Flow Field Modulation

Bo Ren, Chen-Feng Li, Ming Lin, Theodore Kim, and Shi-Min Hu

Abstract

The nonlinear and non-stationary nature of Navier-Stokes equations produces fluid flows that can be noticeably different in appearance with subtle changes. In this paper we introduce a method that can analyze the intrinsic multiscale features of flow fields from a decomposition point of view, by using the Hilbert-Huang transform method on 3D fluid simulation. We show how this method can provide insights to flow styles and help modulate the fluid simulation with its internal physical information. We provide easy-to-implement algorithms that can be integrated with standard grid-based fluid simulation methods, and demonstrate how this approach can modulate the flow field and guide the simulation with different flow styles. The modulation is straightforward and relates directly to the flow's visual effect, with moderate computational overhead.

Index Terms

Physically Based Animation, Fluid Simulation, Flow Style, Hilbert-Huang Transform, Fourier, Wavelet

- *B. Ren and S.M. Hu are with the Department of Computer Science, Tsinghua University, Beijing 100084, China.*
- *C.F. Li is with the College of Engineering, Swansea University, Swansea SA2 8PP, UK.*
- *M. Lin is with the Department of Computer Science, University of North Carolina, Chapel Hill, NC 27599-3175.*
- *T. Kim is with the Media Arts and Technology, University of California, Santa Barbara.*

Flow Field Modulation

1 Introduction

Over the last decade, tremendous success has been achieved in solving the nonlinear Navier-Stokes equations for realistic and stunning visual effects of fluid flows. It is well known that practical fluid flows have intrinsic multiscale structures, especially when turbulence occurs. Although it can be understood intuitively that different levels of flow features, e.g. large and small eddies in a smoke plume, should be related to their different frequency ranges over the space, standard data analysis tools such as Fourier and wavelet decomposition cannot effectively catch the frequency information because the flow field is highly nonlinear and non-stationary. We address this issue and provide a tool to analyze and adjust multiscale features of flow fields during simulation.

In this paper, we present a method that can provide, during the simulation process, information for the intrinsic multiscale features of flow fields which can then be analyzed, manipulated and integrated into the physical simulation. We also define the concept of flow styles, and demonstrate how to achieve flexible feature enhancement and stabilization over all scales. To do this, the complete spatial Hilbert-Huang transform (SHHT) method is introduced. The first part of SHHT is the empirical mode decomposition (EMD), through which the fluid velocity field is decomposed into a series of components with a natural frequency-separating feature. The second part of SHHT is the Hilbert transform of these natural components, which provides the frequency information explicitly. These lead to our main contributions listed as below:

1) Introduce the complete SHHT method to extract intrinsic frequency information of flow fields.
2) Define quantitatively the concept of flow styles.
3) Achieve controllable flow field modulation to enhance and stabilize intrinsic multiscale flow features.
4) Propose an easy-to-implement modulation framework that can be readily integrated into existing grid-based fluid solvers.

2 Previous Works

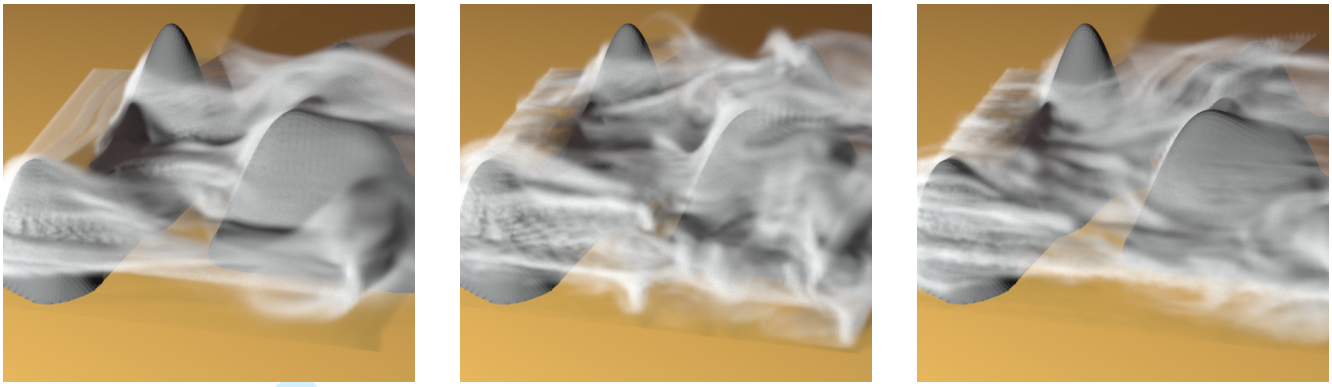
Grid-based fluid simulation has been popular in the graphics community since Jos Stam's unconditionally stable solver [1]. Many different techniques have been introduced to handle problems encountered in solving Navier-Stokes equations under limited resolution. Vorticity methods such as [2] and [3] use the vorticity confinement technique to prevent rapid dissipation of vortices. For dissipation in the advection step Zhu et al. [4] introduced FLIP. Higher order advection methods including BFECC [5], QUICK [6] and semi-Lagrangian MacCormack [7] were also developed through the years. For explosion effects, Feldman et al. [8] introduced artificial divergence sources.

There are also various approaches, both physical and non-physical, that dedicate to add richer details to fluid simulation. The octree method [9] includes an adaptive-grid scheme to save computation in some regions and provide more details in other regions. Many researchers investigated artificial turbulence. Different methods [10], [11], [12], [13] have been proposed using random noises to construct divergence-free velocity fields that have more turbulence details than the original field. Artificial noises were also used in [14], [15], [16], [17], [18] to generate turbulence, but they added the generated turbulence field to the up-sampled low-resolution field or added artificial forces into the simulation, influencing the original velocity field indirectly.

Fluid control was first introduced in [19]. References [20], [21], [22] used key frames whereas references [23], [24], [25] used target shapes to obtain desired shapes of fluid. These methods adopt extra forces or velocity constraints to form the desired shape. Rasmussen et al. [26] presented a particle-based method that applies soft or hard control on liquids, and Thurey et al. [27] preserved small-scale flow details using control particles. Mihalef et al. [28] combined 2D artworks with full 3D fluid simulation to get featured breaking waves. Several approaches are concerned with consistency of fluid appearances between high-resolution and low-resolution simulations; these methods are related to ours in that natural generation of features in fluid simulation is emphasized. Nielsen el al. [29] improved the correspondence between low and high resolutions in smoke simulations by solving an optimization problem. Nielsen el al. [30] guided the high resolution liquid simulations by its low resolution version to ensure consistency. Yuan el al. [31] used the Lagrangian coherent structure to achieve controllable pattern regulation of fluids.

The Hilbert-Huang transform [32] was first used for signal processing. In computer graphics, references [33], [34], [35] used the EMD method in image and texture synthesis. Subr et al. [36] combined EMD decomposition with morphological image filters to achieve edge-preserving image decomposition. The space filling curve technique was introduced into 3D fluid simulations in [37] recently along with some brief explanation and analysis. None of the aforementioned works applied the Hilbert transform or performed frequency analysis except for [35], which used it to compare the deviation between sample and result.

In a wider context, this study is also relevant to spectral methods in computational fluid dynamics, where



(a) (b) (c)

Fig. 1. Example 1, smoke passing through mountains, style modulation. (a) The original; (b) modulation with low-frequency motion boosted by $\alpha = 0.1$, the flow is more energetic showing significant global transformations; (c) modulation with mid-frequency motion boosted by $\alpha = 0.5$, the flow now has a wrinkled appearance but the intrinsic shape is retained.

Fourier series have been used to solve homogeneous flows in the early days [38], Gaussian filters are used in large eddy simulations to reduce the range of length scales of the solution [39], and orthogonal polynomials are increasingly employed to achieve high accuracy for locally-smooth flows [40]. However, none of these techniques addresses the problem targeted in this study, which is to extract the intrinsic spatial frequency information from turbulent flows and modulate the corresponding multiscale flow features to create different flow styles.

3 Background

In this section we first describe our initiative of flow field modulation, and then summarize the background knowledge needed for the algorithm development.

3.1 Introduction to Flow Field Modulation

Modulation methods have proven to be very effective in different scientific fields. In computer graphics, transforming 2D images into the frequency domain and applying the inverse transform after modulation is a common practice in image processing. Such different effects as edge-sharpening, feature extraction and image compression etc. can be achieved. The power of modulation relies on the fact that there are different levels of details stored in the original signal and they occupy different parts of the frequency domain. The multiscale nature of turbulent flow can be easily observed in real life, e.g. a solitary wave approaching from the distance and breaking into spray on the shore. It is desirable to be able to modulate the flow field's intrinsic multiscale structures for different visual effects, such as feature enhancement and stabilization.

There have been many methods developed to add or even design artificial features in fluid simulation, but there is no method available to analyze and modulate the intrinsic multiscale structural features generated by the flow field. This is largely due to the lack of a mathematical representation of the multiscale structure of turbulent flow, which relies on an effective decomposition method for the nonlinear and non-stationary flow field. Therefore, we aim to introduce such a method that can help analyze why one flow appearance differs from another, i.e. the concept of "flow style", and allow flexible frequency-based changing to the decomposed velocity fields. The associated modulation approach is expected to share the common benefits as in other scientific areas.

A trivial attempt is to apply standard data analysis methods, e.g. Fourier decomposition or wavelet decomposition, on the flow field. Unfortunately, the conventional methods do not perform well because (1) the flow field is highly nonlinear and non-stationary, thus unsuitable for traditional methods that feature fixed base functions and constant amplitudes for individual frequency components and (2) traditional decomposition methods cannot reveal the information of "instantaneous frequency and amplitude", which is more important in non-stationary data sets compared to predetermined frequency domains. Test examples and more detailed discussions are given in §5.2. Because of these limitations, we introduce the Hilbert-Huang transform (HHT) for our purpose. The Hilbert-Huang transform was initially designed to analyze time series that are non-stationary and nonlinear, which are unsuitable for Fourier, wavelet or Gaussian pyramid transforms in nature, and it has a natural generalization to higher dimensional spaces. Unlike conventional data analysis methods, instantaneous frequency and amplitude are well defined in HHT. We call the technique of using HHT spatially as the spatial Hilbert-Huang transform (SHHT), and will show later that the 3D SHHT can be approximated by a 1D HHT, which allow compression of 3D velocity data into 1D arrays for analysis and amplitude modulation, resulting in significant benefits in computational efficiency. The SHHT composes of two

main steps, namely the empirical mode decomposition (EMD) and the Hilbert transform. The 1D case and its advantage over traditional decomposition methods for nonlinear and non-stationary data sets were thoroughly discussed in Huang's original paper [32]. Here we briefly explain this method in the next two subsections.

3.2 Empirical Mode Decomposition

The purpose of EMD is to decompose an arbitrary data set into a finite number of so-called "intrinsic mode functions"(IMF) that are suitable for the Hilbert transform. The IMFs can be roughly considered as the counter-part of "frequency components" in traditional decomposition methods. The first fundamental difference is that the EMD does not have pre-defined base functions. Indeed, the concept of "base functions" is not important for EMD because its decomposition is self-adaptive to the data. Secondly, instead of working in the frequency domain, the IMFs are still in the signal domain, thus each IMF can be treated as a sub-data set that carries part of the original signal's information. It so happens that in the EMD progress, IMFs are extracted in the order of their frequency ranges, i.e. IMFs with higher frequency ranges are extracted first. Thus the decomposition result is self-arranged and can represent features on different scales.

Symbol	Meaning
$\mathbf{\Gamma}$,$\mathbf{\Gamma}'$	n-dimensional space coordinate
F	original n-dimensional signal
c_j	j-th IMF in the result
r_j	(temporary) j-th residual after c_j is got
h_k	k-th intermedian signal while calculating c_j
E_{max},E_{min}	upper and lower envelope of the intermedian signal
γ_k,γ'_k	the k-th dimension coordinate
X,Y,Z	n-dimensional signal functions
a	instantaneous amplitude
θ	instantaneous phase
$\mathbf{\Omega}$	instantaneous frequency
τ	1D mapped index of 3D data
ω	instantaneous frequency value in sampled MHS
$s(\omega)$	the style function
$\overline{s}(\omega)$,$s^*(\omega)$	the guiding and target style function
$\mathbf{u}$	vector velocity field value
u	component-wise scalar velocity field value
α,β	controllable strength factors in a single modulation
γ	mixing weight in consistency guiding
q	abstract symbol to represent possible physical values
t	time
$\mathbf{g}$,ρ,p	external force, density, pressure in the flow field
N	size of signal data set
δ,ϵ	constants used in EMD stopping criterion

TABLE 1
Chart of the appeared symbols

Let $F(\mathbf{\Gamma})$ denote an n-dimensional signal defined on space D, the EMD decomposition of $F(\mathbf{\Gamma})$ is:

$$F(\mathbf{\Gamma}) = \sum_{j=1}^{J} c_j(\mathbf{\Gamma}) + r_J(\mathbf{\Gamma}), \tag{1}$$

where $c_j(\mathbf{\Gamma}), j = 1, .., J$ are IMFs, and $r_J(\mathbf{\Gamma})$ the residual. The EMD progress is defined by a double-loop "sifting" algorithm that sequentially extracts the IMFs as follows:

1) Initialization $r_0 = F$, $j = 1$
2) Compute the j-th IMF, c_j
   a) Initialization $h_0 = r_{j-1}$, $k = 1$
   b) Build the upper envelope $E_{max,k-1}$, and build the lower envelope $E_{min,k-1}$ of the data set h_{k-1}
   c) Compute the mean of the upper and lower envelopes, $E_{mean,k-1} = \frac{1}{2}(E_{min,k-1} + E_{max,k-1})$
   d) $h_k = h_{k-1} - E_{mean,k-1}$
   e) If the IMF stopping criterion is satisfied, then $c_j = h_k$, else $k = k + 1$ and go to step 2(b)
3) $r_j = r_{j-1} - c_j$
4) If r_j is monotonic, the decomposition stops, else $j = j + 1$ and go to step 2

In the above decomposition progress the value of $F(\mathbf{\Gamma})$ can be scalar or vector. If the signal is vector valued, the sifting processes are performed on each of its scalar component, and this is a common practice in mathematics and physics for generalization to vector-valued fields especially in Cartesian coordinates. For example, Laplace operator is thus generalized to vector Laplace operator, as well as the gradient operator. A similar component-wise decomposition of a vector-valued field can be found in [41]. In principle, different methods can be adopted to build the upper and lower envelopes, and we will introduce a practical one in §3.4. The stopping criterion in step 2(e) will be discussed later in §4.6. The above EMD algorithm has been widely used in signal processing, and it always converges in practice.

3.3 Hilbert Transform

Applying the Hilbert transform to the sub-data sets IMFs can provide the instantaneous frequency spectrum of the original signal. A brief introduction of the Hilbert transform is given below, and we refer to [42] for further reading.

The Hilbert transform on an n-dimensional function $X(\mathbf{\Gamma})$ is

$$Y(\mathbf{\Gamma}) = \frac{1}{\pi^n} P.V. \int_{-\infty}^{+\infty} \frac{X(\mathbf{\Gamma}')}{\prod_{k=1}^{n}(\gamma_k - \gamma'_k)} d\mathbf{\Gamma}' \tag{2}$$

where $Y(\mathbf{\Gamma})$ is the result after transformation, γ_k is the k-th component of $\mathbf{\Gamma}$, and $P.V.$ indicates the integration result is the Cauchy principal value of the improper integral. Using the Hilbert transform, the instantaneous frequency of $X(\mathbf{\Gamma})$ can be computed as:

1) Define the complex signal from $X(\mathbf{\Gamma})$ and $Y(\mathbf{\Gamma})$

$$Z(\mathbf{\Gamma}) = X(\mathbf{\Gamma}) + iY(\mathbf{\Gamma}) = a(\mathbf{\Gamma})e^{i\theta(\mathbf{\Gamma})} \tag{3}$$

where $a(\mathbf{\Gamma})$ is instantaneous amplitude and $\theta(\mathbf{\Gamma})$ instantaneous phase

$$a(\mathbf{\Gamma}) = [X^2(\mathbf{\Gamma}) + Y^2(\mathbf{\Gamma})]^{\frac{1}{2}}, \tag{4}$$

$$\theta(\mathbf{\Gamma}) = arctan\left(\frac{Y(\mathbf{\Gamma})}{X(\mathbf{\Gamma})}\right). \tag{5}$$

2) Define the instantaneous frequency $\mathbf{\Omega}(\mathbf{\Gamma})$ as the total derivative of the instantaneous phase $\theta(\mathbf{\Gamma})$ over $\mathbf{\Gamma}$

$$\mathbf{\Omega}(\mathbf{\Gamma}) \cdot d\mathbf{\Gamma} = d\theta(\mathbf{\Gamma}). \quad (6)$$

Thus, after applying the Hilbert transform on each IMF, the original signal $F(\mathbf{\Gamma})$ can be written as

$$F(\mathbf{\Gamma}) = Re\left(\sum_{j=1}^{J+1} a_j(\mathbf{\Gamma}) e^{i\int \mathbf{\Omega}_j(\mathbf{\Gamma})\cdot d\mathbf{\Gamma}}\right) \quad (7)$$

where the $J+1$-th term in the above equation corresponds to the residual. The Hilbert amplitude spectrum is the distribution of $a(\mathbf{\Gamma})$ over $\mathbf{\Omega}(\mathbf{\Gamma})$ and $\mathbf{\Gamma}$, denoted as $H(\mathbf{\Omega}, \mathbf{\Gamma})$. Then, the marginal Hilbert spectrum (MHS) is defined as

$$h(\mathbf{\Omega}) = \int_D H(\mathbf{\Omega}, \mathbf{\Gamma}) d\mathbf{\Gamma}. \quad (8)$$

Similar to the EMD process, if $X(\mathbf{\Gamma})$ is vector valued, the computation is performed on each of its scalar component.

As IMFs represent sub-data sets, the integration in Eqn. (8) can be performed on a single IMF, and to compute the MHS of the original signal, one only has to add together the individual IMF results. The MHS describes the global property of the signal, and it can be considered as an unnormalized probability distribution of instantaneous frequencies over the whole data range. A higher value of MHS at a certain frequency means the probability that this instantaneous frequency appears somewhere in the original signal is higher.

3.4 Hilbert-Huang Transform for 3D Flow Fields

The Hilbert-Huang transform fits particularly well the nonlinear and non-stationary flow field data. After HHT, each data position is related to multiple instantaneous frequencies collected from the IMFs, and when applying to 3D fluid velocity fields, this gives an intuitive physical indication. For the spatially varying velocity field, we automatically obtain self-adaptive spatial frequency information grouped at each single point, corresponding to the multi-level features of the flow field.

We use the space-filling curve technique to calculate 3D envelopes of the flow field data in the EMD algorithm to reduce the high computational cost. First a 3D Hilbert space-filling curve sufficiently large to cover the whole simulation domain is built, and moving along the curve, the current grid cell is sequentially assigned with an index if and only if it is located within the simulation domain. Then, according to this index template, the data stored in 3D grids are mapped into an ordered 1D signal array. Finally, after finding all local maxima and minima of each neighboring three entries in the 1D signal array, the approximate upper and lower envelopes are constructed by connecting separately the maxima and minima with two cubic splines. This approach is extremely fast with the same computational efficiency as Huang's original algorithm. We refer the readers to the appendix in the supplemental material for more explanations and examples on usage of this technique.

We also adopt the space-filling curve technique to perform the 1D Hilbert transform on each mapped IMF, which is much faster than directly carrying out 3D Hilbert transform. It should be noted that the instantaneous frequency of a velocity component is a vector-valued function calculated in Eqn. (6) as the total derivative of the instantaneous phase over $\mathbf{\Gamma}$, while in the proposed approach, this is approximated with a directional derivative along the space-filling curve resulting in a sampled MHS. This simplification is justified in practice because: (1) the vector-valued instantaneous frequency is only introduced to measure the local variation of the data set (please refer to [42] for more discussion) and (2) the scalar result obtained using a space-filling curve can be viewed as a semi-random sampling of the total derivative with a directional derivative along a smooth path (except for limited corner points), and it still reflects to a large extent the local variation of the original data over space.

4 Modulating Flow Fields

Based on the mathematical tools described above, we first propose in this section a quantitative description of flow style, a concept that reveals intrinsic structural information of the flow field. Then, three modulation methods are introduced, providing flexible modulation over the flow style and its evolvement during simulation. Finally, we provide a simple and integrated implementation compatible with standard grid-based fluid solvers. For all algorithms in this section, velocity data are mapped from the 3D grid to 1D arrays, and we adopt the 1D version of the formulas introduced in §3.

4.1 Flow Style

Following the intuitions of "flow style" concept discussed in §3.1, a rational quantitative definition should take into account both the global trend of flow fields as well as different levels of local details. Recall the MHS defined in Eqn. (8), which is a probability distribution that contains global information over the spatial dimension and quantifies the distribution of instantaneous frequencies corresponding to different length scales. In addition, during the calculation of MHS each data element is related to several instantaneous frequencies collected from the IMFs, which correspond to different levels of local variations. For these reasons we propose to define "flow style" based on the sampled marginal Hilbert spectrum (MHS) described in §3.3 and §3.4. In our experiments we have found interesting properties of this MHS of flow fields, which we now show to provide a precise definition of "flow style".

Fig. 2(a) shows two flow fields of similar visual styles: the left figure is a single plume rising due to buoyancy forces and the right one shows multiple plumes developing in the same environmental condition. Fig. 2(e) shows

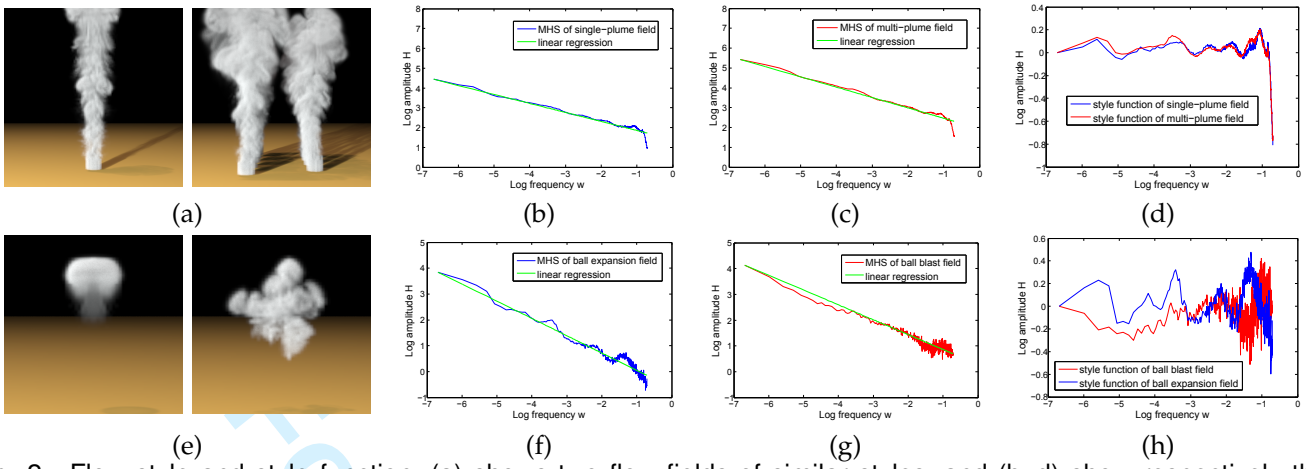


Fig. 2. Flow style and style function. (a) shows two flow fields of similar styles, and (b–d) show respectively their MHSs and style functions. (e) shows two flow fields of different styles, and (f–h) show respectively their MHSs and style functions. (enlarge to view the details)

two different visual styles: to the left is a smoke ball rising under buoyancy forces and to the right is the same ball developing in a blast environment. The MHS of these flow fields are plotted in Figs. 2(b-c) and 2(f-g), in double logarithmic coordinates. The MHS curves exhibit a clear linear relation under the logarithmic coordinate. This linear property are observed for all flow fields in our experiments, and the exponential index varies around -1 in the MHS distribution. Thus, it is more convenient to examine the deviation of MHS from the log-linear trend. Taking the difference between the MHS and its linear regression yields an oscillating deviation function around zero, as shown in Figs. 2(d) and 2(h). In Fig. 2(d) the deviation functions are computed from the flow fields in Fig. 2(a) which are of similar styles, and it can be seen that the two deviation functions are largely identical following exactly the same trend. In Fig. 2(h), the deviation functions are computed from the flow fields in Fig. 2(e) which are of different styles, and it can be seen that the associated deviation functions are also very different both in the global shape and in local characteristics. Therefore, let ω denote the instantaneous frequency computed from the flatten 1D flow field data, the style function $s(\omega)$ for a flow field is computed as follows:

1) Calculate the MHS of the velocity field.
2) Translate the MHS into its log-diagram.
3) Fit this diagram linearly.
4) Take the residual as the style function $s(\omega)$ of the flow.

The information lying in $s(\omega)$ is directly inherited from the MHS, and it reflects the actual flow field in several ways. First, the instantaneous frequency corresponds directly to the flow field's multi-level structural features such that frequency values from low to high indicates respectively large-scale transformations, medium-scale motions, and small eddies and other fine-scale details. Second, for a fixed frequency ω_0, an positive value of $s(\omega_0)$ indicates there is a higher-than-average existence of this instantaneous frequency in the flow field, which means that there are relatively more features of the corresponding scale in the flow; and vice versa. Third, the style function is completely determined by the frequency information of the flow field, which is geometry independent, and therefore one can compare flow styles between any two flows even for those evolved in completely different scenes.

4.2 Style Modulation

As the flow style measured by the style function $s(\omega)$ is essentially determined by the MHS of the velocity field, modulating flow style, i.e. to change a flow style from $s(\omega)$ to $s^*(\omega)$ is essentially to modulate the velocity field so that its MHS can meet the target. Following Eqn. (7), let

$$u(\tau) = \sum_{j=1}^{J+1} u_j(\tau) = Re\left(\sum_{j=1}^{J+1} a_j(\tau) e^{i\int \omega_j(\tau)d\tau}\right) \quad (9)$$

denote the original velocity field whose MHS is $h(\omega)$. The velocity field is vector-valued, but for simplicity reasons the scalar representation is adopted here to indicate the operation is applied independently to each velocity component. In Eqn. (9), $u_j(\tau)$ denotes the j-th IMF of the velocity field, $a_j(\tau)$ the instantaneous amplitude, $\omega_j(\tau)$ the instantaneous frequency, and the variable τ indicates the spatial index in the SHHT method. To modulate the signal so that its MHS becomes $h^*(\omega)$, one only needs to modify the amplitude

$$a_j^*(\tau) = \frac{h^*(\omega_j(\tau))}{h(\omega_j(\tau))} a_j(\tau). \quad (10)$$

It is easy to verify that the transformed signal

$$u^*(\tau) = \sum_{j=1}^{J+1} u_j^*(\tau) = Re\left(\sum_{j=1}^{J+1} a_j^*(\tau) e^{i\int \omega_j(\tau)d\tau}\right) \quad (11)$$

has an MHS as $h^*(\omega)$. Eqn. (11) represents a pure amplitude modulation because the instantaneous frequency $\omega_j(\tau)$ remains the same as in the original signal. According to Eqns. (9) and (11), the relationship between new and old velocity values due to amplitude modulation can be written as

$$\frac{u_j^*(\tau)}{u_j(\tau)} = \frac{a_j^*(\tau)}{a_j(\tau)}. \tag{12}$$

Let $\bar{s}(\omega)$ denote the guiding style function, we define the target style function $s^*(\omega)$ as

$$s^*(\omega) = s(\omega) + \alpha(\bar{s}(\omega) - s(\omega)) \tag{13}$$

where $\alpha \in (0, 1)$ is a constant tuning factor to adjust how much impact the guiding style is having on the target style. Defining the target style using Eqn. (13) instead of directly specifying $s^*(\omega)$ allows the strength and the variation to be modulated separately, giving more flexibility to the user. As style functions $s(\omega)$ and $s^*(\omega)$ are defined in logarithmic coordinates, the relation between new and old MHSs can be written as the following exponential form:

$$\frac{h^*(\omega)}{h(\omega)} = e^{(s^*(\omega) - s(\omega))} = e^{\alpha(\bar{s}(\omega) - s(\omega))}. \tag{14}$$

Combining Eqn. (10), (12) and (14) yields

$$u_j^*(\tau) = u_j(\tau)\, e^{\alpha(\bar{s}(\omega_j(\tau)) - s(\omega_j(\tau)))}. \tag{15}$$

This is the final modulation equation applied to each IMF, and summing together the results as Eqn. (11) completes the modulation process on the flow field. Note that the modulation is not applied to the total velocity, and instead it is applied to each IMF (as well as the residual).

4.3 Direct IMF Modulation

The EMD method sequentially extracts IMFs from higher-frequency range to lower-frequency range. Although every IMF has a varying instantaneous frequency distribution, the central frequency bands of different IMFs are roughly detached with relatively small overlaps. This observation inspires us to modulate the flow field by simply scaling each IMF (and the residual) as a whole, i.e.

$$u^*(\tau) = \sum\nolimits_{j=1}^{J+1} \beta_j u_j(\tau) \tag{16}$$

where $\beta_j > 0$ are constant weight factors. To boost fine-scale features and suppress large-scale motions, one can simply choose larger β values for IMFs with smaller indices and smaller β values for IMFs with larger indices; and vice versa. As IMFs have band overlaps in their frequency distributions, this simple modulation method is not accurate, but for some simple style modulations, this approach is faster than the accurate amplitude modulation (15) since it does not require the Hilbert transform.

4.4 Consistency Guiding

Decomposition of the velocity field can bring other benefits. It is observed in our experiments that, the IMFs with lower frequencies contribute mainly to the global motion of the fluid flow while the higher-frequency IMFs contribute mainly to fine-scale details, an intuitive phenomenon that can be expected from the properties of IMFs. Thus, we propose a simple method to solve the consistency guiding problem, which concerns retaining in a high-resolution simulation the global appearance of the flow field simulated on a low-resolution grid.

It is assumed that the simulation on the low-resolution grid is completed and the fluid velocity data after each add-force step have been saved. While simulating on the high-resolution grid, both the current velocity field obtained on the high-resolution grid and the up-sampled low-resolution data are decomposed into the same number of IMFs and a residual, denoted respectively with $u_j^H(\tau)$ and $u_j^L(\tau)$. Then, to obtain the guided result $u^*(\tau)$, one just need to mix the IMFs such that high-resolution results have higher weights for high-frequency IMFs while low-resolution results have higher weights for low-frequency IMFs and the residual. That is

$$u^*(\tau) = \sum\nolimits_{j=1}^{J+1} \left(\gamma_j u_j^L(\tau) + (1 - \gamma_j) u_j^H(\tau)\right) \tag{17}$$

where the mixing weights $\gamma_j \in (0, 1)$ can be adjusted to give different balance between the high-resolution and low-resolution results.

Based on velocity decomposition, this simple approach avoids tedious adjustment of artificial forces and supports direct manipulation of the intrinsic multiscale structure of a flow field.

4.5 Algorithm Framework

In graphics applications, fluids are often assumed as inviscid and incompressible. Thus, the general Navier-Stokes equations become incompressible Euler equations, which are often solved via the following three steps:

$$\frac{Dq}{Dt} = 0 \tag{18}$$

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{g} \tag{19}$$

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{1}{\rho}\nabla p = 0 \quad s.t. \quad \nabla \cdot \mathbf{u} = 0 \tag{20}$$

where q denotes the physical value to be advected, t time, $\mathbf{u}$ velocity field, $\mathbf{g}$ body force, ρ density of fluid and p pressure. Eqns. (18–20) are called the advection, add-force, and projection steps.

The flow field modulation described in §4.1-§4.4 involves a decomposition of the fluid velocity field, an analyzing and modulation process on the decomposed velocity components, and a summing step that combines the modulated components to form the destination velocity field. These operations can be applied at any time

during the simulation, and we put them just before the projection step to form a simple algorithm framework:

1) Advection, following Eqn. (18).
2) Add force, following Eqn. (19).
3) Modulation, applied separately to individual IMFs, §4.2–§4.4.
4) Projection, following Eqn. (20).

After the add-force step in standard simulations, the velocity field is decomposed using the EMD algorithm described in §3.2, then individual IMFs and the residual can be modulated separately. The combined velocity field is passed to the projection step, which retains the divergence-free property after modulation.

4.6 Implementation Details

Our modulation algorithm is integrated into a standard staggered grid-based fluid solver, and the simulation process is only changed by adding a modulation step before the projection step. We here focus on the implementation of the modulation step itself.

The complete modulation process is as follows. First, each velocity component of the flow field is separately mapped into a 1D array, for which three 3D-to-1D mappings (one for each scalar component) are pre-computed following the technique in [37]. Then, the EMD is applied independently to each 1D array as described in §3.2 and §3.4 resulting in J IMFs and a residual, i.e. $u_j(\tau)$ in Eqns. (9) and (15), after which the 1D version of Hilbert transform (2) is applied on each IMF as well as the residual. Next, following Eqns. (2-8), the MHS of the original field is calculated, and $J+1$ arrays are created to store the instantaneous frequency values at each data point. The style function $s(\omega)$ is obtained from the MHS following the steps in §4.1, which will be discussed later in more detail. Finally, after applying Eqn. (15) to each IMF component (including the residual) $u_j(\tau)$ and summing together the results $u_j^*(\tau)$ as Eqn. (11), the modulated total velocity $u^*(\tau)$ is mapped to the 3D space. For the modulation methods described in §4.3 and §4.4, it is sufficient to directly manipulate each IMF component $u_j(\tau)$, with no need of the Hilbert transform or the instantaneous frequency calculation.

In the EMD algorithm the stopping criterion (step 2(f)) has much influence over both the decomposition result and the computing time. We use a stopping criterion similar to [43]. In the EMD sifting procedure, the envelope mean is defined as $E_{mean} = (E_{min} + E_{max})/2$, and the envelope amplitude is defined as $E_{amp} = (E_{max} - E_{min})/2$. The sifting process is terminated if one of the following two conditions is satisfied:

1) At each data point i,
$$E_{mean}(i) < \epsilon_1 E_{amp}(i). \tag{21}$$
2) Mean of the boolean array $\frac{E_{mean}(i)}{E_{amp}(i)} > \epsilon_2$ is smaller than a tolerance value δ, i.e.
$$\sum_{i=1}^{N} \left(\frac{E_{mean}(i)}{E_{amp}(i)} > \epsilon_2 \right) < N\delta. \tag{22}$$

Here ϵ_1, ϵ_2 and δ are constants, and N is the size of the data set. We set $\epsilon_1 = 0.5$, $\epsilon_2 = 0.05$ and $\delta = 0.35$ in our implementation. We also set an upper bound for the total number of inner iterations in the EMD algorithm. As the inner iteration usually converges to a good approximation between 10 to 30 iterations, the upper bound is set as 40.

The computation steps of the style function $s(\omega)$ are explained in §4.1, and $s(\omega)$ values are stored in a float array of size 400, in ascending order of the spatial frequency ω. To reduce sampling noise introduced in the Hilbert transform step, for each MHS data set, a median filtering is applied to reduce its fluctuation. The filter window at the low-frequency range is 3 and it increases linearly to 10 at the high-frequency range. In the linear fitting, the standard least-squares linear regression usually produces a large deviation at the low-frequency band due to the much higher point density in the high-frequency band. To avoid this undesirable bias, we force the fitting result to go through the point corresponding to the lowest frequency. The guiding style function $\overline{s}(\omega)$ in Eqn. (13) is similarly stored in a float array of size 400.

In the velocity decomposition, the initial values of those positions that are occupied by a solid obstacle are set as the obstacle's velocity. Same as the standard simulation, we leave the no-stick boundary condition to be enforced in the projection step, which for grid-based solvers provides a better approximation of the rapid local velocity change [44].

4.7 Modulation Effects on the Flow Field

As the outcome of some sophisticated derivations in §3 and §4, the modulating operation in Eqn. (15) is rather simple to implement and the frequency-based modulation relates directly to the flow's visual effects. Eqn. (15) acts as an equalizer on the velocity field, where the scaling factor at a certain frequency is determined by the difference between the guiding style and the current style. Thus, an equalizer-like interface can be provided to end users, and the flow field modulation can be carried out similarly to audio equalization. In our practice, this is done by turning up/down the corresponding entries in the $\overline{s}(\omega)$ array (of size 400), which effectively changes the guiding style. Many desirable visual effects can be directly achieved, and we list below some typical effects that will be demonstrated in §5.

Energizing Flow: Boosting low frequencies, i.e. boosting fore entries of the $\overline{s}(\omega)$ array will make the intrinsic main flow more energetic and violent.

Stabilizing Flow: Cutting fore entries of the $\overline{s}(\omega)$ array will suppress eddy generation, leaving a simple and steady main flow.

Adding Patterns: Boosting middle frequencies, i.e. boosting median entries of the $\overline{s}(\omega)$ array can add wrinkling features to the original smooth field.

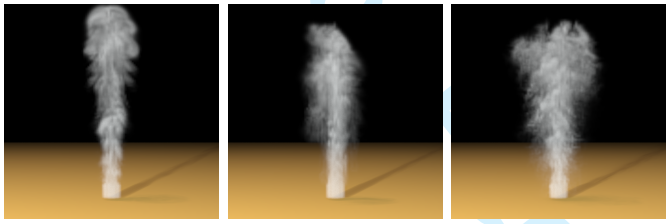
Detail Enhancing: Boosting high frequencies, i.e. boosting rear entries of the $\overline{s}(\omega)$ array will bring chaotic

small turbulence to the flow and add to the splitting trend of the flow front.

Other parameters such as α in Eqn. (15) and β in Eqn. (16) are global constants. They offer control on how fast the modulation takes effect. A smaller α will directly reduce the scaling factor for all frequencies, so that the modulation effect gradually becomes visible in longer time. To the opposite, larger α will bring strong modulations, in which case the effect will become visible within several simulation steps.

5 Results and Discussion

5.1 Results



(a) (b) (c)

Fig. 3. Example 2, single plume case, style modulation. (a) The original; (b) low-frequency suppression with $\alpha = 0.5$, the eddy-like global structures are largely removed resulting in a straight skeleton of plume; (c) high-frequency boost with $\alpha = 0.4$, splitting trend and chaotic fine features are enhanced while retaining the global motion.

(a) (b) (c)

Fig. 5. Example 4, cylinder obstacle case, style modulation. (a) The original; (b) mid-frequency suppression with $\alpha = 0.5$; (c) mid-frequency boost with $\alpha = 0.1$. Mid-frequency modulation affects to some extent both global flow motions and fine-scale details.

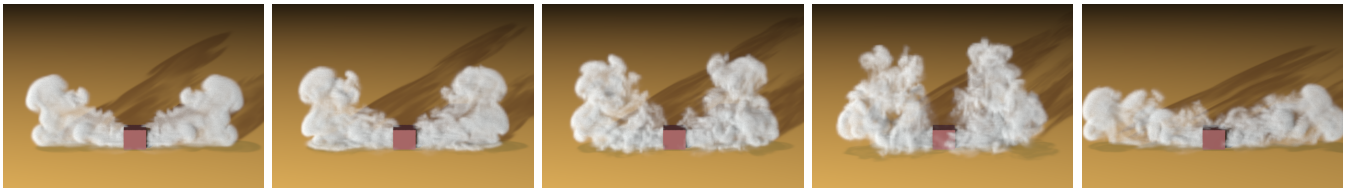
Style Modulation: Examples 1–4 are designed to examine the performance of style modulation described in §4.2. The flow field is modulated following Eqn. (15). Example 1 (Fig. 1) demonstrates a complicated flow field in a mountainous environment. Fig. 1(a) shows the original simulation without modulation. Fig. 1(b) is the result of low-frequency boost by setting the strength factor $\alpha = 0.1$, the first 200 entries of the guiding style array $\overline{s}(\omega)$ as 0.5 and the rest entries as 0. The low-frequency motions are the main energy carrier, and Fig. 1(b) looks much more energetic, showing significant global transformations, but the intrinsic shape of the flow field is largely retained. Fig. 1(c) is the result of mid-frequency boost by setting $\alpha = 0.5$, the middle 200 entries in the $\overline{s}(\omega)$ array as 0.5 and the rest entries as 0. Comparing with Fig. 1(a), Fig. 1(c) has a wrinkled appearance, and as the mid-frequency motions do not carry as much energy as the low-frequency motion, the flow field stays even closer to the original shape comparing with Fig. 1(b).

Example 2 (Fig. 3) is a plume of smoke rising from the ground due to buoyancy. Fig. 3(a) shows the result obtained from the standard simulation without modulation, which contains clear eddy-like global structures as well as some irregular fine-scale features. Fig. 3(b) shows a low-frequency suppression result, obtained by setting the strength factor $\alpha = 0.5$, the first 200 entries of the guiding style array $\overline{s}(\omega)$ as -0.5 and the rest entries as 0. It can be seen that the eddy-like global structures are largely removed such that the skeleton of the plume becomes a simple straight column, and as a result the irregular fine-scale features become relatively more visible. Fig. 3(c) shows the effect of high-frequency boost, obtained by setting $\alpha = 0.4$, the last 200 entries in the $\overline{s}(\omega)$ array as 0.5 and the rest as 0. It can be seen that the splitting trend and chaotic whirls are significantly enhanced while retaining the global motion.

Example 3 (Fig. 4) mimics gas explosion. Fig. 4(a) is the original result without modulation. Fig. 4(b–d) are low-frequency boost results obtained by setting the strength factor α as 0.1, 0.2 and 0.3 respectively, setting the first 200 entries in the $\overline{s}(\omega)$ array as 0.5 and setting the rest entries as 0. In Figs. 4(b–d) the modulation is only applied to velocity components for vertical and depth directions, and therefore the global flow motion along the horizontal direction is not affected. As a result of low-frequency boost, the flow field becomes more vibrant but does not look more noisy. Fig. 4(e) shows a low-frequency boost modulation applied to all velocity components with $\alpha = 0.1$.

Example 4 (Fig. 5) tests mid-frequency modulation with two obstacles inside the fluid domain. Fig. 5(a) is the original result without modulation. Fig. 5(b) is the mid-frequency suppression result with $\alpha = 0.5$, the middle 200 entries in the $\overline{s}(\omega)$ array set as -0.5 and the rest entries set as 0. Fig. 5(c) is the mid-frequency boost result with $\alpha = 0.1$, the middle 200 entries in the $\overline{s}(\omega)$ array set as 0.5 and the rest entries set as 0. Comparing Figs. 5(a–c), it can be seen that the mid-frequency modulation affects to some extent both global flow motions and fine-scale details.

Direct IMF Modulation: Example 5 (Fig. 6) tests the direct IMF modulation described in §4.3. For comparison purposes, we repeat the simulation of example 2. In order to better demonstrate the performance of direct IMF modulation, we choose a further developed plume from a later frame comparing with the snapshot in Fig. 3(a). The velocity field is decomposed into 8 IMFs plus the residual, and following Eqn. (16), the flow field is modulated by directly specifying the weight factors



(a) (b) (c) (d) (e)

Fig. 4. Example 3, box explosion case, style modulation. (a) The original; (b–d) modulation with low-frequency motion boosted respectively by $\alpha = 0.1, 0.2, 0.3$ for vertical and depth directions; (e) modulation with low-frequency motion boosted by $\alpha = 0.1$ for all directions. Controlled by the strength factor α, the global motions are boosted to different extents in (b–e).

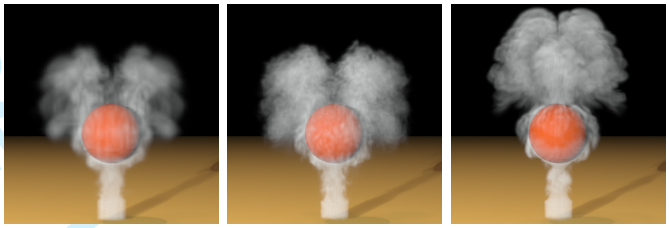
Experiment	Method	Grid Resolution	IMF Number	Simulation (s/frame)	Modulation (s/frame)
example 1	style modulation	$128 \times 64 \times 128$	$8+1$	5.28	5.20
example 2	style modulation	$64 \times 128 \times 64$	$8+1$	2.43	2.62
example 3	style modulation	$128 \times 128 \times 128$	$8+1$	10.58	(b–d) 7.20, (e) 10.79
example 4	style modulation	$64 \times 64 \times 64$	$8+1$	1.16	1.09
example 5	direct IMF modulation	$64 \times 128 \times 64$	$8+1$	2.43	1.68
example 6	consistency guiding	$128 \times 256 \times 128$	$8+1$	19.56	13.25×2

TABLE 2
Simulation parameters and performance data.

(a) (b) (c)

Fig. 6. Example 5, direct IMF modulation. (a) The original; (b) high-frequency suppression with $\beta_j = 0.9 \quad (j = 1, \cdots, 4)$ and $\beta_j = 0 \quad (j = 5, \cdots, 9)$; (c) high-frequency boost with $\beta_j = 1.005 \quad (j = 1, \cdots, 4)$ and $\beta_j = 0 \quad (j = 5, \cdots, 9)$. The direct IMF modulation is fast and can achieve some simple effects, but comparing with the full style modulation, it does not provide the same accuracy or freedom.

β_j. Fig. 6(a) is the original result without modulation. Fig. 6(b) shows a high-frequency suppression result, obtained by setting $\beta_j = 0.9 \quad (j = 1, \cdots, 4)$ and $\beta_j = 1 \quad (j = 5, \cdots, 9)$. It can be seen that the modulation suppresses not only the fine-scale features but also the eddy-like global motions, making the bottom of the plume become a straight column. This is because each IMF has a varying frequency distribution, and therefore modulating any IMF will affect the flow features in a frequency range rather than at a single frequency. Fig. 6(c) shows a high-frequency boost result, obtained by setting $\beta_j = 1.005 \quad (j = 1, \cdots, 4)$ and $\beta_j = 1 \quad (j = 5, \cdots, 9)$. Again, as each IMF corresponds to a frequency band, the boost modulation enhances both chaotic fine features and regular global motions. The direct IMF modulation is fast and can achieve some simple effects, but comparing with the full style modulation, it does not provide the same accuracy or freedom.



(a) (b) (c)

Fig. 7. Example 6, consistency guiding. (a) The low-resolution result; (b) the guiding result with $\gamma_j = 0 \quad (j = 1, \cdots, 8)$ and $\gamma_9 = 1$; (c) the high-resolution result. The example demonstrates the decomposition is able to successfully remove and replace the large scales of a flow.

Consistency Guiding: Example 6 (Fig. 7) demonstrates the effect of consistency guiding described in §4.4. Fig. 7(a) is the simulation result obtained on a $64 \times 128 \times 64$ grid, and Fig. 7(c) is the result obtained on a $128 \times 256 \times 128$ grid. Fig. 7(b) is the guided simulation using Eqn. (17). The velocity field is decomposed into 8 IMFs plus a residual, and the guiding weights γ_j are simply set as $\gamma_j = 0 \quad (j = 1, \cdots, 8)$ and $\gamma_9 = 1$. It is demonstrated that the decomposition is able to successfully remove and replace the large scales of a flow. It is seen that the guided result retains the same global motion as obtained in the low-resolution simulation while providing more fine-scale details. This simple and intuitive consistency guiding approach operates directly on the decomposed velocity field, and does not require solving extra equations or introducing artificial forces.

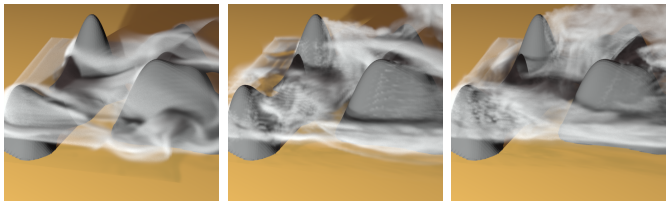
We parallelized our algorithm and implemented and tested on a quad-core computer. The simulation parameters and performance data for all examples are listed in

Table 2. The velocity fields are decomposed into 8 IMFs plus the residual in these examples. The style modulation in examples 1–4 costs roughly the same amount of CPU time as the simulation step. The direct IMF modulation in example 5 costs relatively less, because the computation of Hilbert transform is not required. In example 6, the EMD operation is performed twice for consistency guiding: one performed on the high-resolution result, and the other on the low-resolution result, which can be computed offline. The computational cost of our modulation approach is similar to the standard simulation, and it increases proportionally with the total grid number.

5.2 Comparison with Other Decomposition Methods

(a) (b) (c)

Fig. 8. Fourier modulation. (a) The original; (b) low-frequency components scaled by 0.95, the total flow field is reduced with expansion towards the surrounding; (c) high-frequency components scaled by 1.05, noisy fine-scale details are added causing the total flow field uniformly expanding to all directions.



(a) (b) (c)

Fig. 9. Wavelet modulation. (a) The last low-frequency component is scaled by 1.1; (b) the last two low-frequency components are scaled by 1.1; and (c) the last three low-frequency components are scaled by 1.05. Some macro-scale structures are smoothed out in (a), while significant amount of unnatural motions are introduced in (b) and (c).

Traditional decomposition methods are also investigated in our study. We found that they do not produce satisfactory results for flow field modulation. Here we discuss their difference and limitation in analyzing flow fields. We selected the two most widely used decomposition methods for fluid simulation, namely Fourier- and wavelet-based methods, in this comparison.

A Fourier-based modulation approach can be developed by using Fourier transform to extract frequency information from the velocity field and applying the modulation to the Fourier components in a similar way as §4.3. We tested a number of parameter configurations on different scenes, varying the number of Fourier components and the weight factors. A typical example is shown in Fig. (8), where the same plume as in Fig. (3) is modulated. Fig. (8a) demonstrates low-frequency suppression by scaling the low-frequency components with a factor of 0.95, and Fig. (8b) is the high-frequency boost result obtained by scaling the high-frequency components with a factor of 1.05. The Fourier modulation failed in the low-frequency suppression case: instead of suppressing only the large-scale fluid motions, the total flow field is reduced with the plume expanding towards the surrounding. For the high-frequency boost case, some fine-scale details are added to the plume, but they cause the total flow field uniformly expanding to all directions. As Fourier decomposition relies on pre-defined base functions with constant frequencies and global support over the whole domain, it has fundamental difficulty in capturing the non-stationary multi-scale local character of the flow field, and hence cannot correctly guide the modulation of flow fields.

A wavelet-based modulation approach can be similarly developed by replacing the SHHT with the wavelet transform. We follows the wavelet implementation in [14]. Although the wavelet approach is found in many test cases to perform better than the Fourier approach, it is still not suitable for general flow field modulation. A group of test examples are shown in Fig. (9), where the same fluid scene as in Fig. (1) is adopted. Fig. (9a) is obtained by scaling the lowest frequency component with a factor of 1.1; Fig. (9b) is obtained by scaling the last two low-frequency components with a factor of 1.1; and Fig. (9c) is obtained by scaling the last three low-frequency components with a factor of 1.05. None of these trials succeeded in enhancing the macro-scale fluid motion. Comparing to the original flow shown in Fig. (1a), some macro-scale structures are smoothed out in Fig. (9a) at the center of the mountain scene. However, what we are doing here is to boost the features. This failure is not due to the strength of boost, and it shows wavelets unpredictable effects in modulating low-frequency motions. Processing more low-frequency components does not help, and significant amount of unnatural motions are introduced into Figs. (9b-c) even causing the flow field to diverge. In these examples, the failure of the wavelet modulation is due to the misrecognition of the low-frequency motion of the flow field. The wavelet base functions feature local support sets, and comparing to the Fourier basis they have a better potential in coping with the non-stationarity of turbulent flow. However, the wavelet transform still relies on pre-defined base functions with a fixed set of spatial frequencies. Hence, it cannot accurately represent the flow field's "instantaneous frequencies and amplitudes" which vary continuously through the fluid domain. The misrepresentation is particularly worse for the low-frequency range, where the wavelet frequencies

are "sparse".

In contrast to traditional data analysis tools, the SHHT is self-adaptive to the data set, and can well define instantaneous frequencies and amplitudes at any value for the whole frequency range. It is this unique property that allows the SHHT to correctly extract the intrinsic multi-scale features from the nonlinear and non-stationary flow field and to be a general modulation method for various flow types. For example, without damaging the flow's global shape, the SHHT modulation can enhance separately intrinsic flow features at different length scales (see e.g. Fig. (1)); besides boosting effects, the SHHT modulation also allows suppression of multi-scale flow features producing naturally stabilized results (see e.g. Figs. (3) and (5)), which cannot be achieved by turning up buoyancy or adding artificial forces.

6 Conclusion

We have introduced a novel method that uses Hilbert-Huang transform to decompose flow fields from 3D fluid simulation into a set of frequency components, which reveal the intrinsic multiscale structure and the associated frequency information of the flow fields. Based on the proposed spatial Hilbert-Huang transform, we define a quantitative measure of flow style, describing intrinsic flow features at different length scales as a function of spatial frequencies of the flow field. The flow style from a fluid simulation can be modulated by directly modifying velocity fields at specific spatial frequencies. Several modulation methods are proposed, and they allow the multiscale flow features to be boosted and suppressed independently. This modulation method is very easy to implement, and can be readily integrated into standard fluid solver within the existing algorithm framework.

There are some limitations that deserve discussion. The 3D-to-1D mapping technique affects the accuracy of Hilbert transform by sampling the total derivatives with directional derivatives, and theoretically it shares common drawbacks of any sampling method in the sense of information loss. Calculating derivatives along the space-filling curve can cause false estimation at some positions on the grid edge, where the space-filling curve travels out of the grid for some while and comes back to another nearby but not bordering position. Thus when processing a flow with violent motions near the grid edge, instantaneous frequency calculation can be unreliable at these positions. To overcome this problem, one can extrapolate the velocity values or enlarge the grid coverage, involving extra computational cost. Also, for a neighborhood where a steady flow undergos a bare low-frequency motion, sampling the total derivative semi-randomly along the space-filling curve can potentially introduce artificial medium-to-high frequency components. When these frequency components are boosted, fine-scale structures may be generated earlier than their physical evolvement. Hence, extra caution must be taken when boosting a plain flow field whose large-scale structures have not yet broken to motivate small-scale features.

As the modulation operates on the intrinsic flow modes and intrinsic frequencies that are captured by the current simulation grid, it cannot add fine-scale features beyond the grid resolution. It is noted that practical flows can have time-dependent features. However, although the guiding style functions are not required to remain the same during simulation, time variance of the style curves is not taken into account yet. These two important aspects will be investigated in our future work.

References

[1] J. Stam, "Stable fluids," in *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, ser. SIGGRAPH '99. New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 1999, pp. 121–128. [Online]. Available: http://dx.doi.org/10.1145/311535.311548

[2] R. Fedkiw, J. Stam, and H. W. Jensen, "Visual simulation of smoke," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, ser. SIGGRAPH '01. New York, NY, USA: ACM, 2001, pp. 15–22. [Online]. Available: http://doi.acm.org/10.1145/383259.383260

[3] A. Selle, N. Rasmussen, and R. Fedkiw, "A vortex particle method for smoke, water and explosions," *ACM Trans. Graph.*, vol. 24, pp. 910–914, July 2005. [Online]. Available: http://doi.acm.org/10.1145/1073204.1073282

[4] Y. Zhu and R. Bridson, "Animating sand as a fluid," *ACM Trans. Graph.*, vol. 24, pp. 965–972, July 2005. [Online]. Available: http://doi.acm.org/10.1145/1073204.1073298

[5] T. F. Dupont and Y. Liu, "Back and forth error compensation and correction methods for removing errors induced by uneven gradients of the level set function," *Journal of Computational Physics*, vol. 190, pp. 311–324, 2003.

[6] J. Molemaker, J. M. Cohen, S. Patel, and J. Noh, "Low viscosity flow simulations for animation," in *Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, ser. SCA '08. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2008, pp. 9–18. [Online]. Available: http://portal.acm.org/citation.cfm?id=1632592.1632595

[7] A. Selle, R. Fedkiw, B. Kim, Y. Liu, and J. Rossignac, "An unconditionally stable maccormack method," *J. Sci. Comput.*, vol. 35, no. 2-3, pp. 350–371, 2008.

[8] B. E. Feldman, J. F. O'Brien, and O. Arikan, "Animating suspended particle explosions," *ACM Trans. Graph.*, vol. 22, pp. 708–715, July 2003. [Online]. Available: http://doi.acm.org/10.1145/882262.882336

[9] F. Losasso, F. Gibou, and R. Fedkiw, "Simulating water and smoke with an octree data structure," *ACM Trans. Graph.*, vol. 23, pp. 457–462, August 2004. [Online]. Available: http://doi.acm.org/10.1145/1015706.1015745

[10] J. Stam and F. Eugene, "Turbulent wind fields for gaseous phenomena," in *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, ser. SIGGRAPH '93. New York, NY, USA: ACM, 1993, pp. 369–376. [Online]. Available: http://doi.acm.org/10.1145/166117.166163

[11] A. Lamorlette and N. Foster, "Structural modeling of flames for a production environment," *ACM Trans. Graph.*, vol. 21, pp. 729–735, July 2002. [Online]. Available: http://doi.acm.org/10.1145/566654.566644

[12] N. Rasmussen, D. Q. Nguyen, W. Geiger, and R. Fedkiw, "Smoke simulation for large scale phenomena," *ACM Trans. Graph.*, vol. 22, pp. 703–707, July 2003. [Online]. Available: http://doi.acm.org/10.1145/882262.882335

[13] R. Bridson, J. Houriham, and M. Nordenstam, "Curl-noise for procedural fluid flow," *ACM Trans. Graph.*, vol. 26, July 2007. [Online]. Available: http://doi.acm.org/10.1145/1276377.1276435

[14] T. Kim, N. Thürey, D. James, and M. Gross, "Wavelet turbulence for fluid simulation," *ACM Trans. Graph.*, vol. 27, pp. 50:1–50:6, August 2008. [Online]. Available: http://doi.acm.org/10.1145/1360612.1360649

[15] R. Narain, J. Sewall, M. Carlson, and M. C. Lin, "Fast animation of turbulence using energy transport and procedural synthesis," *ACM Trans. Graph.*, vol. 27, pp. 166:1–166:8, December 2008. [Online]. Available: http://doi.acm.org/10.1145/1409060.1409119

[16] H. Schechter and R. Bridson, "Evolving sub-grid turbulence for smoke animation," in *Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, ser. SCA '08. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2008, pp. 1–7. [Online]. Available: http://portal.acm.org/citation.cfm?id=1632592.1632594

[17] T. Pfaff, N. Thuerey, A. Selle, and M. Gross, "Synthetic turbulence using artificial boundary layers," *ACM Trans. Graph.*, vol. 28, pp. 121:1–121:10, December 2009. [Online]. Available: http://doi.acm.org/10.1145/1618452.1618467

[18] T. Pfaff, N. Thuerey, J. Cohen, S. Tariq, and M. Gross, "Scalable fluid simulation using anisotropic turbulence particles," *ACM Trans. Graph.*, vol. 29, pp. 174:1–174:8, December 2010. [Online]. Available: http://doi.acm.org/10.1145/1882261.1866196

[19] N. Foster and D. Metaxas, "Controlling fluid animation," in *Computer Graphics International, 1997. Proceedings*, jun 1997, pp. 178 –188.

[20] A. Treuille, A. McNamara, Z. Popović, and J. Stam, "Keyframe control of smoke simulations," in *ACM SIGGRAPH 2003 Papers*, ser. SIGGRAPH '03. New York, NY, USA: ACM, 2003, pp. 716–723. [Online]. Available: http://doi.acm.org/10.1145/1201775.882337

[21] A. McNamara, A. Treuille, Z. Popović, and J. Stam, "Fluid control using the adjoint method," *ACM Trans. Graph.*, vol. 23, pp. 449–456, August 2004. [Online]. Available: http://doi.acm.org/10.1145/1015706.1015744

[22] J.-M. Hong and C.-H. Kim, "Controlling fluid animation with geometric potential," *Comput. Animat. Virtual Worlds*, vol. 15, pp. 147–157, July 2004. [Online]. Available: http://dx.doi.org/10.1002/cav.v15:3/4

[23] R. Fattal and D. Lischinski, "Target-driven smoke animation," *ACM Trans. Graph.*, vol. 23, pp. 441–448, August 2004. [Online]. Available: http://doi.acm.org/10.1145/1015706.1015743

[24] L. Shi and Y. Yu, "Controllable smoke animation with guiding objects," *ACM Trans. Graph.*, vol. 24, pp. 140–164, January 2005. [Online]. Available: http://doi.acm.org/10.1145/1037957.1037965

[25] ——, "Taming liquids for rapidly changing targets," in *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, ser. SCA '05. New York, NY, USA: ACM, 2005, pp. 229–236. [Online]. Available: http://doi.acm.org/10.1145/1073368.1073401

[26] N. Rasmussen, D. Enright, D. Nguyen, S. Marino, N. Sumner, W. Geiger, S. Hoon, and R. Fedkiw, "Directable photorealistic liquids," in *Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation*, ser. SCA '04. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2004, pp. 193–202. [Online]. Available: http://dx.doi.org/10.1145/1028523.1028549

[27] N. Thürey, R. Keiser, M. Pauly, and U. Rüde, "Detail-preserving fluid control," *Graph. Models*, vol. 71, pp. 221–228, November 2009. [Online]. Available: http://dl.acm.org/citation.cfm?id=1651926.1652003

[28] V. Mihalef, D. Metaxas, and M. Sussman, "Animation and control of breaking waves," in *Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation*, ser. SCA '04. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2004, pp. 315–324. [Online]. Available: http://dx.doi.org/10.1145/1028523.1028565

[29] M. B. Nielsen, B. B. Christensen, N. B. Zafar, D. Roble, and K. Museth, "Guiding of smoke animations through variational coupling of simulations at different resolutions," in *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, ser. SCA '09. New York, NY, USA: ACM, 2009, pp. 217–226. [Online]. Available: http://doi.acm.org/10.1145/1599470.1599499

[30] M. B. Nielsen and R. Bridson, "Guide shapes for high resolution naturalistic liquid simulation," *ACM Trans. Graph.*, vol. 30, pp. 83:1–83:8, July 2011. [Online]. Available: http://doi.acm.org/10.1145/1964921.1964978

[31] Z. Yuan, F. Chen, and Y. Zhao, "Pattern-guided smoke animation with lagrangian coherent structure," in *Proceedings of the 2011 SIGGRAPH Asia Conference*, ser. SA '11. New York, NY, USA: ACM, 2011, pp. 136:1–136:8. [Online]. Available: http://doi.acm.org/10.1145/2024156.2024170

[32] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N. C. Yen, C. C. Tung, and H. H. Liu, "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis," *Proc. R. Soc. London Ser. A-Math. Phys. Eng. Sci.*, vol. 454, no. 1971, pp. 903–995, Mar. 1998. [Online]. Available: http://dx.doi.org/10.1098/rspa.1998.0193

[33] J. Nunes, "Image analysis by bidimensional empirical mode decomposition," *Image and Vision Computing*, vol. 21, no. 12, pp. 1019–1026, 2003. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0262885603000945

[34] 1. Deléchelle, J.-C. Nunes, and J. Lemoine, "Empirical mode decomposition synthesis of fractional processes in 1d- and 2d-space," *Image Vision Comput.*, vol. 23, pp. 799–806, September 2005. [Online]. Available: http://dx.doi.org/10.1016/j.imavis.2005.05.012

[35] Y. Zhang, Z. Sun, and W. Li, "Texture synthesis based on direction empirical mode decomposition." *Computers and Graphics*, pp. 175–186, 2008.

[36] K. Subr, C. Soler, and F. Durand, "Edge-preserving multiscale image decomposition based on local extrema," *ACM Trans. Graph.*, vol. 28, pp. 147:1–147:9, December 2009. [Online]. Available: http://doi.acm.org/10.1145/1661412.1618493

[37] Y. Gao, C.-F. Li, B. Ren, and S.-M. Hu, "View-dependent multiscale fluid simulation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 99, no. PrePrints, 2012.

[38] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, *Spectral Methods in Fluid Dynamics*. Springer-Verlag, November 1987.

[39] P. Sagaut, *Large Eddy Simulation for Incompressible Flows: An Introduction*. Springer-Verlag, November 2010.

[40] G. E. Karniadakis and S. J. Sherwin, *Spectral/hp Element Methods for Computational Fluid Dynamics*. Oxford University Press, August 2005.

[41] J. Stam, "A simple fluid solver based on the fft," *J. Graph. Tools*, vol. 6, no. 2, pp. 43–52, Sep. 2002. [Online]. Available: http://dx.doi.org/10.1080/10867651.2001.10487540

[42] F. W. King, *Hilbert Transforms*. Cambridge University Press, April 2009.

[43] G. Rilling, P. Flandrin, and P. Gonçalvès, "On empirical mode decomposition and its algorithms," in *Proceedings of the 6th IEEE/EURASIP Workshop on Nonlinear Signal and Image Processing (NSIP '03), Grado, Italy*, 2003.

[44] R. Bridson, *Fluid Simulation for Computer Graphics*. A K Peters/CRC Press, Sep. 2008.