

# Seeing and Hearing the Eigenvectors of a Fluid

Aaron Demby-Jones, JoAnn Kuchera-Morin and Theodore Kim

Media Arts and Technology Program

University of California, Santa Barbara

aaron.demby.jones@mat.ucsbg.edu, jkm@create.ucsbg.edu, kim@mat.ucsbg.edu

## Abstract

The intricate shapes and sounds that arise from the eigenvectors of vibrating Chladni plates are a well-known phenomenon. We explore a similar phenomenon by computing and visualizing the patterns formed by the empirical eigenvectors of a computational fluid dynamics simulation. The eigenvectors form complex, turbulent, spatial patterns. The eigenvalues form a spectrum that does not possess a unique sonic mapping, so we construct a sonification that allows us to listen to the trajectory of the fluid.

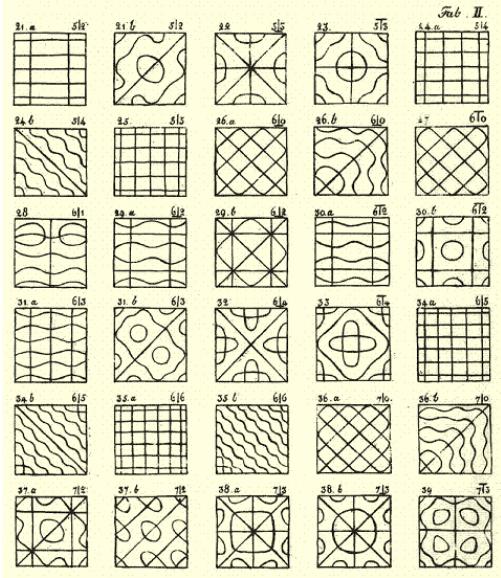
## Introduction

Chladni plates beautifully reveal the vibration modes, also known as the eigenvectors, of a rigid plate (Fig. 1a). These eigenvectors can be computed analytically, as they correspond to very small deformations (i.e. vibrations) on a very simple domain, and can be written in terms of the Laplacian operator ( $\Delta$ ) over a square domain. These visual representations can be extended to more complex scenarios, where the eigenvectors and their corresponding eigenvalues are instead computed numerically. In this work, we compute the eigenvectors of a dynamic computational fluid dynamics simulation, which involves large deformations instead of small vibrations, and the Navier-Stokes equations in lieu of a Laplacian. We then visualize the turbulent motion of the eigenvectors in this spectral representation, and construct a sonification of the resulting frequencies to produce a mapping between fluid trajectories and sound. The results are organic visual forms with a unique visual character, and sounds with ♠ say something nice about the sounds ♠.

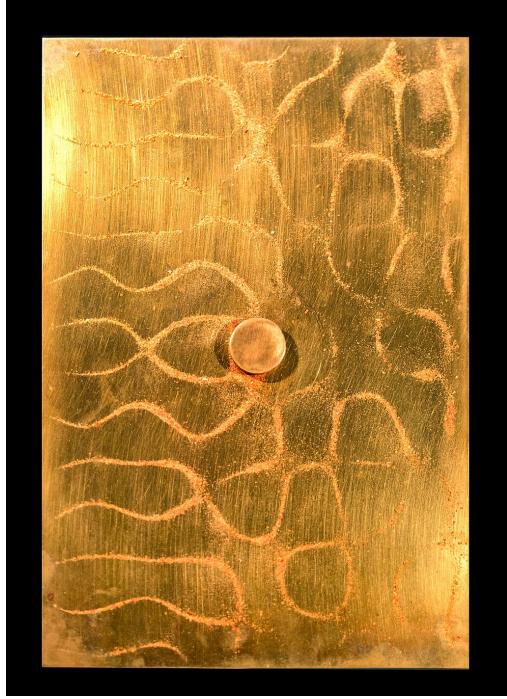
## Subspace fluids

(/PLACEHOLDER) While many subspace fluid simulation algorithms have been developed for computer graphics, they all use Galerkin projection. The basic strategy of this approach is to represent a large-dimensional velocity field in terms of a linear combination of a small number of carefully chosen basis vectors. To that end, a large-dimensional velocity field that contains  $3N$  degrees of freedom over  $N$  grid cells is projected into a smaller,  $r$ -dimensional velocity subspace  $S \simeq \mathbb{R}^r$  using a static linear basis. We denote this change-of-basis matrix as  $\mathbf{U} \in \mathbb{R}^{3N \times r}$ , where each column of  $\mathbf{U}$  is itself a velocity field. Specifically, we can project a vector  $\mathbf{u} \in \mathbb{R}^{3N}$  down to its reduced-order counterpart  $\mathbf{q} \in \mathbb{R}^r$  by computing the product  $\mathbf{U}^T \mathbf{u} = \mathbf{q}$ . In a well-designed subspace simulation, the velocity fields spanned by the columns of  $\mathbf{U}$ ; i.e., the subspace  $S$ , will be sufficient to capture the space of fields that interest the user. This projection often occurs at the beginning of each timestep, as a force vector  $\mathbf{f}$  that contains important information such as user inputs, buoyancy, and vorticity confinement forces must be projected into the subspace.

Once this projection has been performed, the equations of motion can be integrated over this reduced number of variables very quickly. This can be done using exponential integration, Stable Fluids-like semi-Lagrangian advection coupled with Chorin projection, variational integration, or even simple explicit Euler integration. Once time integration has been performed, the change-of-basis must be reversed so that the results can be displayed. This is known as the “velocity reconstruction” stage. The most direct method is a



(a) A table of hand-drawn Chladni figures.  
Source: Wikimedia Commons.



(b) The physical experiment with a rectangular plate. Source: Wikimedia Commons.

*full* reconstruction where the  $3N$ -dimensional velocity field is reconstructed directly, i.e.  $\mathbf{u} = \mathbf{U}\mathbf{q}$ ; and it is the approach we take in this paper. (/endPLACEHOLDER)

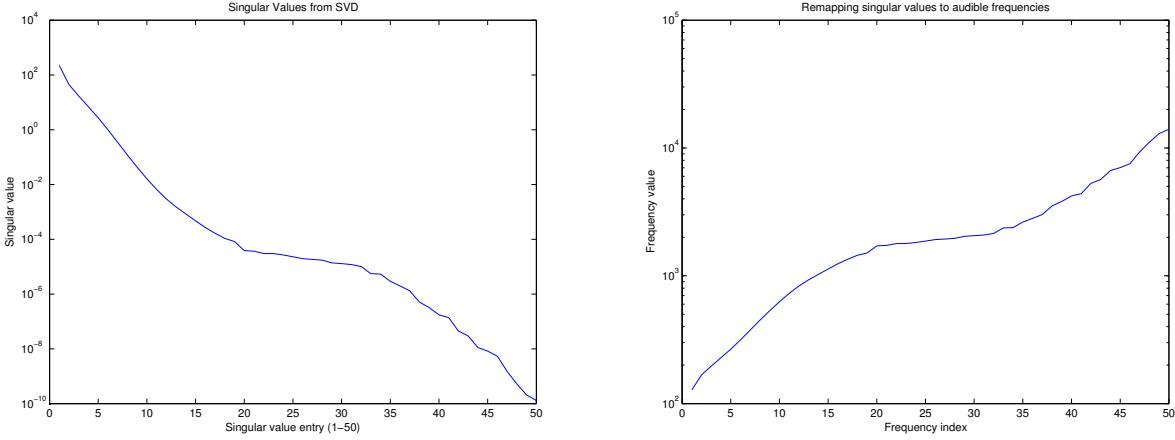
In previous work, the subspace  $S$  is discovered by first performing an appropriate training simulation at the full dimension and performing a singular value decomposition. Discarding the singular values less than a small threshold leaves us with  $r$  singular values, with  $r \ll 3N$ , and their corresponding  $r$  singular vectors, which are the basis vectors in the full space  $\mathbb{R}^{3N}$  that generate  $S$ . In an analogy to the Chladni plate vibrations, these singular vectors correspond to the eigenfunctions, and the singular values correspond to the frequencies.

For each time step of the subspace re-simulation, when we perform the reconstruction phase, obtaining a full-space velocity field  $\mathbf{v} \in \mathbb{R}^{3N}$  from a reduced-order subspace vector  $\mathbf{q} \in \mathbb{R}^r$ , we are simply taking a linear combination of the  $r$  singular vectors using the corresponding coefficients given by the entries of  $\mathbf{q}$ . Hence, the subspace vectors  $\mathbf{q}$  can be intuitively thought of as instructions for how to mix together the corresponding singular vectors with the appropriate weightings. Indeed, we can even identify a path  $\varphi: [0, 1] \rightarrow S$ , sampled at  $T$  timesteps  $\varphi_1, \dots, \varphi_T$ , as determining the corresponding  $T$  timesteps of the full-space re-simulation, after reconstruction.

## Sonification

We interpret the  $r$  singular values  $\sigma_1, \dots, \sigma_r$  from the singular-value decomposition as characteristic frequencies.

There are two immediate concerns about the raw data. Firstly, from Figure 2a, we see that the values range over approximately 12 orders of magnitude, which is approximately 40 octaves. However, the human audible dynamic range is generally considered to be from 20 Hz to 20000 Hz, which is only about 10 octaves. Thus, we will have to rescale the octaves into an appropriate range. Secondly, starting from the principal



(a) **Singular values:**  $r = 50$ . Note the logarithmic scale on the y-axis.  
(b) **Remapped frequencies:**  $f = 128 \text{ Hz}$ ,  $s = 6$ . Note again the logarithmic scale on the y-axis.

singular value, the singular values are decreasing, whereas an audio spectrum increases from its fundamental frequency. Hence, we will have to invert the singular values. One way to proceed with a sensible mapping is to specify a desired fundamental frequency  $f$  and an octave scaling  $s$ . Writing our audible frequencies as  $f_1, \dots, f_r$ , we can define our mapping from singular values to audible frequencies as follows:

$$\begin{aligned} f_i &= f \cdot \left( \frac{\sigma_i^{-1}}{\sigma_{\max}^{-1}} \right)^{\frac{1}{s}} \\ &= f \cdot \left( \frac{\sigma_{\max}}{\sigma_i} \right)^{\frac{1}{s}}, \quad i = 1, \dots, r. \end{aligned} \tag{1}$$

The effect of this remapping can be seen in Figure 2b, where we have used a fundamental frequency  $f = 128 \text{ Hz}$  and an octave scaling of  $s = 6$ . The spectrum now begins at the fundamental,  $f = 128 \text{ Hz}$ , and ranges up to a maximum of approximately 14000 Hz, an acceptable spread.

With an audible spectrum in hand, we now turn to the problem of choosing amplitudes for each individual frequency. These can be mapped from a corresponding subspace vector  $\mathbf{q} \in S$ , as each of its  $r$  components,  $q_1, \dots, q_r$  can be thought of as an amplitude for the  $r$  corresponding frequencies  $f_1, \dots, f_r$ . Care must be taken here too, as the overall sum of all the frequencies can exceed unity gain, and thus lead to clipping. Mathematically, this means that we require the  $L^1$  norm of  $\mathbf{q}$ ,  $|\mathbf{q}|_1 = \sum_{i=1}^r |q_i|$ , to be at most 1. A typical subspace vector  $\mathbf{q}$  may also contain negative components. However, these can simply be thought of as encoding a positive amplitude and a reversal of phase. The phase reversal can be discarded, as its perceivable effect is typically undesirable clicking artifacts.

With these considerations in hand, we design a mapping from a subspace vector  $\mathbf{q}$  to an amplitude vector  $\mathbf{a}$  of unit  $L^1$  norm as follows:

$$a_i = \frac{|q_i|}{|\mathbf{q}|_1}, \quad i = 1, \dots, r. \tag{2}$$

Given a sequence  $(\mathbf{q}_j)$  of vectors that describe a sampled trajectory  $\mathbf{q}_1, \dots, \mathbf{q}_T$  through the subspace  $S$ , we can either normalize each one of the corresponding amplitude vectors on an individual basis, or we can determine the subspace vector of maximum  $L^1$  norm,  $\mathbf{q}_{\max}$ , and normalize each amplitude vector based on  $|\mathbf{q}_{\max}|_1$ . The effect of the former is a relatively uniform volume level, while the effect of the latter is usually a significant audible decay from the first time step to the last, as in general, the first few  $\mathbf{q}$  vectors have a

significantly higher  $L^1$  norm than the remaining. Each approach thus has its musical merit, depending on the compositional situation.

## Synthesis

With our mappings carried out, we can now produce sensible audible sound. The most basic technique to try is additive synthesis—namely, mixing together pure sine tones at the corresponding frequencies and amplitudes, creating an overall sound with a rich spectral content. With an instrument in hand that can produce the corresponding sound given a set of input amplitudes (SuperCollider’s ‘DynKlang’ unit generator is a suitable choice), a particular trajectory  $\phi: [0, 1] \rightarrow S$  through the subspace, while retaining the same set of frequencies throughout time, determines a time-varying set of amplitudes, allowing the subspace evolution to dictate the time evolution of the sound. The overall effect is a subtle change in timbre, as the amplitudes of the various overtones fluctuate.

Subtractive synthesis is also an interesting option. We start with a spectrally rich input sound, such as an impulse or broadband noise. We then create a filter bank of resonances at the appropriate set of frequencies, weighted in turn by the appropriate set of amplitudes. (The SuperCollider unit generator ‘DynKlank’ is the point of departure on an implementation level.) Some extra parameters must be addressed, as each resonant mode also is given a ring time for the mode to decay by 60 dB. The nature of the input sound also strongly influences the resulting timbre—broadband noise creates a more atmospheric effect, while impulses can create driving rhythmic textures.

## Time evolution

Static sound, while interesting as a new timbre for a few seconds, eventually grows stale. Thus, we would like to capture the time evolution of the subspace trajectory as a dynamic sonic event. This can be achieved by cycling through the sequence of amplitude vectors  $\mathbf{a}_1, \dots, \mathbf{a}_T$  corresponding to the subspace trajectory  $\mathbf{q}_1, \dots, \mathbf{q}_T$ . Indeed, different trajectories through the subspace generated different sequences of amplitude vectors. The unfolding of these trajectories over time occurs on a micro-scale musically, but the choice of different possible abstract trajectories, and the shifting from one trajectory to another, is expressed on more of a meso- or even macro-level scale.

Experimentally, we have explored several different categories of subspace trajectories. The simplest is the original subspace re-simulation trajectory, which is faithful to the original simulation. Another possibility is the ‘frequency response’ trajectory, which is the result of setting the first subspace vector  $\mathbf{q}_1 = \mathbf{1}$ , the vector of all 1s, and allowing the simulation to unfold from there. Both of these possibilities retain the original physics-based dynamics to update each timestep. It is also possible, however, to abandon the physics and simply choose a trajectory by other means. For example, starting from the same initial vector  $\mathbf{q}_1$  as in the original re-simulation, we can determine a trajectory simply by walking in a random direction at each time step, or walking along the surface of some abstract manifold  $M \subset S$ , such as a sphere. These lead both to unique visual and aural results.

Since the set of frequencies itself remains static over time, and only the amplitudes are changing, musically, it will also make sense to explore dynamically changing the pitch  $f$  which was specified as the original fundamental frequency. This can be achieved simply by multiplying the entire spectrum by an interval, which will create a new fundamental without affecting the ratios between the partials. Alternatively, we could consider the original spectrum as a musical scale with which to compose melodies if we desire to compose more at the note scale.

