕ_1 so that the projection of the data onto this element is maximized. This problem can be written

$$\max_{\phi} \quad \phi^T ZZ^T \phi \qquad [1]$$

$$\text{s.t.} \quad \phi^T \phi = 1. \qquad [2]$$

The solution to this problem is the eigenvector of ZZ^T with the largest eigenvalue. The second basis element ϕ_2 is found by projecting out the component of the data in the ϕ_1 direction and repeating, yielding that ϕ_2 is the eigenvector of ZZ^T with the second-largest eigenvalue. Basis elements ϕ_i solve

$$\max_{\phi} \quad \left\| \phi^T \left(Z - \sum_{j=1}^{i-1} \phi_j \phi_j^T Z \right) \right\|_2^2 \qquad [3]$$

$$\text{s.t.} \quad \phi^T \phi = 1, \quad \phi^T \phi_j = 0,\ j = 1, \ldots, i-1. \qquad [4]$$

This formulation is recursive, producing a hierarchy of subspaces ordered by how much of the dataset's energy (Frobenius norm) they contain: $\mathbb{R}^N = \mathrm{span}\{\phi_1\} \oplus \ldots \oplus \mathrm{span}\{\phi_N\}$. The basis elements ϕ_i are the eigenvectors of ZZ^T. For statistically homogeneous data in a periodic domain, ZZ^T (more precisely, its expected value) is circulant, in which case the ϕ_i are simply discrete Fourier modes.

Traditional wavelet decompositions also produce a hierarchy of orthogonal subspaces, but there are important differences from PCA. First, the basis elements are not determined from data but are selected a priori; there are many standard options (18). Second, by construction, the decomposition produces a hierarchy of orthogonal subspaces ordered by scale, as shown in Fig. 2*A*. We consider periodic vectors on $\mathbb{R}^N$, with N even (19). This space is split into subspaces V_{-1} and W_{-1}, each of dimension $N/2$, and each spanned by the even translates of vectors ϕ_{-1} (the father wavelet) and ψ_{-1} (the mother wavelet), respectively. Once ϕ_{-1} is known, ψ_{-1} can be found, and vice versa. The father and mother wavelets, and their even translates, are mutually orthonormal by construction. Subspace V_{-1} is called an approximation subspace because it contains all of the low frequencies, and W_{-1} is called a detail subspace because it contains all of the high frequencies. Given a signal, its projection onto V_{-1} produces a low-pass-filtered version of the signal, and its projection onto W_{-1} produces the detail needed to reconstruct the full signal. We then recursively split the approximation subspaces. For $N = 2^p$ (which we assume throughout), we get a hierarchy of

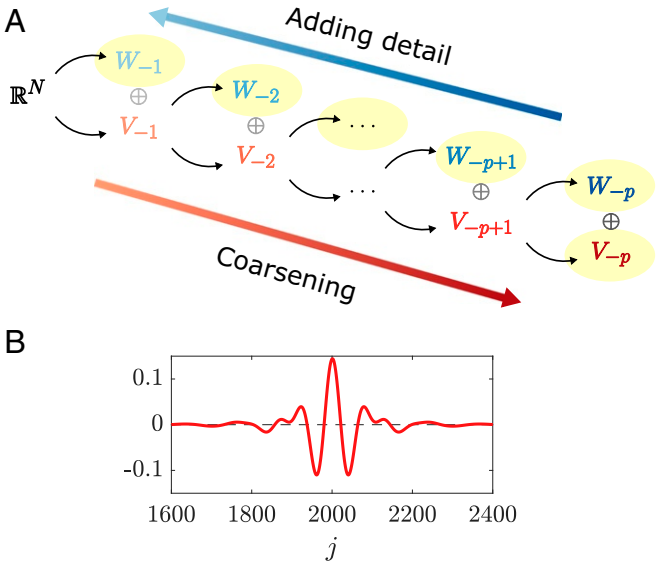

Fig. 2. (*A*) Subspaces from wavelets on $\mathbb{R}^N$. At stage l, approximation subspace V_{-l} is split into detail subspace W_{-l-1} and approximation subspace V_{-l-1}, each half the dimension of V_{-l}. Subspace V_{-l} is spanned by the $N/2^l$ translates by 2^l of ϕ_{-l}, and W_{-l} is spanned by the $N/2^l$ translates by 2^l of ψ_{-l}. The full space is decomposed into progressively coarser subspaces, $\mathbb{R}^N = W_{-1} \oplus \ldots \oplus W_{-p} \oplus V_{-p}$, or, going the other way, into the addition of progressively finer details. These subspaces are highlighted. In the present work, an ensemble of data is used to define a specific decomposition of this form. (*B*) Discrete Meyer wavelet for $N = 4{,}096$ and $l = 6$.

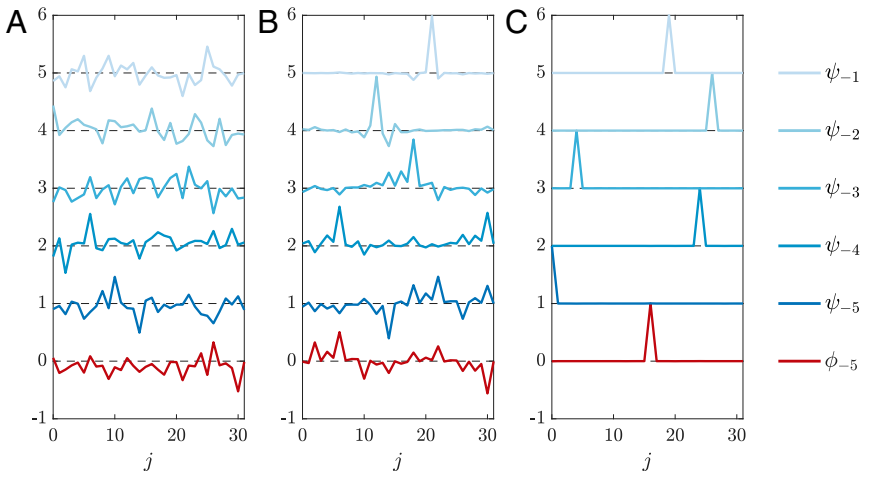

Fig. 3. White noise wavelets on $\mathbb{R}^{2^5}$. Coloring as in Fig. 2*A*. No variance penalty (*A*), small variance penalty (*B*), and large variance penalty (*C*).

subspaces of progressively coarser scales: $\mathbb{R}^N = W_{-1} \oplus \ldots \oplus W_{-p} \oplus V_{-p}$. For traditional wavelets, the sets of wavelets $\{\phi_i\}$ and $\{\psi_i\}$ are determined from the father and mother wavelets, respectively, by a rescaling operation that is essentially a simple dilation by a factor of two (see *SI Appendix*, section 4A, for more details). This process leads to a hierarchical basis structure of the form shown in Fig. 2*A*.

The DDWD combines the hierarchical structure of wavelets that is shown in Fig. 2*A* with the energetic optimization of PCA. Namely, each time we split a subspace, we design the subsequent subspaces so that the approximation subspace contains as much of the dataset's energy as possible.

The first step of the process is to find the wavelet generator u, for which the projection of the data onto this vector and its even translates is maximized. We define V_{-1} as the subspace spanned by these vectors, thus beginning the data-driven construction of a hierarchy with the structure of Fig. 2*A*. This maximization is subject to 1) the constraint that u and its even translates are mutually orthonormal and 2) a penalty on the width of u, as measured by its circular variance $\mathrm{Var}(u)$. This problem is stated as

$$\max_u u^T A u - \lambda^2 \mathrm{Var}(u), \quad A = \frac{1}{\|Z\|_F^2} \sum_{k=0}^{N/2-1} R^{-2k} Z Z^T R^{2k} \qquad [5]$$

$$\text{s.t. } u^T R^{2k} u = \delta_{k0},\ k = 0, \ldots, N/2 - 1. \qquad [6]$$

Here λ measures the penalty on the variance, whose effect on the results we illustrate below, and R is the circular shift operator: For example, if $u = [a, b, c, d]^T$, then $Ru = [d, a, b, c]^T$. The solution u and its even translates generate the vectors ϕ_{-1} and ψ_{-1}; the former span V_{-1} and the latter W_{-1}. We then project the data onto V_{-1}, replace N by $N/2$ in the definition of A and the orthonormality constraints, decrease λ by a factor of 2, and repeat, yielding ϕ_{-2} and ψ_{-2}, and thus the subspaces V_{-2} and W_{-2}. We proceed recursively, finding the subspaces V_{-l} and W_{-l} such that V_{-l} contains the maximal amount of energy of the dataset. Extensive details are found in *SI Appendix*, section 2. In the end, we find an energetic hierarchy of subspaces, optimized stage by stage, whose elements are orthogonal and localized. In contrast to previous data-driven methods incorporating wavelets, which impose restrictive structure, the only structure we impose is orthogonality, localization, and the hierarchy of Fig. 2*A*. In *SI Appendix*, section 3 we also draw parallels between the DDWD and convolutional neural networks and show how the DDWD naturally incorporates pooling and skip connections, two tricks that improve the performance of neural network architectures (39). Together with its inverse transform, the DDWD is akin to a convolutional autoencoder, but with the additional features of orthogonality of all elements, stagewise energetic optimality, and the ability to unambiguously extract structure, which make the results interpretable.

We make a point to note that for the DDWD the stage l of the hierarchy should not be conflated with the concept of scale. For traditional wavelets, stage and scale are interchangeable since whenever a subspace is split the lower half of frequencies is always pushed to the approximation subspace and the upper half of frequencies is always pushed to the detail subspace. For the DDWD, however, the distribution of frequencies among the subspaces is dictated by energetic considerations, which depends on the dataset under consideration. An example below will elucidate this point.

Results

We will demonstrate the DDWD on three datasets with increasingly complex structure to show that the method extracts structure inherent to the data.

Gaussian Random Data. The first dataset we consider consists of Gaussian white noise, which has no structure. By construction, the basis produced by the DDWD is orthonormal, so the change-of-basis transformation is orthogonal. Any orthogonal

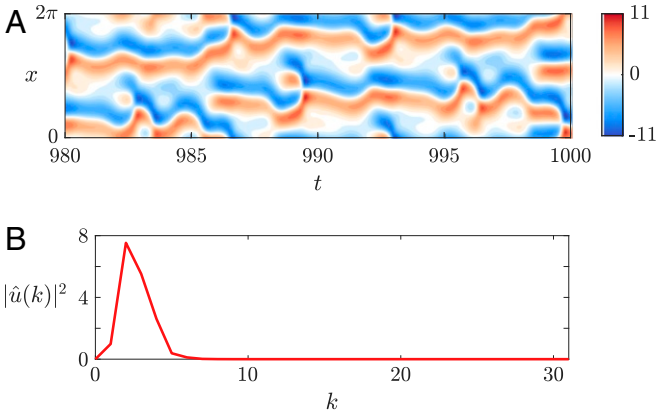

Fig. 4. Trajectory (*A*) and attendant power spectrum (*B*) of the Kuramoto–Sivashinsky equation.

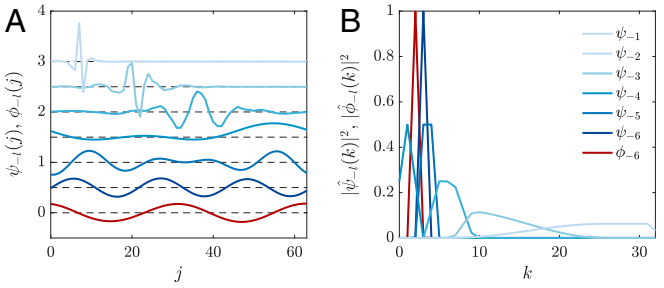

Fig. 5. Kuramoto–Sivashinsky wavelets (*A*), offset from each other by 0.5, and their power spectra (*B*). Coloring as in Fig. 2*A*. The variance penalty is $\lambda^2 = 0.1$.

transformation of Gaussian white noise produces Gaussian white noise. Therefore, applied to Gaussian white noise, the coordinates of the data in the DDWD basis (the wavelet coefficients) will be Gaussian white noise, so all wavelet coefficients will be uncorrelated and have energy equal to that of the input Gaussian white noise. As long as we do not impose a variance penalty, this result implies that for Gaussian white noise there is no optimal set of wavelets, in the sense we have defined. In other words, the DDWD reflects that the dataset has no structure. If we do impose a variance penalty, then the optimal wavelets become discrete delta functions (i.e., the Euclidean basis vectors). The reason for this is simple: All wavelets capture the energy of white noise equally well, but the delta function will be the most localized among them.

The result that all wavelets capture the energy of Gaussian white noise equally well highlights an interesting fact about the DDWD. In Fig. 3 we show three sets of wavelets that are computed from a dataset of Gaussian white noise. Fig. 3*A* has no variance penalty, Fig. 3*B* has a small variance penalty, Fig. 3*C* has a large variance penalty, and all wavelets are colored according to the color coding used in Fig. 2*A*. Despite the fact that we have used the structure of Fig. 2*A*, there is no apparent hierarchy of scales among the left set of wavelets. This highlights what we noted earlier, that the concept of scale is not built into the DDWD; rather, it must be learned from the data. When we add a small variance penalty, wavelets corresponding to finer-detail subspaces are more localized, but all wavelets are jagged; this will contrast with our later examples where wavelets corresponding to later stages are smoother, reflecting the inherent structure of the later examples. Note that although the central set of wavelets was computed with nonzero variance penalty, they are not delta functions as we had asserted earlier; this is due to the dataset containing a finite number of samples, and this effect weakens as the number of samples increases or as the variance penalty is increased (as for the right set of wavelets). In Fig. 3*C*, all of the vectors are discrete delta functions; while this might seem redundant, only certain translates of the discrete delta function are included in each stage, and the resulting basis consists of delta functions localized at each mesh point.

Kuramoto–Sivashinsky Chaos. The second dataset we consider comes from the Kuramoto–Sivashinsky equation,

$$u_t + uu_x + u_{xx} + \nu u_{xxxx} = 0, \qquad [7]$$

for $0 \le x \le 2\pi$, with periodic boundary conditions and $\nu = (\pi/11)^2$, which yields chaotic dynamics. We compute a numerical solution using a pseudo-spectral method with 64 Fourier modes and assemble a dataset consisting of 90,001 snapshots taken from a single trajectory. The latter part of the trajectory and the power spectrum in Fig. 4 clearly show that the structure is dominated by a single length scale with wavenumber k around 2 to 3.

We compute the DDWD with a range of variance penalties, showing the result for $\lambda^2 = 0.1$ in Fig. 5 (others are shown in *SI Appendix*, section 4B). We only show one set of wavelets because, no matter the variance penalty, the coarsest subspaces are the same: V_{-6} is spanned by a sine wave with wavenumber $k = 2$ (the most energetic wavenumber), W_{-6} is spanned by a sine wave with wavenumber $k = 3$ (the second-most-energetic wavenumber), and W_{-5} is spanned by a vector (and its translate) containing only wavenumbers $k = 3$ and 4 ($k = 4$ is the next most energetic wavenumber). The DDWD is thus robust in pushing the dominant (most energetic) length scales of the system to the lowest stages. Moreover, the energy contained in each subspace is also robust to the variance penalty (*SI Appendix*). The first difference between wavelets computed with different variance penalties appears in the subspace W_{-4}, spanned by the four translates of ψ_{-4}. As the variance penalty is increased the wavenumber $k = 8$ is exchanged for $k = 0$. Energetically, this makes little difference since $k = 8$ is highly damped by the hyperviscous term and contains very little energy, and $k = 0$ contains identically zero energy (for the boundary conditions we use, the spatial mean is constant and can be set to zero). The compositions of the finer detail subspaces do not change qualitatively with variance penalty, with finer detail subspaces containing higher wavenumbers. As the variance penalty is increased, localization in the Fourier domain is exchanged for localization in the spatial domain.

Homogeneous Isotropic Turbulence. The final and primary dataset we consider is of forced HIT, taken from the Johns Hopkins Turbulence Databases (7–10). We use a single snapshot from a direct numerical simulation on a $4{,}096^3$ periodic grid with a Taylor-scale Reynolds number of 610.57, shown in Fig. 1; more details are available in the database's documentation. Our dataset consists of the velocity component aligned with 16,384 randomly sampled lines (the "longitudinal velocity") that are parallel to the axes. Each sample is a vector of length N = 4,096. The power spectrum is broad and has the expected $-5/3$ power law in the inertial subrange, which roughly contains wavenumbers $k \in [2, 60]$.

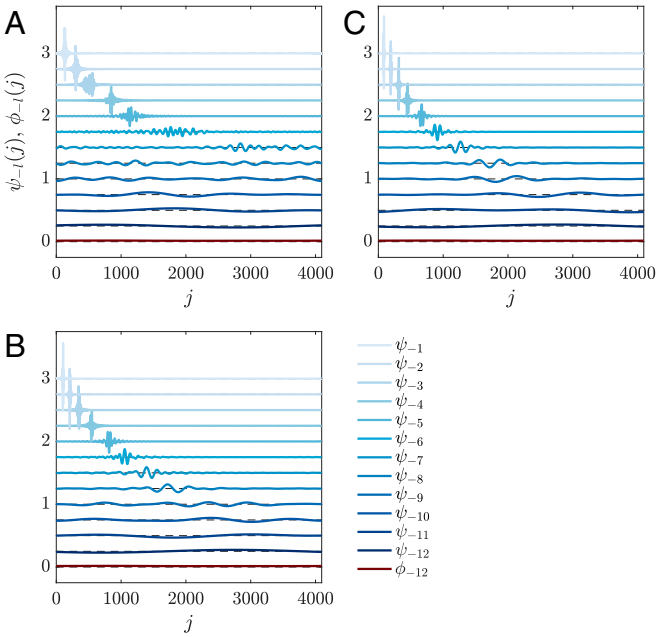

Fig. 6. HIT wavelets, vertically offset from each other by 0.25. Coloring as in Fig. 2*A*. The variance penalties are $\lambda^2 = 10^{-1}$ (*A*), $\lambda^2 = 10^{0}$ (*B*), and $\lambda^2 = 10^{1}$ (*C*).

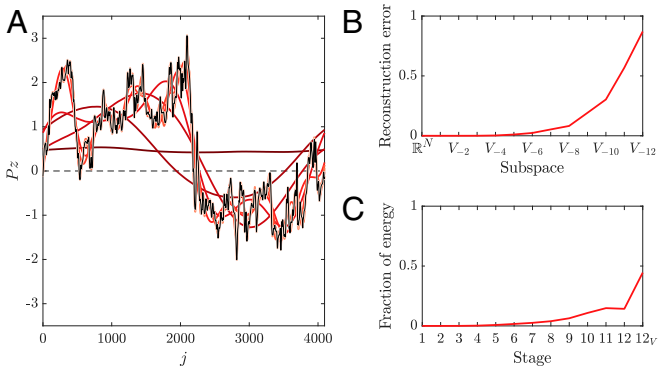

Fig. 7. Projection (denoted *P*) of one vector (denoted *z*) in the turbulence dataset onto the subspaces V_{-l} computed with $\lambda^2 = 10^1$ (*A*), with coloring as in Fig. 2*A*. The thin dashed line shows the origin, and the thin solid line shows the original vector. Also shown are the reconstruction error of each projection (*B*) and the energy of the dataset contained in each stage for all variance penalties considered (*C*) ($\lambda^2 = 0, 10^{-1}, 10^0$ and 10^1; only the result for $\lambda^2 = 10^1$ [red] can be seen).

Fig. 6 shows the DDWD with various variance penalties (their power spectra are shown in *SI Appendix*, section 4C). While at $\lambda^2 = 10^{-1}$ the wavelets are well-localized only for $l \leq 5$, for $\lambda^2 = 10^0$ and 10^1 localization is observed for $l \leq 8$ and 9, respectively. Moreover, despite the order-of-magnitude difference in λ^2 between the latter two cases, the wavelets for $4 \leq l \leq 8$ are nearly indistinguishable (see *SI Appendix* for more details). Furthermore, with increasing l, the wavelets have increasing scale: The DDWD reveals a hierarchy of scales present in the dataset, a known feature of turbulence. Recall that this feature is not built into the DDWD; rather, the method has extracted the concept of scale hierarchy from the turbulence dataset. In this case, it is appropriate to conflate stage and scale.

It is also worth noting that with increasing variance penalty the composition of each scale in the Fourier domain (shown in *SI Appendix*) becomes smoother and more robust, varying less across different trials. Overall, the composition of the wavelets in the Fourier domain is robust to the variance penalty.

To illustrate the reconstruction of data vectors using the DDWD basis, Fig. 7*A* shows one vector from the turbulence dataset and its projections onto the subspaces V_{-l} computed with $\lambda^2 = 10^1$. Lighter colors show more detailed reconstructions, and the thin black line shows the original data vector. At the coarsest level of approximation, we essentially reconstruct the spatial mean and then add progressively finer-scale features as we add smaller scale wavelet components. Fig. 7 *B* and *C*, respectively, show the reconstruction errors of the progressively finer projections, and the energy of the entire dataset contained in each stage, for $\lambda^2 = 0, 10^{-1}, 10^0$, and 10^1. The differences in these quantities as λ changes are visibly indistinguishable, indicating robustness of the DDWD with respect to variance penalty.

Most interestingly, we check the wavelets that arise from the HIT data for self-similarity across stages. We present here results for the most localized wavelets, corresponding to $\lambda^2 = 10^1$, and show in *SI Appendix*, Fig. S10, that the same conclusions hold for $\lambda^2 = 10^0$. Fig. 8 *A*–*E* show wavelets ψ_{-l} for $4 \leq l \leq 8$; note the change in horizontal scale from plot to plot. Aside from their horizontal scale, these wavelets are evidently very similar looking. The figure also shows on each plot the rescaled version of the wavelet at the previous level, $S\psi_{-l+1}$, where S essentially dilates a vector by a factor of 2 and rescales it so that it has unit norm. (See *SI Appendix*, section 4A for a precise description of S and *SI Appendix*, Figs. S10 and S11 for plots of ψ_{-l} and $S\psi_{-l+1}$ for all l.) For ease of comparison, we have shifted the wavelets and in some cases reflected them about their axes. In all cases shown, ψ_{-l} and $S\psi_{-l+1}$ are nearly indistinguishable, indicating strong self-similarity across stages $l = 4$ to $l = 8$. This observation can be quantified: Fig. 8*F* shows the inner product $\psi_{-l}^T S\psi_{-l+1}$, whose absolute value is bounded by 0 and 1, for all stages. It is very close to unity for $l > 3$. This strong self-similarity also holds for the lower variance penalty $\lambda^2 = 10^0$, as shown in *SI Appendix*, Fig. S10, indicating that it is a robust feature derived from the data. Stages 4 to 8 contain the approximate wavenumbers $k \in [10, 200]$, which coincides with the inertial subrange where self-similarity is expected. (The larger scales are no longer localized, so we draw no significance from the high measure of similarity in those cases.) Interestingly, the wavelets in the self-similar range are quite similar to the discrete Meyer wavelet (18), shown in Fig. 2*B*, as well as to the Battle–Lemarié wavelet used by Meneveau in his analysis of turbulent flows (27). Performing Meneveau's analysis with our data-driven wavelets would likely yield similar results, at least in the self-similar range.

It bears repeating that the self-similarity of the wavelets produced by the DDWD is not a result of the method; rather, it is a reflection of the system. In the case of the Kuramoto–Sivashinsky system, where we know there is no similarity across scales, there is generally no relation between the data-driven wavelets across scales. For HIT, where self-similarity is hypothesized in a certain range of scales, the data-driven wavelets show self-similarity. Hellström et al. (14) made a somewhat related observation in turbulent pipe flow. They performed PCA on a set of experimentally obtained velocity fields from a cross-section of the pipe and found that they could rescale the modes so that they overlapped. This observation is consistent with the attached eddy hypothesis about the structure of wall turbulence (5, 40). Their modes were global in space, as usually results from PCA; this is particularly true for the azimuthal direction, for which PCA yields Fourier modes due to periodicity. For the HIT data, which are periodic in all three directions, PCA would yield Fourier modes in all three directions, revealing no information about the system that could not be obtained from Fourier decomposition.

Conclusions

We have introduced a method that integrates key aspects of PCA and wavelet analysis to yield a DDWD. This method takes an ensemble of data vectors corresponding to field values at a lattice of points in space (or time) and generates a hierarchical

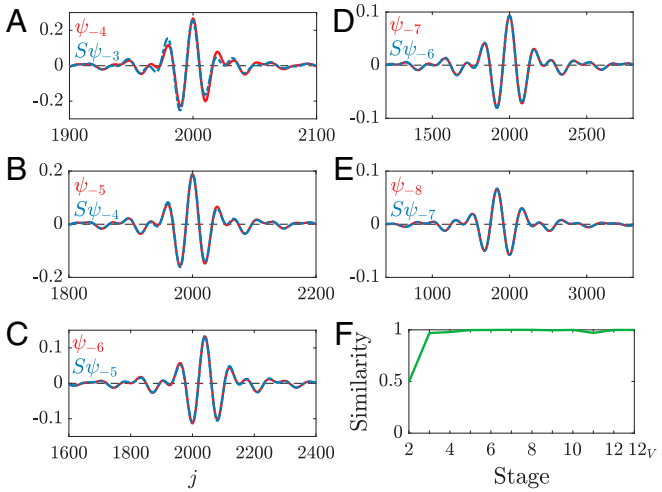

Fig. 8. Comparison between computed wavelets ($\lambda^2 = 10^1$) and ones obtained by dilating and rescaling the wavelet from the previous stage for stages $l = 4$ to $l = 8$ (*A*–*E*) and the level of similarity across all stages (*F*).

orthogonal basis. In contrast to traditional wavelet bases, the basis elements at each stage are not simply dilations of given mother or father wavelets but rather are determined stage-by-stage from the data. For data that is not self-similar, neither are the resulting basis elements. Rather, these represent the differing structures at the different stages. In contrast, for self-similar data, the basis vectors at different stages are related to one another by a simple rescaling. Indeed, for data from HIT—a high-dimensional, nonlinear, multiscale process—we show self-similarity of the wavelet basis elements, which in turn reveals the self-similarity of the data, providing direct evidence for a century-old phenomenological picture of turbulence.

Future work on the DDWD will need to extend the methodology to multiple dimensions, different boundary conditions, and unstructured domains. As a start, tensor products can be used to address the first issue, boundary wavelets can be used to address the second issue (18), and wavelets on graphs can be used to address the last issue (41). For incompressible fluid flows, velocity fields are vector-valued and divergence-free; Farge et al. (23) provide a few options to handle this case that may be generalizable to the data-driven case. Attention must also be given to the development of efficient optimization algorithms for computing the basis. Finally, based on the ability of the present method to extract self-similar basis elements from self-similar turbulent flow data, we view it as a potentially important new starting point for identification and characterization of localized hierarchical turbulent structures in a wide variety of fluid flows, as well as other complex multiscale systems. We are particularly interested in applying the DDWD to wall-bounded flows and making connections with the attached eddy model of turbulence.

Data Availability. Simulation data and code have been deposited in GitHub, available at https://github.com/dfloryan/DDWD.

ACKNOWLEDGMENTS. This work was supported by *Air Force Office of Scientific Research* grant FA9550-18-0174 and *Office of Naval Research* grant N00014-18-1-2865 (Vannevar Bush Faculty Fellowship).

1. W. K. M. Lau, D. E. Waliser, *Intraseasonal Variability in the Atmosphere-Ocean Climate System* (Springer Science & Business Media, 2011).
2. P. F. J. Lermusiaux *et al.*, Multiscale modeling of coastal, shelf, and global ocean dynamics. *Ocean Dyn.* **63**, 1341–1344 (2013).
3. Y. Y. Ahn, J. P. Bagrow, S. Lehmann, Link communities reveal multiscale complexity in networks. *Nature* **466**, 761–764 (2010).
4. S. Bradshaw, P. N. Howard, "Challenging truth and trust: A global inventory of organized social media manipulation" (Computational Propaganda Research Project, University of Oxford, Oxford, 2018).
5. I. Marusic, J. P. Monty, Attached eddy model of wall turbulence. *Annu. Rev. Fluid Mech.* **51**, 49–74 (2019).
6. P. Sagaut, C. Cambon, *Homogeneous Turbulence Dynamics* (Springer International Publishing, Cham, Switzerland, 2018).
7. E. Perlman, R. Burns, Y. Li, C. Meneveau, "Data exploration of turbulence simulations using a database cluster" in *Proceedings of the 2007 ACM/IEEE Conference on Supercomputing* (ACM, 2007), pp. 1–11.
8. Y. Li *et al.*, A public turbulence database cluster and applications to study Lagrangian evolution of velocity increments in turbulence. *J. Turbul.* **9**, N31 (2008).
9. P. K. Yeung, D. A. Donzis, K. R. Sreenivasan, Dissipation, enstrophy and pressure statistics in turbulence simulations at high Reynolds numbers. *J. Fluid Mech.* **700**, 5–15 (2012).
10. P. K. Yeung, *et al.*, Forced isotropic turbulence dataset on 4096^3 grid. Johns Hopkins Turbulence Databases. https://doi.org/10.7281/T1DN4363. Accessed 8 July 2020.
11. M. Farge, Wavelet transforms and their applications to turbulence. *Annu. Rev. Fluid Mech.* **24**, 395–458 (1992).
12. P. Holmes, J. L. Lumley, G. Berkooz, C. W. Rowley, *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*, (Cambridge University Press, Cambridge, UK, ed. 2, 2012).
13. K. Taira et al., Modal analysis of fluid flows: An overview. *AIAA J.* **55**, 4013–4041 (2017).
14. L. H. O. Hellström, I. Marusic, A. J. Smits, Self-similarity of the large-scale motions in turbulent pipe flow. *J. Fluid Mech.* **792**, R1–R12 (2016).
15. G. Strang, Wavelets and dilation equations – A brief introduction. *SIAM Rev.* **31**, 614–627 (1989).
16. I. Daubechies, *Ten Lectures on Wavelets* (SIAM, 1992).
17. Y. Meyer, *Wavelets and Operators: Volume 1* (Cambridge University Press, 1992), vol. 37.
18. S. Mallat, *A Wavelet Tour of Signal Processing* (Elsevier, 1999).
19. M. W. Frazier, *An Introduction to Wavelets Through Linear Algebra* (Springer Science & Business Media, 2006).
20. F. Argoul *et al.*, Wavelet analysis of turbulence reveals the multifractal nature of the Richardson cascade. *Nature* **338**, 51–53 (1989).
21. R. Everson, L. Sirovich, K. R. Sreenivasan, Wavelet analysis of the turbulent jet. *Phys. Lett.* **145**, 314–322 (1990).
22. M. Farge, G. Pellegrino, K. Schneider, Coherent vortex extraction in 3D turbulent flows using orthogonal wavelets. *Phys. Rev. Lett.* **87**, 054501 (2001).
23. M. Farge, K. Schneider, G. Pellegrino, A. A. Wray, R. S. Rogallo, Coherent vortex extraction in three-dimensional homogeneous turbulence: Comparison between CVS-wavelet and POD-Fourier decompositions. *Phys. Fluids* **15**, 2886–2896 (2003).
24. N. Okamoto, K. Yoshimatsu, K. Schneider, M. Farge, Y. Kaneda, Coherent vortices in high resolution direct numerical simulation of homogeneous isotropic turbulence: A wavelet viewpoint. *Phys. Fluids* **19**, 115109 (2007).
25. J. Ruppert-Felsot, M. Farge, P. Petitjeans, Wavelet tools to study intermittency: Application to vortex bursting. *J. Fluid Mech.* **636**, 427–453 (2009).
26. M. Yamada, K. Ohkitani, An identification of energy cascade in turbulence by orthonormal wavelet analysis. *Prog. Theor. Phys.* **86**, 799–815 (1991).
27. C. Meneveau, Analysis of turbulence in the orthonormal wavelet representation. *J. Fluid Mech.* **232**, 469–520 (1991).
28. J. Gilles, Empirical wavelet transform. *IEEE Trans. Signal Process.* **61**, 3999–4010 (2013).
29. R. A. Gopinath, J. E. Odegard, C. S. Burrus, Optimal wavelet representation of signals and the wavelet sampling theorem. *IEEE Trans. Circuit. Syst. II Analog Digit. Signal Process.* **41**, 262–277 (1994).
30. U. Grasemann, R. Miikkulainen, Evolving wavelets using a coevolutionary genetic algorithm and lifting. *Lect. Notes Comput. Sci.* 3103, 969–980 (2004).
31. M. A. Mendez, M. Balabane, J. M. Buchlin, Multi-scale proper orthogonal decomposition of complex fluid flows. *J. Fluid Mech.* **870**, 988–1036 (2019).
32. B. Ophir, M. Lustig, M. Elad, Multi-scale dictionary learning using wavelets. *IEEE J. Select. Topics Signal Process.* **5**, 1014–1024 (2011).
33. D. Recoskie, R. Mann, Learning sparse wavelet representations. arXiv:1802.02961 (8 February 2018).
34. D. Recoskie, R. Mann, "Learning filters for the 2D wavelet transform" in *2018 15th Conference on Computer and Robot Vision (CRV)* (IEEE, 2018), pp. 198–205.
35. D. Recoskie, R. Mann, Gradient-based filter design for the dual-tree wavelet transform. arXiv:1806.01793 (4 June 2018).
36. A. Søgaard, Learning optimal wavelet bases using a neural network approach. arXiv:1706.03041 (25 March 2017).
37. A. H. Tewfik, D. Sinha, P. Jorgensen, On the optimal choice of a wavelet for signal representation. *IEEE Trans. Inf. Theor.* **38**, 747–765 (1992).
38. Y. Zhuang, J. S. Baras, "Optimal wavelet basis selection for signal representation" in *Wavelet Applications*, H. H. Szu, Ed. (International Society for Optics and Photonics, 1994), vol. 2242, pp. 200–211.
39. I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning* (MIT Press, 2016).
40. Y. Hwang, Statistical structure of self-sustaining attached eddies in turbulent channel flow. *J. Fluid Mech.* **767**, 254–289 (2015).
41. D. K. Hammond, P. Vandergheynst, R. Gribonval, Wavelets on graphs via spectral graph theory. *Appl. Comput. Harmon. Anal.* **30**, 129–150 (2011).

PNAS
www.pnas.org

Supplementary Information for

Discovering multiscale and self-similar structure with data-driven wavelets

Daniel Floryan and Michael D. Graham

Michael D. Graham.
E-mail: mdgraham@wisc.edu

This PDF file includes:

Supplementary text
Figs. S1 to S11 (not allowed for Brief Reports)
SI References

Supporting Information Text

1. Background on wavelets

Since we work with data, we restrict our attention to discrete vectors of finite length. Such vectors can be represented by many bases, the most common being the Euclidean basis and the Fourier basis. Two nice features of the discrete Fourier basis are that it diagonalizes translation-invariant linear transformations, and the coordinates of a vector in the discrete Fourier basis can be computed quickly using the fast Fourier transform (FFT). Furthermore, the elements of the discrete Fourier basis have perfect localization in frequency, that is, the discrete Fourier transform (DFT) of any element of the discrete Fourier basis is a vector of zeroes aside from a single entry with unit magnitude. A drawback of the discrete Fourier basis, however, is that its elements have no localization in space, that is, the modulus of any element is a vector with all entries equal to the same constant. In contrast, the elements of the Euclidean basis have no localization in frequency, but perfect localization in space.

Wavelets provide a happy medium, allowing us to construct a basis whose elements have some degree of localization in both space and frequency. A vector's expansion in a wavelet basis will provide both spatial and frequency information. Below, we describe wavelets in $\mathbb{C}^N$, the space of length N vectors with inner product

$$\langle z, w\rangle = \sum_{k=0}^{N-1} z(k)\overline{w(k)}, \qquad [1]$$

and associated norm

$$\|z\| = \left(\sum_{k=0}^{N-1} |z(k)|^2\right)^{1/2}, \qquad [2]$$

where the overbar denotes complex conjugation. Throughout, $z(k)$ refers to the k^{th} element of the vector z, indexed beginning from zero. In addition, we extend $z \in \mathbb{C}^N$ to be defined at all integers by requiring z to be periodic with period N: $z(j+N) = z(j)\ \forall\, j \in \mathbb{Z}$. The following is based on chapter 3 of **(author?)** (1).

Assume N is divisible by 2. A first-stage wavelet basis for $\mathbb{C}^N$ is an orthonormal basis for $\mathbb{C}^N$ of the form

$$\{R^{2k}u\}_{k=0}^{N/2-1} \cup \{R^{2k}v\}_{k=0}^{N/2-1}, \qquad [3]$$

for some $u, v \in \mathbb{C}^N$. The operator R shifts elements of a vector by one place as follows: $Rz = [z(N-1), z(0), z(1), \ldots, z(N-2)]^T$. Note that R^j shifts elements by j places; we call $R^j z$ the translate of z by j. So a first-stage wavelet basis consists of the even translates of u and v, which are called the generators, or sometimes the father and mother wavelets, respectively. In order to generate an orthonormal basis, we require that u, v, and their translates be mutually orthonormal,

$$\langle u, R^{2k}u\rangle = \begin{cases} 1, & k = 0 \\ 0, & k = 1, 2, \ldots, N/2-1 \end{cases}, \qquad [4]$$

$$\langle v, R^{2k}v\rangle = \begin{cases} 1, & k = 0 \\ 0, & k = 1, 2, \ldots, N/2-1 \end{cases}, \qquad [5]$$

$$\langle u, R^{2k}v\rangle = 0,\ k = 0, 1, \ldots, N/2-1. \qquad [6]$$

These constraints are equivalent to

$$|\hat{u}(n)|^2 + |\hat{u}(n+N/2)|^2 = 2,\ n = 0, 1, \ldots, N/2-1, \qquad [7]$$

$$|\hat{v}(n)|^2 + |\hat{v}(n+N/2)|^2 = 2,\ n = 0, 1, \ldots, N/2-1, \qquad [8]$$

$$\hat{u}(n)\overline{\hat{v}(n)} + \hat{u}(n+N/2)\overline{\hat{v}(n+N/2)} = 0,\ n = 0, 1, \ldots, N/2-1. \qquad [9]$$

Here, $\hat{}$ denotes the DFT of a signal, and $\hat{z}(m)$ is the m^{th} component of $\hat{z}$, given by $\hat{z}(m) = \sum_{n=0}^{N-1} z(n)e^{-2\pi imn/N}$. Formulating the constraints in the Fourier domain makes it clear that we may select u to contain only low-frequency components and v to contain only high-frequency components (or vice versa). Many common wavelet generators are constructed in the Fourier domain because Eq. (7)–Eq. (9) make satisfying the orthonormality constraints easy. Standard notation has u contain the low frequencies and v contain the high frequencies.

One may wonder, why construct an orthonormal basis from even translates of two vectors instead of all the translates of a single vector? One may show that $\{R^k w\}_{k=0}^{N-1}$ is an orthonormal basis for $\mathbb{C}^N$ if and only if $|\hat{w}(n)| = 1\ \forall\, n \in \mathbb{Z}_N$. In words, a basis of this form has no frequency localization.

Given u, we can construct v (or vice versa). Suppose $\{R^{2k}u\}_{k=0}^{N-1}$ is an orthonormal set. Define v by

$$v(k) = (-1)^{k-1}\overline{u(1-k)} \quad \forall\, k. \qquad [10]$$

Then one can check that $\{R^{2k}u\}_{k=0}^{N/2-1} \cup \{R^{2k}v\}_{k=0}^{N/2-1}$ is indeed a first-stage wavelet basis.

Once we have a first-stage wavelet basis, we can calculate the coordinates of $z \in \mathbb{C}^N$ in this basis quickly using convolutions by noting that $\langle z, R^{2k}v\rangle = z * \tilde{v}(2k)$, and similarly for u. Here, the convolution $z * w \in \mathbb{C}^N$ is the vector with components

$z * w(m) = \sum_{n=0}^{N-1} z(m-n)w(n)\,\forall\, m$, and the $\tilde{}$ denotes conjugate reflection: for any $w \in \mathbb{C}^N$, define $\tilde{w} \in \mathbb{C}^N$ by $\tilde{w}(n) = \overline{w(-n)} = \overline{w(N-n)}\,\forall\, n$. Convolutions are quick to compute because $z * w = (\hat{z}\hat{w})\check{}$: we perform elementwise multiplication of the DFTs of z and w, and then take the inverse discrete Fourier transform (IDFT) of the result, denoted by $\check{}$. For $w \in \mathbb{C}^N$, $\check{w} \in \mathbb{C}^N$ is defined as the vector whose n^{th} entry is $\check{w}(n) = \frac{1}{N}\sum_{m=0}^{N-1} w(m)e^{2\pi imn/N}$. Thus, we can calculate the coordinates of z in a first-stage wavelet basis quickly by two convolutions of z with $\tilde{u}$ and $\tilde{v}$, followed by throwing out the odd-indexed terms, which we call downsampling. The downsampling operator, D, is defined formally as follows. Suppose $M \in \mathbb{N}$ and $N = 2M$. Define $D : \mathbb{C}^N \to \mathbb{C}^M$ by setting, for $z \in \mathbb{C}^N$, $D(z)(n) = z(2n)$ for $n = 0, 1, \ldots, M-1$.

To recover the original signal from its first-stage wavelet coordinates, we upsample, convolve with u and v, and add the results. The upsampling operator, $U : \mathbb{C}^M \to \mathbb{C}^{2M}$, is defined by setting $U(z)(n) = z(n/2)$ for n even, and 0 for n odd. The forward and inverse transforms are shown schematically in Figure S1, where "$\downarrow 2$" and "$\uparrow 2$" denote downsampling and upsampling, respectively.

A. Iteration step. The arrangement in Figure S1 suggests the possibility for iteration. In standard wavelet analysis, the same convolve-downsample and upsample-convolve steps are performed only on the lower branch containing the lower frequencies; a two iteration example is shown in Figure S2. One motivation for this choice is that it is often natural to think of frequencies on a logarithmic scale (e.g., in music, and even in turbulence). One could iterate on both branches, but we will follow convention and iterate only on the lower branch.

When N is divisible by 2^p, we may perform p iterations, which yields a p^{th}-stage wavelet basis. At each stage l, we require vectors $u_l, v_l \in \mathbb{C}^{N/2^{l-1}}$ satisfying the constraints Eq. (7)–Eq. (9) (as before, v_l can be automatically constructed from u_l by Eq. (10), and vice versa). We denote the coefficients output at each stage by $x_l, y_l \in \mathbb{C}^{N/2^l}$, with $x_1 = D(z * \tilde{v}_1)$, $y_1 = D(z * \tilde{u}_1)$, and the others defined inductively by $x_l = D(y_{l-1} * \tilde{v}_l)$ and $y_l = D(y_{l-1} * \tilde{u}_l)$. The output of the forward p^{th}-stage wavelet transform is the set of vectors $\{x_1, x_2, \ldots, x_p, y_p\}$. Note that this set has a total of N numbers, so there is no lost or redundant information.

The recursive description is useful for algorithmic purposes, but there is an equivalent nonrecursive formulation which gives us more insight. Define

$$f_1 = v_1, \quad g_1 = u_1. \tag{11}$$

Then inductively define $f_l, g_l \in \mathbb{C}^N$ by

$$f_l = g_{l-1} * U^{l-1}(v_l), \quad g_l = g_{l-1} * U^{l-1}(u_l). \tag{12}$$

Now the vectors x_l and y_l introduced above are given by

$$x_l = D^l(z * \tilde{f}_l), \quad y_l = D^l(z * \tilde{g}_l). \tag{13}$$

Now, for $j = 1, 2, \ldots, p$ and $k = 0, 1, \ldots, N/2^j - 1$, let

$$\psi_{-j,k} = R^{2^j k} f_j, \quad \phi_{-j,k} = R^{2^j k} g_j. \tag{14}$$

Then the set of vectors

$$\{\psi_{-1,k}\}_{k=0}^{N/2-1} \cup \{\psi_{-2,k}\}_{k=0}^{N/4-1} \cup \cdots \cup \{\psi_{-p,k}\}_{k=0}^{N/2^p-1} \cup \{\phi_{-p,k}\}_{k=0}^{N/2^p-1} \tag{15}$$

is an orthonormal basis for $\mathbb{C}^N$, and its elements are called wavelets on $\mathbb{Z}_N$. The basis Eq. (15) comprises $N/2$ translates by two of $\psi_{-1,0}$, $N/4$ translates by four of $\psi_{-2,0}$, and so on, down to $N/2^p$ translates by 2^p of $\psi_{-p,0}$, and $N/2^p$ translates by 2^p of $\phi_{-p,0}$. For compactness, we write ψ_{-l} in place of $\psi_{-l,0}$ and ϕ_{-l} in place of $\phi_{-l,0}$.

Now define the spaces $W_{-l} = \text{span}\{\psi_{-l,k}\}_{k=0}^{(N/2^l)-1}$ and $V_{-l} = \text{span}\{\phi_{-l,k}\}_{k=0}^{(N/2^l)-1}$. Then one may show that $V_{-l} \oplus W_{-l} = V_{-l+1}$, meaning that V_{-l} and W_{-l} are subspaces of V_{-l+1}, they are orthogonal to each other, and every element in V_{-l+1} can be written as a sum of some element in V_{-l} and some element in W_{-l}. We then get the picture sketched in Figure 2A in the main text (replacing $\mathbb{R}^N$ in the figure with the more general $\mathbb{C}^N$ considered here), where the arrows represent containment. This is a conceptually important picture. Beginning at the left, we break $\mathbb{C}^N$ into orthogonal subspaces V_{-1} and W_{-1}. We then break V_{-1} into orthogonal subspaces V_{-2} and W_{-2}. We proceed until the p^{th} stage, where we are left with orthogonal subspaces V_{-p} and W_{-p}.

We can interpret this recursive splitting as follows. Recall that V_{-l} is associated with u_l and W_{-l} is associated with v_l, and u_l contains low frequencies while v_l contains high frequencies. Beginning at the left, we break $\mathbb{C}^N$ into a "coarse" or "approximation" subspace (V_{-1}) and a "fine" or "detail" subspace (W_{-1}). We then progressively split the coarse subspaces into coarser and detail subspaces. Beginning at the right, we take the coarsest subspace (V_{-p}) and add some detail (W_{-p}) to it to produce the next coarsest subspace. We progressively add details to produce richer subspaces, until we finally produce $\mathbb{C}^N$. So as we go from left to right, we coarsen our view by removing details, while as we go from right to left, we sharpen our view by adding details.

Up to now, we have not required any relationship between the u_l, v_l at different stages. There is a way to construct the u_l, v_l from u_1, v_1 that will give an orthonormal basis. When this is done, we say that we have a wavelet basis with repeated filters; this is what is usually meant by "wavelets". To do so, we set

$$u_l(n) = u_{l-1}(n) + u_{l-1}(n + N/2^{l-1}), \quad \text{for } n = 0, 1, \ldots, N/2^{l-1}, \tag{16}$$

and similarly for v_l. This is part of what is called the folding lemma, since we obtain u_l by cutting u_{l-1} just before its halfway point $N/2^{l-1}$, folding that part over the first part, and summing. Iterating Eq. (16) yields that

$$u_l(n) = \sum_{k=0}^{2^{l-1}-1} u_1\left(n + \frac{kN}{2^{l-1}}\right), \quad \text{for } n = 0, 1, \ldots, N/2^{l-1}, \tag{17}$$

and similarly for v_l. This way, we only need to construct a u_1 that is mutually orthonormal with its even translates, and then we can automatically construct v_1 using Eq. (10), and the rest of the u_l and v_l using Eq. (17).

Finally, the reader will note that the hierarchical basis structure illustrated in Figure 2A of the main text is intimately linked with the factor of two between scales, as well as to the orthogonality of each wavelet basis element with respect to the version of itself shifted by two lattice points.

B. An example: Haar wavelets. To demonstrate what we have written about so far, we show the simplest wavelet basis with repeated filters: the discrete version of the Haar wavelets. We will work in $\mathbb{R}^8$.

The first step is to find the father and mother wavelets, respectively ϕ_{-1} and ψ_{-1}, which are equal to the generators, respectively u_1 and v_1. Recall that if we know one of them, we can automatically construct the other such that all the required constraints are satisfied. The Haar father wavelet is $\phi_{-1} = \left[1/\sqrt{2}, 1/\sqrt{2}, 0, 0, 0, 0, 0, 0\right]^T$; one can easily check that it has unit norm and its four translates by two are mutually orthogonal. Using Eq. (10), we automatically generate the Haar mother wavelet $\psi_{-1} = \left[-1/\sqrt{2}, 1/\sqrt{2}, 0, 0, 0, 0, 0, 0\right]^T$; one can easily check that it has unit norm and its four translates by two are mutually orthogonal, as well as orthogonal to the four translates by two of the father wavelet. One can also check that the father wavelet comprises low frequencies, while the mother wavelet comprises high frequencies. In fact, the mother wavelet has a mean of zero; this is actually imposed for wavelets on $\mathbb{R}$.

Taking the inner product of a vector $z \in \mathbb{R}^8$ with the father wavelet and its translates produces local averages of z, making it clear that the subspace spanned by $\{\phi_{-1,k}\}_{k=0}^3$ is a "coarse" or "approximation" subspace. Taking the inner product of z with the mother wavelet and its translates produces local differences of z, making it clear that the subspace spanned by $\{\psi_{-1,k}\}_{k=0}^3$ is a "fine" or "detail" subspace; it provides the details that are filtered out of the approximation subspace.

Continuing on to the next stages, we use Eq. (16) to automatically generate the u_l and v_l. In Figure S3, we show the subspace view of the Haar wavelets (analogous to Figure 2A in the main text). V_{-1} is spanned by $\{\phi_{-1,k}\}_{k=0}^3$, and W_{-1} is spanned by $\{\psi_{-1,k}\}_{k=0}^3$. We then break down V_{-1} into V_{-2} and W_{-2}, respectively spanned by $\{\phi_{-2,k}\}_{k=0}^1$ and $\{\psi_{-2,k}\}_{k=0}^1$. Finally, V_{-2} is broken down into V_{-3} and W_{-3}, respectively spanned by ϕ_{-3} and ψ_{-3}. As we move to later stages, the approximation subspaces become progressively coarser. As we move to earlier stages, we add progressively finer details to produce progressively richer subspaces. The later stages contain large-scale features, and the earlier stages contain small-scale features; we will make much use of this terminology.

Finally, notice that ϕ_{-l} and ψ_{-l} are respectively dilations by two (properly normalized) of ϕ_{-l+1} and ψ_{-l+1}. This perfect self-similarity is unusual for discrete wavelets of finite length due to boundary effects. In Figure S4, we show an example of wavelets (the Daubechies-2 wavelets (2)) that are not simply rescaled dilations of wavelets from the previous stage (but nearly are), and see that the departure from simple dilation increases as the width of the wavelet increases. Finally, we note that wavelets on the unbounded domain $\mathbb{R}$ are constructed such that wavelets at different stages are exactly rescaled dilations of each other.

2. Computing the data-driven wavelet decomposition

With the above standard material as background, we now describe our method for constructing a wavelet basis from an ensemble of data. Suppose we are given a dataset whose elements are in $\mathbb{R}^N$. When we split $\mathbb{R}^N$ into the approximation and detail subspaces V_{-1} and W_{-1}, some fraction of the energy of the dataset will be contained in V_{-1}, and the rest in W_{-1}, since $\mathbb{R}^N = V_{-1} \oplus W_{-1}$. By energy, we mean the squared norm. Typically, the most energetic features of a dataset are large in scale, i.e., coarse, so they will be contained in the approximation subspace. This motivates the following sense of optimality: we would like to find the wavelet that maximizes the fraction of a dataset's energy in the large scales. In spirit, this approach is very similar to PCA, but it has the additional structure of the discrete wavelet framework. Additionally, we will encourage the wavelet basis elements to be localized.

A. The optimization problem. We now state the mathematical problem. Given a dataset $\{z_i\}_{i=1}^M \in \mathbb{R}^N$, where N is divisible by 2, we begin by finding a wavelet generator $u \in \mathbb{R}^N$ such that the coarse reconstruction error is minimized, subject to a penalty on the spread of the wavelet. This problem can be posed as

$$\min_u \quad \frac{1}{\sum_{i=1}^M \|z_i\|^2} \sum_{i=1}^M \|z_i - \sum_{k=0}^{N/2-1} \langle z_i, R^{2k}u\rangle R^{2k}u\|^2 + \lambda^2 \mathrm{Var}(u) \tag{18}$$

$$\text{s.t.} \quad \langle u, R^{2k}u\rangle = \begin{cases} 1, & k = 0 \\ 0, & k = 1, \ldots, N/2-1 \end{cases}. \tag{19}$$

The first term in the objective function is the normalized squared reconstruction error of the data when it is projected onto V_{-1}, or equivalently the data's normalized energy contained in W_{-1}. By normalizing it, the first term is bounded between 0

and 1. The second term is the variance of the wavelet generator u, multiplied by a penalization factor λ^2 that encourages the computed wavelets to be localized. As we will show next, the variance is also bounded between 0 and 1. Our normalization makes the two terms the same order of magnitude, and λ^2 sets the balance between them in the objective function.

Because the domain is periodic, the definition of the variance on the real line will not work. Since u has unit norm, squaring its values gives a probability mass function p, with $p(k) = u(k)^2$. We imagine the domain to be the unit circle, broken into N equal segments. Each segment on the unit circle corresponds to a point $(x, y) = (\cos\theta, \sin\theta)$ in the Cartesian plane, and the mean is $(\overline{x}, \overline{y}) = (\overline{r}\cos\overline{\theta}, \overline{r}\sin\overline{\theta})$, with

$$\overline{x} = \sum_{k=0}^{N-1} \cos\left(\frac{2\pi k}{N}\right) p(k), \qquad \overline{y} = \sum_{k=0}^{N-1} \sin\left(\frac{2\pi k}{N}\right) p(k). \tag{20}$$

(Think of the unit circle as a hoop with N segments whose masses are given by the probability mass function. Then $(\overline{x}, \overline{y})$ is the center of mass of the hoop.) The radius $\overline{r}$ gives a measure of the tightness of the distribution, and $0 \leq \overline{r} \leq 1$. In fact, the variance on a periodic domain (called the circular variance) is defined as $1 - \overline{r}$ (3). Explicitly, the variance is

$$\mathrm{Var}(u) = 1 - \sqrt{\overline{x}^2 + \overline{y}^2} = 1 - \sqrt{\left[\sum_{k=0}^{N-1} \cos\left(\frac{2\pi k}{N}\right) u(k)^2\right]^2 + \left[\sum_{k=0}^{N-1} \sin\left(\frac{2\pi k}{N}\right) u(k)^2\right]^2}. \tag{21}$$

We can formulate the optimization problem in terms of matrices. Let $Z = [z_1 \ldots z_M]$ contain the data as columns. Then the minimization problem is equivalent to the following maximization problem,

$$\max_u \qquad u^T A u - \lambda^2 \mathrm{Var}(u) \tag{22}$$

$$\text{s.t.} \qquad \langle u, R^{2k} u\rangle = \begin{cases} 1, & k = 0 \\ 0, & k = 1, \ldots, N/2 - 1 \end{cases}, \tag{23}$$

where

$$A = \frac{1}{||Z||_F^2} \sum_{k=0}^{N/2-1} R^{-2k} Z Z^T R^{2k} = \frac{1}{||Z||_F^2} \sum_{k=0}^{N/2-1} (R^{-2k} Z)(R^{-2k} Z)^T. \tag{24}$$

Note that $(R^{2k})^T = R^{-2k}$. The matrix A is symmetric, and for statistically homogeneous data it is also circulant, in which case its eigenvectors are discrete Fourier modes.

Aside from the variance penalty, this formulation is now much like PCA. In PCA, $A = ZZ^T$, and we just require u to have unit norm. The maximizer is the dominant eigenvector of A. We then project out the component in the u direction and repeat, which is the same thing we do in DDWD.

By solving the maximization problem, we find the generator u_1 that yields the most energetic approximation subspace V_{-1}. The complementary generator v_1 is constructed from u_1 using Eq. (10). We then proceed recursively, at each stage solving an analogous maximization problem to maximize the energy of the data contained in that stage's approximation subspace. The data matrix Z used in stage l comes from convolving the data used in stage $l-1$ with $\tilde{u}_{l-1}$ and downsampling, following Figure S2. At each stage, the N that appears in the orthonormality constraints and definition of A is the dimension of the data vectors at that stage, i.e., it is halved as we move from one stage to the next. The u_l at each stage l is the result of a maximization problem (v_l follows automatically from u_l), and there is no predetermined relationship between the u_l across stages, in contrast to traditional wavelets that use repeated filters. That is, Eq. (16) and Eq. (17) are not imposed upon the wavelets obtained with DDWD.

It is worth emphasizing that we work directly with the generators $\{u_l\}$, not the wavelets $\{\phi_{-l}\}$ and $\{\psi_{-l}\}$. We do so because the recursive formulation leads to fast transform algorithms with $O(N \log_2 N)$ complexity, an improvement over the $O(N^2)$ complexity of direct methods. This is directly analogous to the FFT algorithm. Although the variance penalty is imposed directly on the generators, Section 1A shows that the wavelets are constructed by repeated convolutions of the generators, so localized generators yield localized wavelets.

Finally, we address the issue of how λ should change at each stage. In conventional wavelets (as described in Section 1A), the variance of u_l is approximately four times that of u_{l-1}, with the factor being closer to four the more localized u_{l-1} is. To see why this is so, recall that the variance of u is given

$$\mathrm{Var}(u) = 1 - \sqrt{\left[\sum_{k=0}^{N-1} \cos\left(\frac{2\pi k}{N}\right) u(k)^2\right]^2 + \left[\sum_{k=0}^{N-1} \sin\left(\frac{2\pi k}{N}\right) u(k)^2\right]^2}. \tag{25}$$

Assuming that u is compact and concentrated near $k = 0$, we Taylor expand in $\epsilon = 1/N$. To leading order, the variance is given by

$$\mathrm{Var}(u) = 2\pi^2 \epsilon^2 \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} j(j-k) u(j)^2 u(k)^2. \tag{26}$$

Since $u(j)^2 u(k)^2 \geq 0$, the sum is non-negative (it is zero only when u has one non-zero entry). Since u_l is equal to the first half of u_{l-1}, its variance (to leading order) is given by the same formula with the same values, except N is cut in half, i.e., ϵ is doubled. Based on the leading order expansion, the variance of u_l will be a factor of four greater than that of u_{l-1}. Motivated by this property of conventional wavelets, in DDWD we decrease λ^2 by a factor of four from one stage to the next, maintaining a consistent level of penalization against the variance across stages.

B. Solving the constrained maximization problem. Employing the method of Lagrange multipliers yields a necessary condition for a local optimum to the constrained maximization problem without an obvious solution. Consequently, we reformulate the problem to make it amenable to gradient-based optimization.

Recall that the orthogonality constraints can be stated in the Fourier domain as

$$|\hat{u}(k)|^2 + |\hat{u}(k+N/2)|^2 = 2 \quad \text{for } k = 0, 1, \ldots, N/2-1. \tag{27}$$

Let $\hat{u}(k) = r_k e^{i\theta_k}$. Since u is real and N is even, we know that

$$\theta_0 = \theta_{N/2} = 0 \text{ or } \pi, \tag{28}$$

$$\theta_{N-k} = -\theta_k \quad \text{for } k = 1, \ldots, N/2-1, \tag{29}$$

$$r_{N-k} = r_k \quad \text{for } k = 1, \ldots, N/2-1. \tag{30}$$

Substituting the polar representation for $\hat{u}(k)$ into the constraints gives

$$r_k^2 + r_{N/2-k}^2 = 2 \quad \text{for } k = 0, 1, \ldots, N/2-1. \tag{31}$$

These constraints are quadratic in r_k. Notice that r_k and $r_{N/2-k}$ are constrained to lie on a circle of radius $\sqrt{2}$ (actually on the upper-right quadrant of the circle since $r_k, r_{N/2-k} \geq 0$). We can, therefore, replace these constraints by defining γ_k such that $r_k = \sqrt{2}\cos\gamma_k$ and $r_{N/2-k} = \sqrt{2}\sin\gamma_k$, and the constraints become $0 \leq \gamma_k \leq \pi/2$. In fact, we can remove these constraints on γ_k; this allows for negative values of r_k, which creates redundancies (i.e., $r_k e^{i\theta_k} = -r_k e^{i(\theta_k+\pi)}$), but simplifies the optimization task since it becomes unconstrained.

Because u is real, some of the constraints are redundant. For example, $r_k^2 + r_{N/2-k}^2 = 2$ gives the same constraint for $k = 1$ and $k = N/2-1$. The following are the non-redundant constraints,

$$N \text{ divisible by 4:} \quad \begin{cases} r_k^2 + r_{N/2-k}^2 = 2 & \text{for } k = 0, \ldots, N/4-1 \\ r_{N/4} = 1 \end{cases} \tag{32}$$

$$N \text{ not divisible by 4:} \quad r_k^2 + r_{N/2-k}^2 = 2 \quad \text{for } k = 0, \ldots, \tfrac{N-2}{4}. \tag{33}$$

Therefore, to find the optimal u, we only have to optimize γ_k for $k = 0, \ldots, N/4-1$ if N is divisible by 4, or until $(N-2)/4$ if N is not divisible by 4, and θ_k for $k = 1, \ldots, N/2-1$. We do so using gradient-based optimization, and the constraints are automatically satisfied because of our change of coordinates.

Besides turning the constrained optimization problem into an unconstrained one, another nice feature of this formulation is that it allows us to directly impose sparsity in the frequency domain (by setting certain γ_k equal to 0 or $\pi/2$, although we do not pursue this avenue in the present work). It also allows us to force the wavelets $\psi_{-l,k}$ to have zero mean by setting $\gamma_0 = 0$, as for wavelets on $\mathbb{R}$, but we will generally not enforce this. We note that the first term in the objective function, $u^T A u$, is not convex in the optimization variables $\{\gamma_k\}$ and $\{\theta_k\}$, so we generally find local optima. For all the results shown in this work, we have performed several trials with random initial guesses for the optimization variables. We have found the values of the objective function to be consistent across trials, suggesting that bad local optima may not be a problem.

3. Parallels with convolutional neural networks

It is well known that PCA is equivalent to a linear autoencoder, in the sense that a linear autoencoder learns to span the same subspace as PCA (4). However, PCA provides additional structure and knowledge, namely an orthogonal basis for the subspace, the energy associated with each basis element, and each basis element satisfies an optimality principle; these features make the results of PCA highly interpretable.

Similarly, the DDWD and its inverse are together equivalent to a linear convolutional autoencoder with special structure. Figure S5 shows how the discrete wavelet transform and its inverse can be realized as neural networks. In the DDWD, the generators are learned from data. The forward transform may be thought of as an encoder, and the inverse transform may be thought of as a decoder. The encoder and decoder consist of a series of convolutions. In between convolutions, there are downsampling or upsampling steps, akin to the pooling steps used in convolutional neural networks. In the encoder, the result of a layer is directly fed to the final output, and in the decoder, different parts of the input are directly fed to different layers; this is akin to skip connections. Notice that the neural network architectures induced by the forward and inverse transforms incorporate three features that have been found to yield good results in the neural network literature: convolutions, pooling, and skip connections.

We also point out some differences. The major difference is that the DDWD is linear. Another difference is that the DDWD is trained quite differently from convolutional autoencoders. In particular, we sequentially maximize several layer-wise objective

functions, learning one layer at a time, rather than learning all filters at once by maximizing a single objective function. As in PCA, the basis elements learned by the DDWD are orthogonal, energy can be easily associated with each basis element, and each stage satisfies an optimality principle, altogether making the results highly interpretable. We hope that the network architecture induced by the discrete wavelet transform, and the training process of the DDWD, will inspire developments in convolutional neural networks and autoencoders.

4. Additional results

Here we present additional results for the Kuramoto-Sivashinsky and homogeneous isotropic turbulence datasets. Since it will be useful in understanding the similarity results, we first describe how to produce the action of the similarity/dilation operator, which we denoted S in the main text, on a wavelet.

A. Similarity/dilation. Following Section 1A, we may develop explicit relations for the wavelets,

$$\psi_{-l} = u_1 * U(u_2) * U^2(u_3) * \ldots * U^{l-2}(u_{l-1}) * U^{l-1}(v_l), \quad [34]$$
$$\phi_{-l} = u_1 * U(u_2) * U^2(u_3) * \ldots * U^{l-2}(u_{l-1}) * U^{l-1}(u_l). \quad [35]$$

To be clear, $\psi_{-1} = v_1$ and $\phi_{-1} = u_1$. The similarity-transformed wavelets are produced by applying the folding lemma Eq. (16) to produce the next wavelet generator, that is,

$$S\psi_{-l} = u_1 * U(u_2) * U^2(u_3) * \ldots * U^{l-2}(u_{l-1}) * U^{l-1}(u_l) * U^l(F_l(v_l)), \quad [36]$$
$$S\phi_{-l} = u_1 * U(u_2) * U^2(u_3) * \ldots * U^{l-2}(u_{l-1}) * U^{l-1}(u_l) * U^l(F_l(u_l)), \quad [37]$$

where the action of $F_l : \mathbb{C}^{N/2^{l-2}} \to \mathbb{C}^{N/2^{l-1}}$ is defined by Eq. (16). (Since u_l and v_l are related by Eq. (10), the folding lemma Eq. (16) may be applied to both.) Note that $S\psi_{-l}$ and ψ_{-l-1} are closely related, the only difference being that $S\psi_{-l}$ uses $F_l(v_l)$ produced by the folding lemma in place of the computed v_{l+1}; an analogous relation holds between $S\phi_{-l}$ and ϕ_{-l-1}.

B. Kuramoto-Sivashinsky. The computed wavelets and their power spectra for the Kuramoto-Sivashinsky dataset are shown for all variance penalties ($\lambda^2 = 0$, 0.01, and 0.1) in Figure S6. Note that the lowest stages (where we push the most energy) comprise the same wavenumbers no matter the variance penalty. This demonstrates the robustness of the DDWD in pushing the dominant (most energetic) length scales of the system to the lowest stages.

Figure S7 shows the energy contained in each subspace for all variance penalties. The energy curves are perceptually indistinguishable, again demonstrating the robustness of the DDWD. Note that the energy curve is non-monotonic. The reason for this non-monotonicity is that V_{-6} and W_{-6} have dimension 1, W_{-5} has dimension 2, and W_{-4} has dimension 4. In other words, the energy contained in W_{-4} is spread amongst the 4 translates of ψ_{-4}, whereas all of the energy contained in W_{-6} is attributed to ψ_{-6}, and similarly for the other subspaces.

Figure S8 shows how similar the wavelets are from stage to stage.

C. Homogeneous isotropic turbulence. For the HIT data, the computed wavelets and their power spectra are shown for all variance penalties ($\lambda^2 = 0, 10^{-1}, 10^0$, and 10^1) in Figure S9. DDWD successfully pushes the high energy low wavenumbers to the lower stages, no matter the variance penalty, demonstrating the robustness of the DDWD. As the variance penalty increases, localization in the Fourier domain is exchanged for localization in the spatial domain, and the cutoffs for each scale in the Fourier domain become more gradual. The power spectra of the finest scale wavelets are spread out in many patches, and we have found that they differ somewhat across random trials. The reason that the DDWD is not as robust to the highest wavenumbers for the HIT dataset is that they comprise a very small fraction of the dataset's energy (the energy in $k = 1000$ is more than eight orders of magnitude less than the energy in $k = 1$), near or below tolerances in the optimization algorithms used.

Figures S10 and S11 show how similar the wavelets are from stage to stage for $\lambda^2 = 10^0$ and 10^1, respectively. In particular, the localized wavelets in stages $4 \leq l \leq 8$ are nearly identical across this range of λ. This is also the range over which the wavelets show strong self-similarity from stage to stage, indicating that it is a robust feature derived from the data.

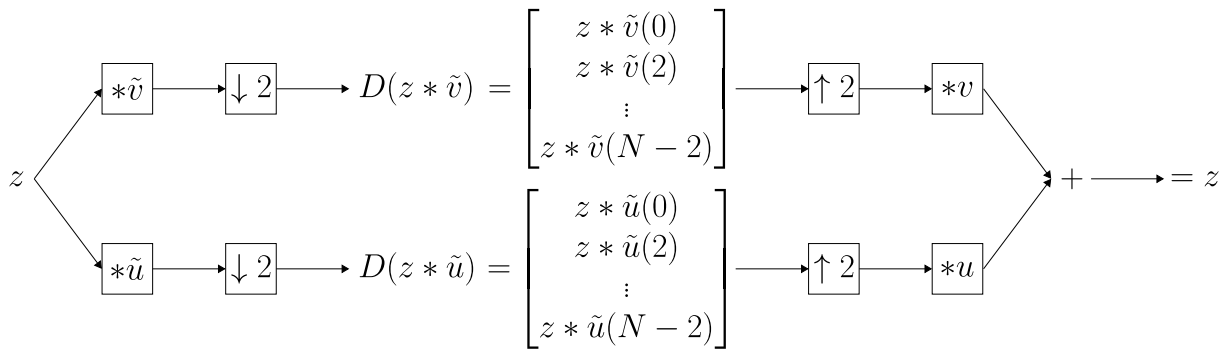

Fig. S1. Change of basis to and from a first-stage wavelet basis. The two vectors in the middle give the first-stage wavelet coordinates of z.

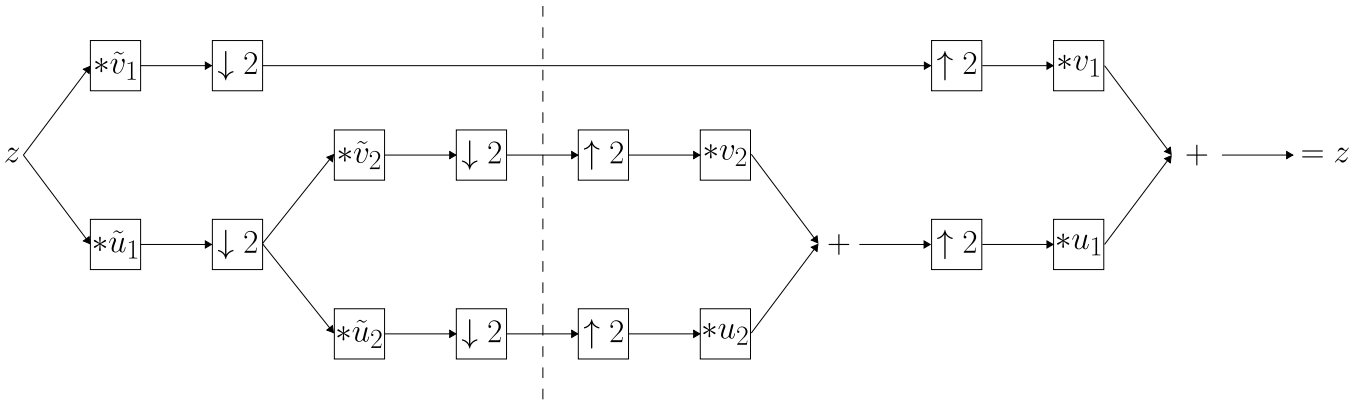

Fig. S2. Change of basis to and from a second-stage wavelet basis. The dashed line separates the forward and inverse transforms.

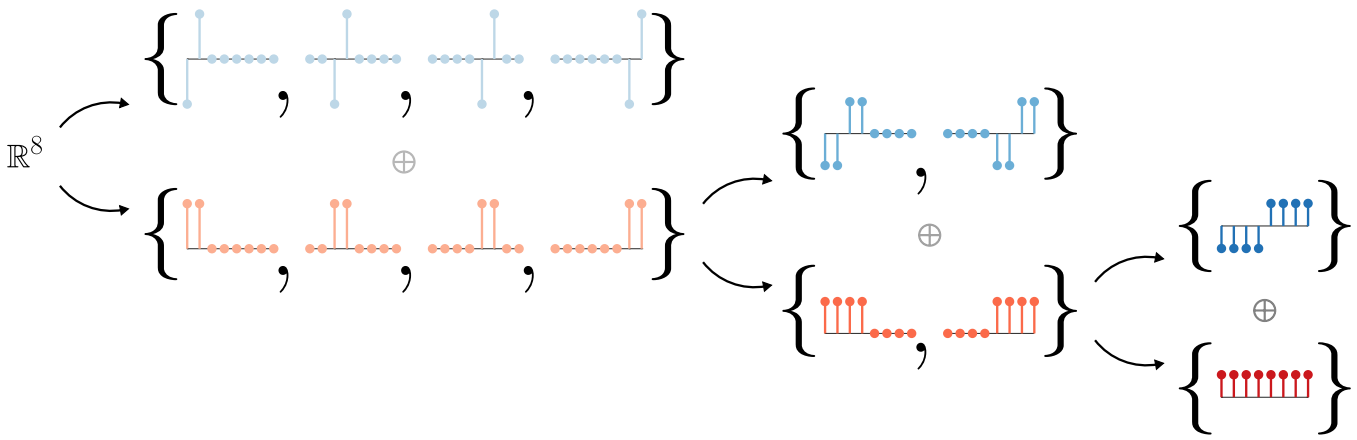

Fig. S3. Subspaces from Haar wavelets on $\mathbb{R}^8$, analogous to Figure 2A in the main text.

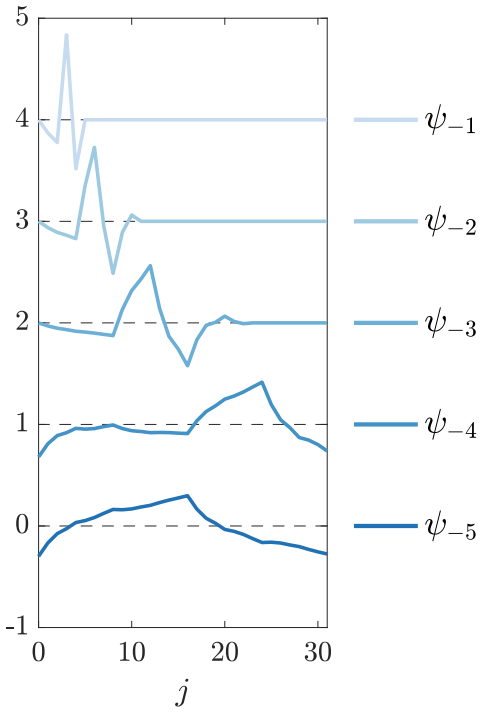

Fig. S4. Daubechies-2 wavelets on $\mathbb{R}^{2^5}$.

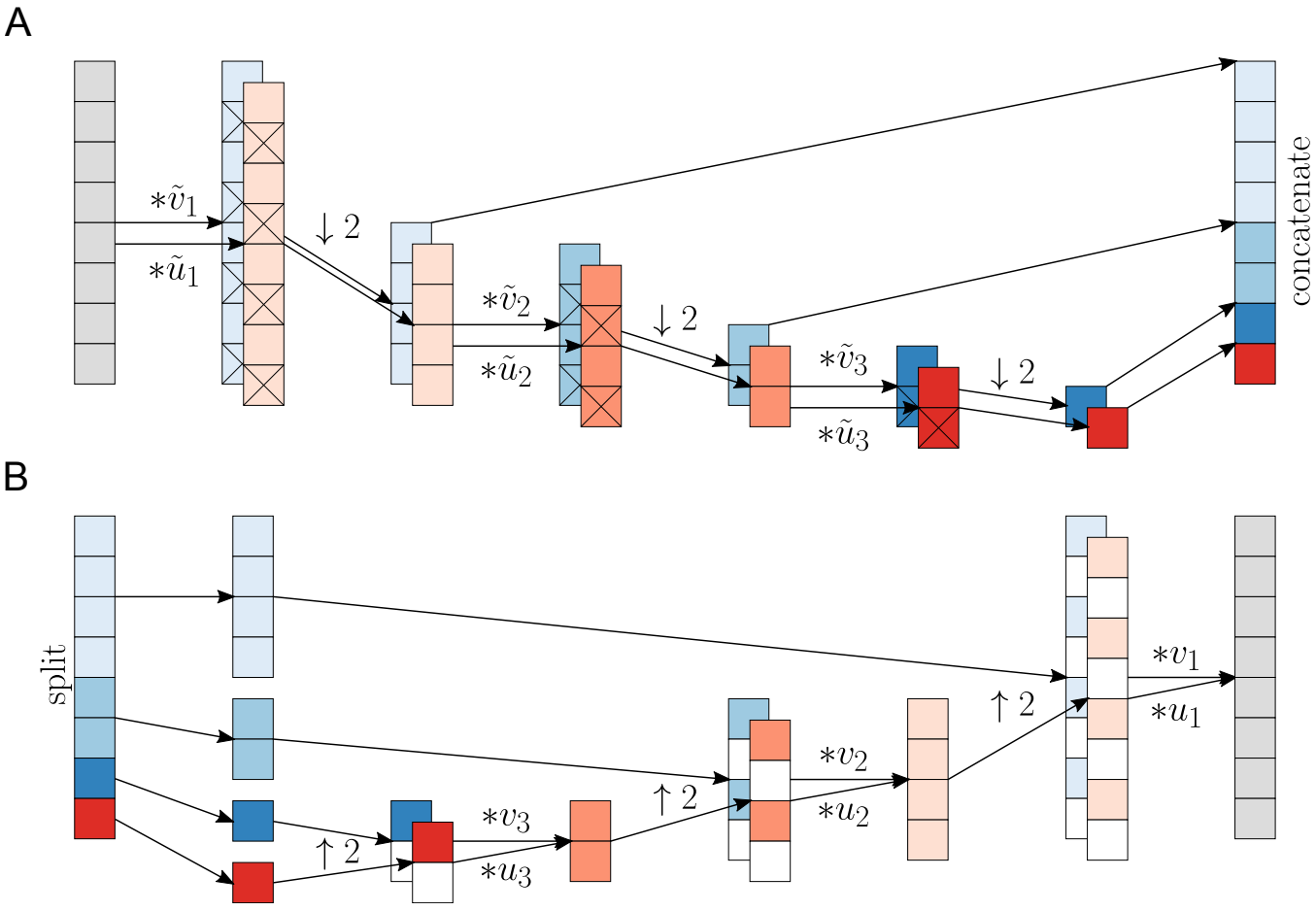

Fig. S5. Discrete wavelet transform (A) and inverse discrete wavelet transform (B) as neural networks. The example is for a signal with eight entries. For data-driven wavelets, the generators are learned from data.

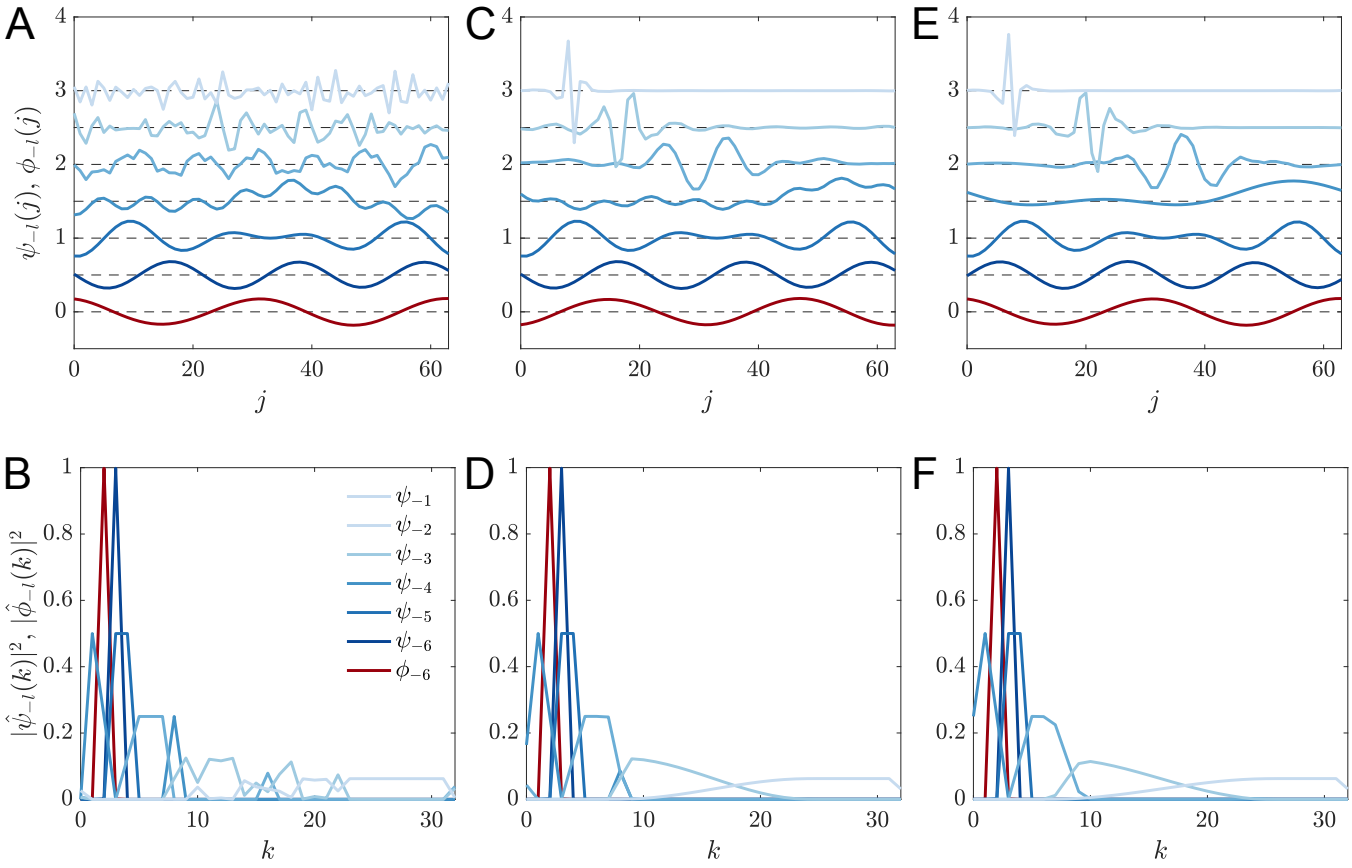

Fig. S6. Kuramoto-Sivashinsky wavelets (top row), offset from each other by 0.5, and their power spectra (bottom row). Colouring as in Figure 2A in the main text. The variance penalties are $\lambda^2 = 0$ (A–B), 0.01 (C–D), and 0.1 (E–F).

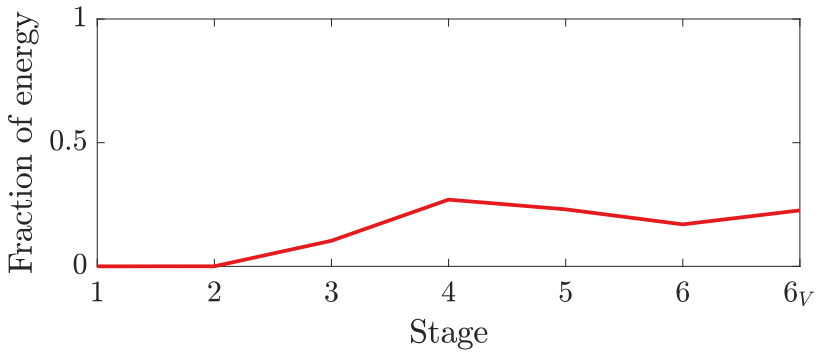

Fig. S7. Energy of the Kuramoto-Sivashinsky dataset contained in each stage for all variance penalties considered ($\lambda^2 = 0, 0.01$, and 0.1; only the result for $\lambda^2 = 0.1$ (red) can be seen).

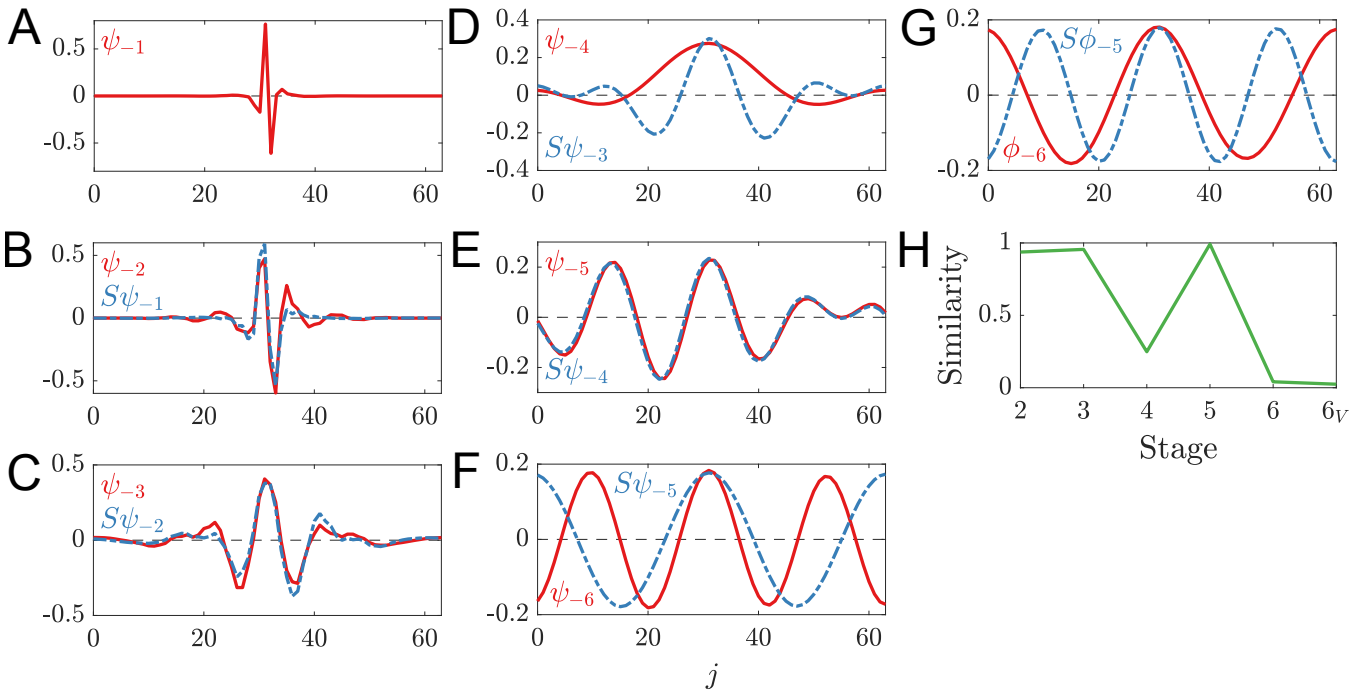

Fig. S8. Comparison between computed Kuramoto-Sivashinsky wavelets ($\lambda^2 = 0.1$) and ones obtained by dilating and rescaling the wavelet from the previous stage (A–G), and the level of similarity across all stages (H).

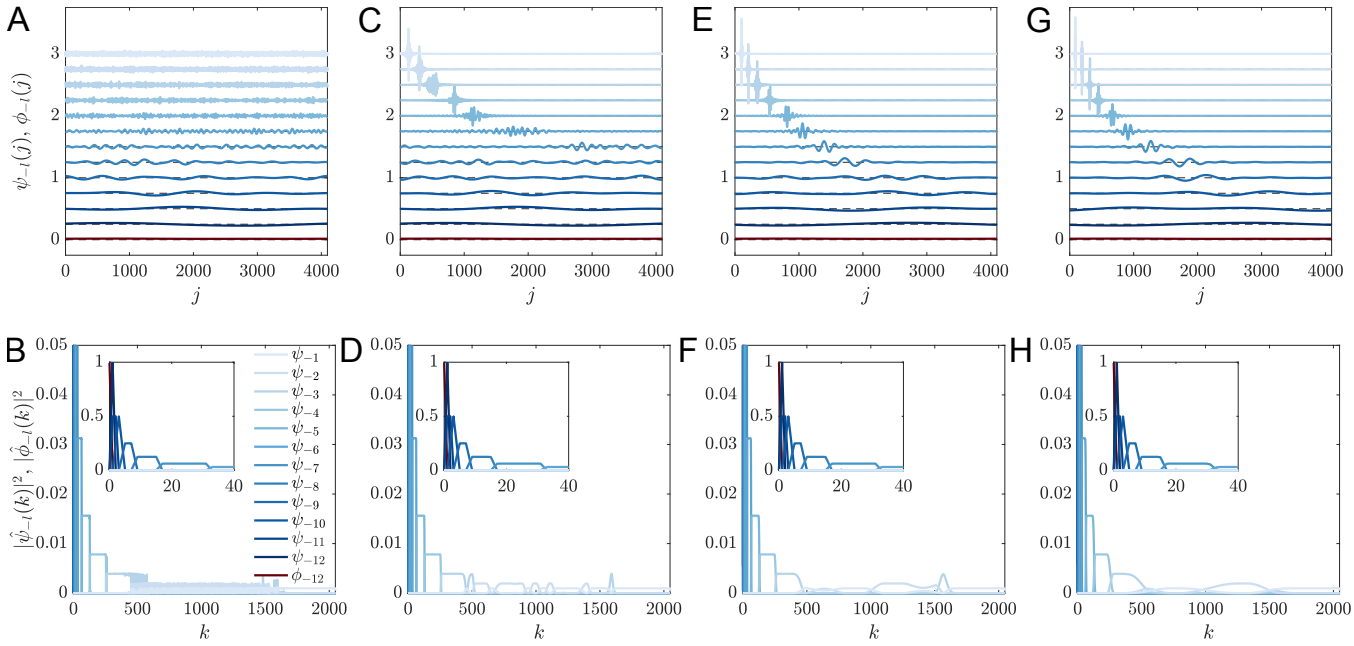

Fig. S9. HIT wavelets (top row), offset from each other by 0.25, and their power spectra (bottom row). Colouring as in Figure 2A in the main text. The variance penalties are $\lambda^2 = 0$ (A–B), 10^{-1} (C–D), 10^0 (E–F), and 10^1 (G–H).

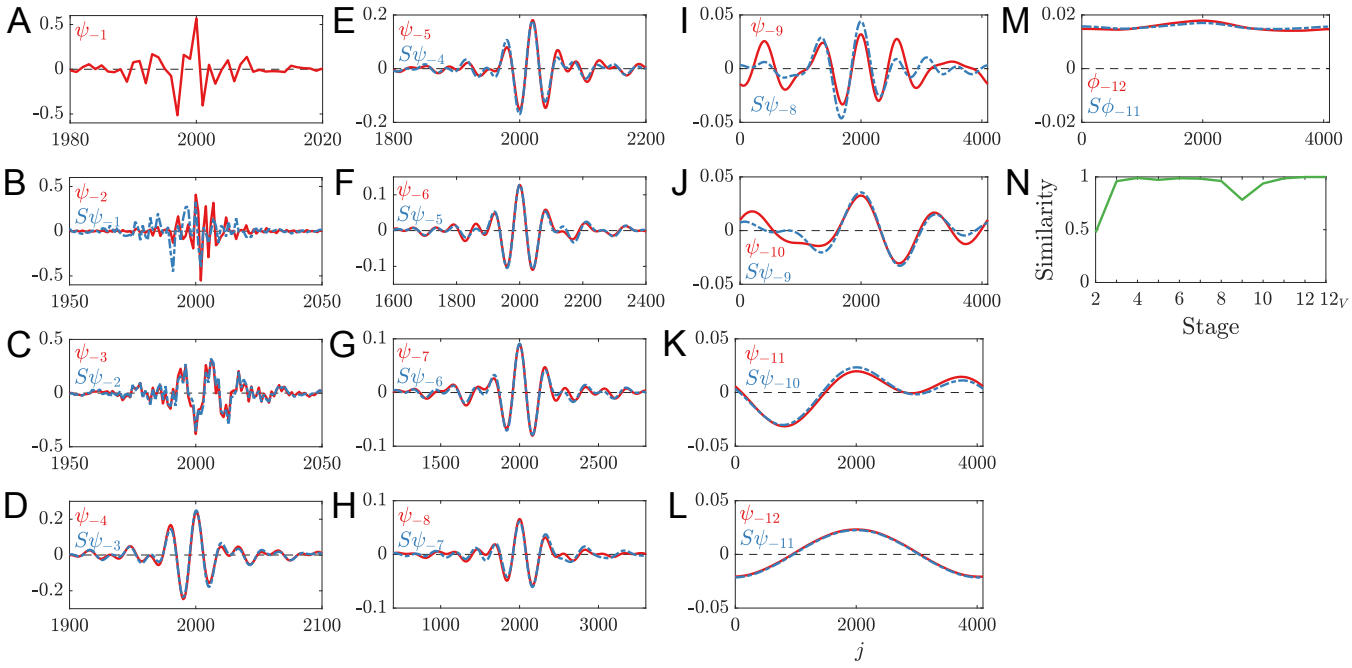

Fig. S10. Comparison between computed HIT wavelets ($\lambda^2 = 10^0$) and ones obtained by dilating and rescaling the wavelet from the previous stage (A–M), and the level of similarity across all stages (N).

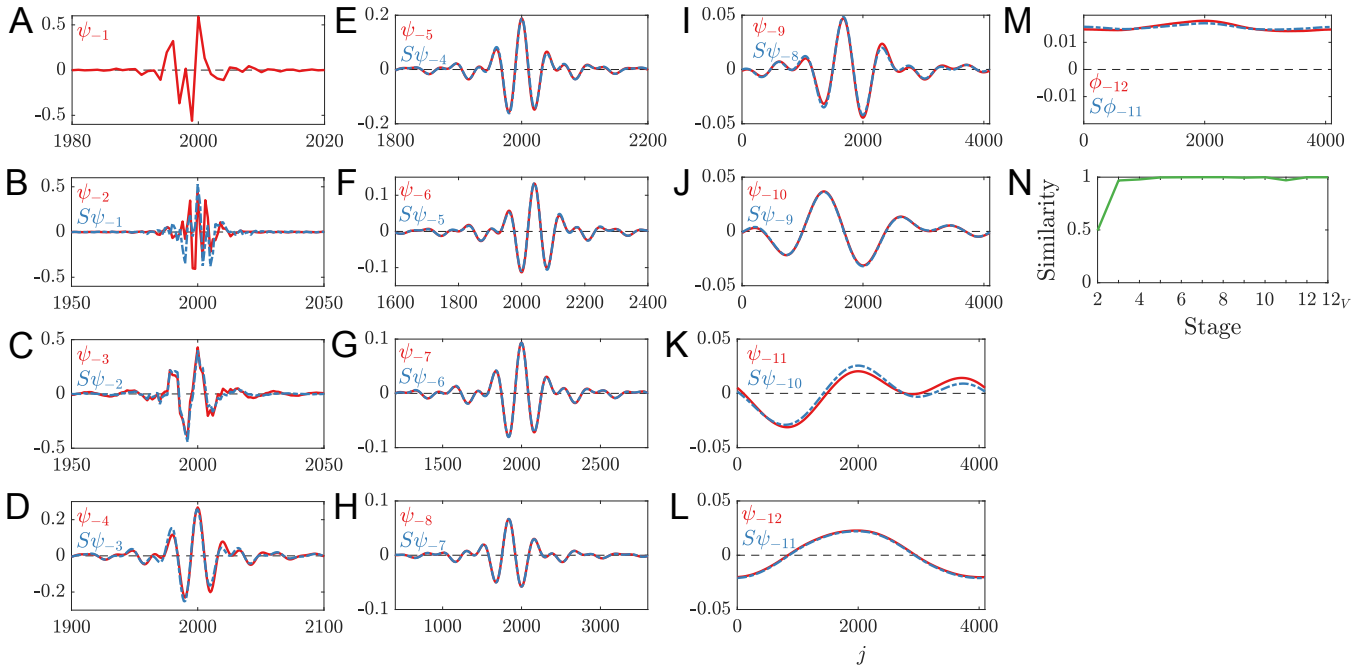

Fig. S11. Comparison between computed HIT wavelets ($\lambda^2 = 10^1$) and ones obtained by dilating and rescaling the wavelet from the previous stage (A–M), and the level of similarity across all stages (N).

References

1. Frazier MW (2006) *An introduction to wavelets through linear algebra.* (Springer Science & Business Media).
2. Mallat S (1999) *A wavelet tour of signal processing.* (Elsevier).
3. Mardia KV, Jupp PE (2009) *Directional statistics.* (John Wiley & Sons) Vol. 494.
4. Goodfellow I, Bengio Y, Courville A (2016) *Deep learning.* (MIT Press).