Artists’ Television Access (ATA), a small experimental media arts gallery located in San Francisco’s Mission district, is a haven for independent artists. The ATA’s film screenings, gallery showings and stage performances allow artists to explore modalities and propel their careers. One evening in March, I visited the ATA in search of something unexpected: ongoing science research. I watched as a small, fair-haired Norwegian woman stood on stage, flossing the strings of a massive double bass with a pearl necklace. Next to her, a bearded man dragged a dough knife across a drum-head. To their left, an intimidating bassist and a bespectacled guitarist fondled their instruments. In this dark place, filled with loud, chaotic sounds, San Francisco State University computer science professor William Hsu finds a world of improvisational music that is, to him, both beautiful and scientifically productive.
While the four musicians on the ATA stage built a musical conversation from unconventional sounds and techniques, a computer stage right, facilitating the complex animations being projected behind the performers. On a 20-foot tall screen, smoke danced in time with the music. Later, an amorphous evolution of dots vaguely reminiscent of a CAT scan filled the screen. Hsu’s “experiments” exploded visually, yet were calming; they were abstract, but also poigniant. Perhaps most fascinating, through some sophisticated computational alchemy, the musical sounds themselves influenced how the animations unfolded and panned across the screen.

During my evening at the ATA, I watched the musicians soften their playing to a whisper that lasted for almost 10 minutes. At the center of the screen behind them, orbs slowly rotated, emitting smoke. The clouds were like heavy fog, languidly swirling with the hypnotic effect of cream rotating and dissolving in coffee. After this quiet phase, the musicians suddenly began banging on their respective instruments, and the orbs exploded into a fast and frantic whirl around the screen. Hsu, however, had not touched his computer. Because the musicians were improvising, it would have been impossible to program any sort of deliberately timed visual accompaniment ahead of time. Clearly, the music itself had caused the images to change their shapes and movements. But how?

Hsu revealed some of his techniques, explaining that the process begins with the musicians. Microphones on stage feed live audio directly into his computer. Hsu’s software then breaks the music down into its separate features such as tempo, loudness, and roughness. These separate features are known as “audio descriptors.” If music were a face, the features would describe whether that face is long and skinny, short and round, or square and sharp. Hsu’s computer, performing what’s known as audio feature extraction, examines each of these descriptors individually.

Hsu uses the audio descriptors to influence the high-level behavior of the animation. As the tempo speeds up, the music crecscendos, or the sound roughens, the animation reacts by sometimes growing, shrinking, speeding up, or slowing down. It is often hard to perceive any direct reaction, however, as Hsu avoids an obvious cause-and-effect between sound and image. “The idea,” Hsu asserts, “is that as the musicians play, they see the animation changing, and that influences how they play. They’re interacting with the animation, but they can’t control specific details.”

The idea,” Hsu asserts, “is that as the musicians play, they see the animation changing, and that influences how they play. They’re interacting with the animation, but they can’t control specific details.”

“During my evening at the ATA, I watched the musicians soften their playing to a whisper that lasted for almost 10 minutes. At the center of the screen behind them, orbs slowly rotated, emitting smoke. The clouds were like heavy fog, languidly swirling with the hypnotic effect of cream rotating and dissolving in coffee. After this quiet phase, the musicians suddenly began banging on their respective instruments, and the orbs exploded into a fast and frantic whirl around the screen. Hsu, however, had not touched his computer. Because the musicians were improvising, it would have been impossible to program any sort of deliberately timed visual accompaniment ahead of time. Clearly, the music itself had caused the images to change their shapes and movements. But how?

Hsu revealed some of his techniques, explaining that the process begins with the musicians. Microphones on stage feed live audio directly into his computer. Hsu’s software then breaks the music down into its separate features such as tempo, loudness, and roughness. These separate features are known as “audio descriptors.” If music were a face, the features would describe whether that face is long and skinny, short and round, or square and sharp. Hsu’s computer, performing what’s known as audio feature extraction, examines each of these descriptors individually.

Hsu uses the audio descriptors to influence the high-level behavior of the animation. As the tempo speeds up, the music crescendos, or the sound roughens, the animation reacts by sometimes growing, shrinking, speeding up, or slowing down. It is often hard to perceive any direct reaction, however, as Hsu avoids an obvious cause-and-effect between sound and image. “The idea,” Hsu asserts, “is that as the musicians play, they see the animation changing, and that influences how they play. They’re interacting with the animation, but they can’t control specific details.”

Guo Moe, the female double bassist who wielded the pearl necklace, had previously worked with Hsu at the New Interfaces for Musical Expression conference in Oslo. She explains her experience of the music and computer imagery. “We like to take from the musicians in a positive feedback loop. The key to understanding it is “influence” not “control.” Synchrony is the only obvious connection. When the music stops, the animation eventually stops. When the music plays, worlds unfold on the screen.

Hsu’s computer animations are, at first glance, abstract. Their swirling, whooshing, evolving, and exploding suggest, but evade concrete images. At times, the abstract elements do congeal into definite shapes—skulls, symbols, moons—only to eventually fall apart. In some of the animations, Hsu employs Eulerian, or grid-based, fluid simulations to generate and propel the basic motion. “It’s like a pan of fluid,” Hsu muses, “and the idea is you divide the pan, the two-dimensional surface, into a grid. Then, you keep track-at each grid point (on the pan)—of the velocity and pressure as you stir the fluid.” Hsu grid-based simulation creates a very natural, organic looking animation. And just as in a pan of fluid, the main point of control is the fluid itself.

“I think what’s interesting,” Hsu says, “is there’s a complex process going on in here that results in hopefully interesting complex visual results. I can kind of influence it, but I can’t control all the details.” Hsu can “stir” the fluid, for example, and set up tidal currents, or place objects in the “pan” such as sinkholes, or create events such as explosions. Once the fluid is stirred, however, the physics-based processes take control.

between the conscious and the unconscious, says Moe—a link improvisational music itself seeks to create. Throughout each performance, the musicians influence the animation and the animation influences the musicians in a positive feedback loop. The key to understanding it is “influence” not “control.” Synchrony is the only obvious connection. When the music stops, the animation eventually stops. When the music plays, worlds unfold on the screen.

Hsu’s computer animations are, at first glance, abstract. Their swirling, whooshing, evolving, and exploding suggest, but evade concrete images. At times, the abstract elements do congeal into definite shapes—skulls, symbols, moons—only to eventually fall apart. In some of the animations, Hsu employs Eulerian, or grid-based, fluid simulations to generate and propel the basic motion. “It’s like a pan of fluid,” Hsu muses, “and the idea is you divide the pan, the two-dimensional surface, into a grid. Then, you keep track-at each grid point (on the pan)—of the velocity and pressure as you stir the fluid.” Hsu grid-based simulation creates a very natural, organic looking animation. And just as in a pan of fluid, the main point of control is the fluid itself.

“I think what’s interesting,” Hsu says, “is there’s a complex process going on in here that results in hopefully interesting complex visual results. I can kind of influence it, but I can’t control all the details.” Hsu can “stir” the fluid, for example, and set up tidal currents, or place objects in the “pan” such as sinkholes, or create events such as explosions. Once the fluid is stirred, however, the physics-based processes take control.
Hsu concludes. "The emphasis for this kind of animation is not in figures, the background remains, but you do much less drawing. It's more like a flip-book, because it stays the same. As you flip through the changing images, the background remains static. Our ears perceive this newly generated pressure wave as sound.

To simulate any resonant surface being struck, such as two cymbals crashing, a computer scientist must simulate the cymbal with a two-dimensional grid of points. The computer must then simulate the ripples propagating through every grid point, and—as with the animations—it must update each grid point at a specific frame-rate. This sort of simulation is nothing new. But traditional implementations run so slowly that the computer cannot generate the resulting sound in real-time. Instead, it has to be recorded to a static sound file and played back. This playback of the static sound file afterwards is clearly neither dynamic nor interactive.

Hsu and Sosnick have been working to change this. Together, they have succeeded in rewriting the code for simulating resonant metallic plates to run in real-time on a Graphics Processing Unit, or GPU. According to Sosnick, "this use of a GPU is very unusual." Generally, computer code runs on the Central Processing Unit, or CPU. If you think of code as the instructions or steps in a recipe, a CPU is a very fast chef that can complete each step very quickly and finish the overall recipe in a short amount of time. A GPU, on the other-hand, acts more like hundreds of slower chefs, each with lower processing power than a CPU, but able to complete steps simultaneously. In other words, the CPU "chefs" can dice carrots and onions faster, but he must dice one vegetable at a time.

In the GPU "kitchen," by contrast, one chef can "dice carrots" while another simultaneously "dices onions." Hsu and Sosnick used the GPU to complete separate chunks of the code in parallel. The challenge was finding where to partition the code, since some steps must be completed before others, and then how to synchronize the action at crucial points. This was Hsu’s and Sosnick’s victory: By applying their understanding of the code and hardware, the two created a detailed, dynamic simulation of plates of different materials being struck. The result is a new development in real-time audio synthesis: “We are on the verge of a new instrument,” Sosnick says excitedly.

Hsu’s work nestles somewhere between art, computer engineering, and invention. Straddling the worlds of physics, computer science, and animation, he has explored new means for artistic expression. His experiments have helped blur the line between hard science and creative projects. At the Artists’ Television Access gallery, I searched amongst pastry tools and pearl necklaces for science in art. Instead, through William Hsu’s research, I found art in science.