The Immersive Computer-controlled Audio Sound Theater:
Experiments in multi-mode sound diffusion systems for electroacoustic music performance

Stephen David Beck, Joseph Patrick, Brian Willkie, and Kenley Malveaux
Laboratory for Creative Arts & Technologies, Center for Computation & Technology
Louisiana State University
sdbeck@lsu.edu, jpatrick@cct.lsu.edu, bwillk1@lsu.edu, kmalve1@lsu.edu

Abstract

Multi-channel sound diffusion has been an essential part of the electroacoustic music process from the very beginning of the genre. We see early experimentation in sound spatialization in Varèse's use of loudspeaker "paths" in Poéme Èlectronique, *Stockhausen's experiments in quad and cube-based speaker arrangements (Chadabe 1997), and Chowning's computational approach to sound diffusion in works like* Turenas *(Roads and Strawn 1985).*

Experimentation in deployment and usage of audio loudspeakers is seen in the loudspeaker orchestras at Bourges (Clozier 2001), GRM and Birmingham (Harrison 1999), Richard Moore's notion of loudspeakers as "windows" to the virtual world beyond (Moore 1989), and the development of standard configurations for multichannel audio in digital video (5.1, 6.1, 7.1, 10.2) (Stampfl and Dobler 2004). And the recent interest in Ambisonic recording technologies (Malham and Myatt 1995) has furthered the aesthetic goal of an immersive audio environments that place the listener in an alternative sonic world – a "virtual world" created by composers and sound artists.

*The Immersive Computer-controlled Audio Sound Theater (**ICAST**, pronounced "eye-cast") is a project that addresses the various needs of multiple performance modes through a computer-controlled loudspeaker environment that is malleable, adaptable and intuitive. The system also explores new and novel metaphors for sound control using both conventional and unconventional interfaces.*

ICAST is a hardware and software solution, based on a client-server computer control system, commodity audio interface hardware, and high-quality audio amplifiers and loudspeakers. What we have learned from this research has applications that go well beyond concert hall performances, and include virtual reality environments, cinema, and video games.

1 Introduction

Motivation for this project grew out of our experiences hosting the 2001 SEAMUS[1] National Conference, our work with the Electric LaTex[2] student conference and the multitude of concerts we produce annually. In preparation for these events, we found it difficult to accommodate the various modes of multichannel diffussion in electroacoustic performance requested by composers.

During the 2001 SEAMUS conference, about a third of the composers requested a speaker array that utilized point-source location or surround-sound mixing techniques, a third of the composers requested an acousmatic-style speaker array, and a third expressed no preference. And although we created two performance venues, one with a surround-sound speaker array and one with an acousmatic speaker array, composers often did not take advantage of the diffusion paradigms available to them, either because they did not have enough time to practice, or they did not know what to do with the performance mode through which their piece was presented.

In response to our experiences, we began designing a multi-mode diffusion system that would allow composers and diffusionists multiple methods for electroacoustic performance without reconfiguring the speaker array, mixing console or patch bay. The system can handle presentations in surround sound modes, *acousmatique* modes, or using Ambisonic sound fields. We have also been conceptualizing new control interfaces and diffusion metaphors that facilitate complex sound diffusion processes without using the "mixing console" paradigm.

This paper will describe the rationale for this project, specifics about its implementation, and the results from our

[1] Society for Electro-Acoustic Music in the United States

[2] Electric LaTex is an annual collaboration between electroacoustic music programs at LSU, the University of Texas - Austin, the University of North Texas, and Rice University

initial set of concerts using the system. Additionally, we will describe the foundations of new metaphors and interfaces for sound diffusion as they will be implemented in ICAST.

2 Concepts

While it is true that there are as many diffusion strategies for the performance of electroacoustic music as there are composers of electroacoustic music, approaches can be reduced to two basic modes: point-source surround or acousmatic diffusion. Ambisonics, a 4-channel audio recording technology, has also been utilized to produce multi-channel sound diffusions. But the mixing console metaphor (faders controlling individual speakers) does not easily permit the real-time manipulation of Ambisonic fields for the purposes of sound movement or localization.

Our goal with ICAST is to engage multiple modes of sound diffusion for the purpose of electroacoustic music concert performance. Ultimately, ICAST will adapt and enable many diffusion modes simultaneously in the same concert, without repositioning speakers, or reconfiguring cables.

2.1 Modes of Diffusion

Point-source surround is rooted in the notion that sound objects exist in a defined and unique position in space, and can be positioned as a point-source through the appropriate level balancing between speakers. Speaker placements for these surround environments are usually in symmetrical layouts, with speakers roughly equidistant from each other, and surrounding the audience in a roughly uniform manner.

Acousmatic diffusion (Harrison 1999) (or *Acousmatique*) considers not only the radial or vertical position of a sound, but also the size (width, height, volume) of a sound. Sounds are diffused using large "loudspeaker orchestras" which are configured as speaker-pairs that are arranged in different parts of the concert hall. A small speaker-pair placed closed together creates a narrowly focused sound, while a large speaker-pair creates a much wider sonic image. This approach is commonly used in Europe and Canada, and is slowly gaining popularity in the United States.

Ambisonics recordings (Malham and Myatt 1995) capture not only the instantaneous amplitude information of standard audio recordings, but phase and spatial information as well. The sound field can then be recreated using any number of speakers placed in symmetric (or quasi-symmetric, i.e. 5.1) configurations. Simple calculations can rotate and expand soundfields around their center point, creating effective illusions of spatial movement and distance. Mono sounds can also be placed in Ambisonic soundfields through a calculation that simulates the phase and spatial information based on arbitrary angular and radial distance.

ICAST has been designed to handle sound diffusion within each of these diffusion modes. As we continue our research, we will be exploring how these modes can be used simultaneously, as well as the controllers that will be required to do so. Moreover, we are looking at new modes and metaphors for sound diffusion that build upon our existing research.

2.2 Previous Work

Prior to ICAST, we developed a 14-channel audio diffusion system that we called *The Howler*[3]. Funded in part by a grant from the Louisiana Board of Regents, this system was strictly an analog system, based around a Mackie 24 • 8 mixing console. Loudspeakers were set-up in an acousmatic fashion, with stereo pairs placed as near and wide mains, upstage speakers, side and rear speakers and floor monitors that were pointed at the ceiling to create virtual overhead speakers.

Figure 1: Photograph of the LSU School of Music Recital Hall with the front part of *The Howler*.

After having used *The Howler* for over 4 years, we began to document critical issues we wanted resolved in a computer-controlled system. We found that it was very difficult to accommodate 8-channel works in our Howler system, as the acousmatic layout simply did not account for 8 equidistant loudspeakers along the perimeter. And even when we did position speakers in an embedded 8-channel circle within the Howler, we found it very difficult to manually maneuver individual sound objects effectively around the circle without some additional hardware (i.e. a Kyma or Max/MSP patch). Additionally, we were becoming increasingly interested in

[3]after the object in *Harry Potter and the Sorcerer's Stone*

the developments in Ambisonics as a means of soundfield reproduction and simulation (Austin 2001).

It became clear that we needed to rethink our sound diffusion strategy, and move forward toward an alternative that could encompass point-source, acousmatic, and ambisonic strategies. Funding from the Center for Computation & Technology (CCT) and the CCT Laboratory for Creative Arts & Technology provided the basis for developing and implementing this project.

3 Implementation

ICAST is a multi-phase project, where we have set incremental goals over a two year period. Phase I was the acquisition of audio hardware, loudspeakers, and related materials required to assemble the audio delivery portion of the system. Phase II was our initial software development where we implemented a digital mixing system so that we could begin to use the diffusion system in concerts. We are currently in Phase III, where we are developing software and integrating new interface technologies to deploy new and novel diffusion modes.

3.1 Audio Hardware

ICAST is supported with a 24 loudspeaker array, built with BagEnd TA5000 and TA2000 loudspeakers, Stewart PM1000 amplifiers, and a BagEnd INFRA subwoofer system. This hardware gives us a relatively flat frequency response down to 8 Hz[4]. ICAST drives the loudspeaker array through 3 Mark of the Unicorn (MOTU) HD192 audio interfaces which are connected to the main ICAST computer. A MOTU 2408mk3 audio interface is used for audio input, and a Hammerfall RME ADI-8 DD provides digital I/O for the system.

Our choices for the loudspeaker were made based on our previous experience with the combination of BagEnd speakers and Stewart amps, as well as a careful consideration of price/performance/durability issues. The MOTUs were chosen because it was the only solution that would give us expandability up to 48 channels of simultaneous audio I/O at 192 kHz sampling rate through a single computer. We looked at a range of audio I/O options at 192kHz, 24-bit. But again, price/performance considerations led us to the MOTU solution.

3.2 Client/Server Architecture

The ICAST system is built on a client/server model, where the human control interface is managed by one computer which then sends control signals to the audio server, which processes and distributes the audio signals. We chose this method for two reasons: we wanted to use the MOTU HD192 audio interfaces, and they require a computer with PCI slots, and we wanted to hide the computer and audio equipment to minimize noise in the concert space. By using a lightweight laptop computer as a control client, we are able to put the audio server almost anywhere and still have robust control over the audio diffusion process.

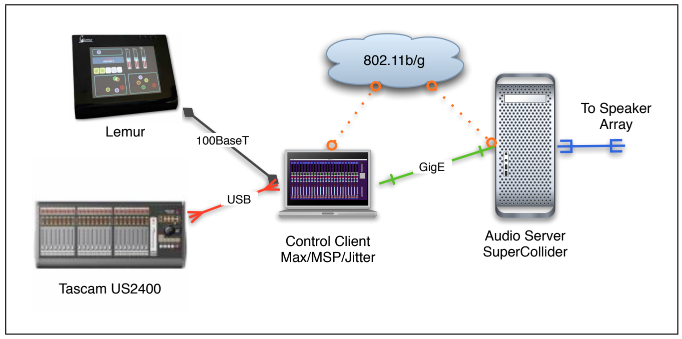

Figure 2: The client/server architecture for ICAST. Communication between client and server computers can use either wired or wireless Ethernet.

The client computer runs a custom Max/MSP (Zicarelli 1997) patch designed to emulate a mixing console. A TASCAM US-2400 fader controller serves as a haptic interface for the diffusionist, and we are beginning to experiment with a Lemur multi-point touch screen as an alternative controller. The diffusion application communicates with the audio server using OpenSoundControl (Wright and Freed 1997) (OSC) over a 1GB network connection. On the server, SuperCollider (McCartney 1996) is used to manage audio distribution and Ambisonic processing.

3.3 Client Software

We created a Max/MSP patch that emulated the TASCAM US-2400 fader board, and translated the fader board movement into actions sent to the server software. The TASCAM board sends MIDI messages over USB[5] which we then processed with Max. Once inside Max/MSP, the messages are visually represented on screen, formatted for OSC, and then sent to the server computer. Messages from the screen are also passed back out to the US-2400 to reflect any changes made with a mouse.

The US-2400 has several modes that tailor its operations to one of several commercially available DAWs. Initially, we used the Logic Audio mode because it closely matched out

[4] http://www.bagend.com

[5] http://www.tascam.com/products/us-2400.html

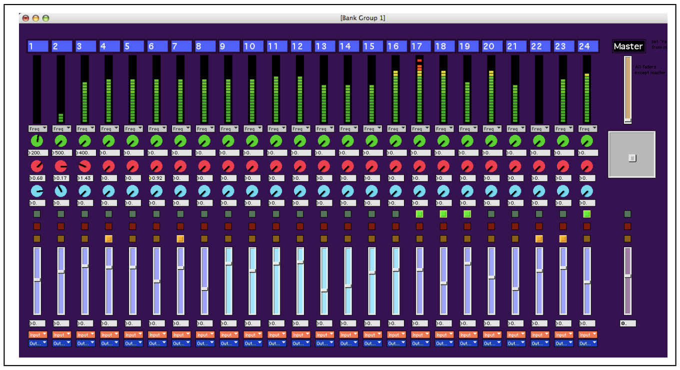

Figure 3: ICAST user interface in Max/MSP. The UI was intentionally designed to mirror the TASCAM US-2400 mixing console. Note the XY controller on the right, which maps to rotation of the ambisonic field.

desired mapping scheme. However, limitations within this mode have caused us to switch to the Native Mode of the US-2400, allowing us more complete and flexible control of the device.

Our basic scheme for the US-2400 is that each fader controls one output or one input, emulating the standard analog mixing board. That said, it is a simple extension to have one fader control multiple outputs (or other parameters) by assigning its control messages to multiple output modules running on the SuperCollider server.

In another scenario, faders are toggled between different modes, representing different control values. In cycling between these different representations, motorized faders on the US-2400 are synchronized to the new value for that mode. In one example, the Max client patch assigns Master Volume and Subwoofer functions to a single fader. The user toggles a button above the fader to choose between the functions.

We did run into a complicated situation in using the joystick on the US-2400. We routed the joystick position to the angular and radial position of a mono or stereo sound source within the Ambisonic sound field. Despite the physical interface being circular, output from the US-2400 was that of a square in Cartesian space. Because the Ambisonic synthdef uses polar coordinates, the Max patch converts the joystick data into polar space.

In actual use, we found that our users expected the rim of the joystick to represent the edge of the unit circle, even though that position in Cartesian space was beyond the circle. As a solution, we scaled the radius value to the longest possible length along its angular position. For example, the angular position $\frac{\pi}{2}$ points points straight up and has a length of the unit radius. However, the angular position $\frac{3\pi}{4}$ has a radial length of $\sqrt{2}$(radius). We scale the radial position by this angular distortion to properly map the Cartesian information onto the more useful Polar environment.

3.4 Server Software

The server software was written in SuperCollider. SuperCollider serves primarily as the signal router for ICAST. Several types of *synthdefs* had to be developed to accomplish this. The most fundamental and ubiquitous synthdef is the oneToOne.sc. It maps one input to one output and allows the user the flexibillity to dynamically remap either input, output, or both, as well as dynamically adjust amplitude, filtering, mute, etc.

The entire server is built from a small number of simple, modular synthdefs. The simplicity of this design along with the inherent flexibility in routing signals within SuperCollider has facilitated our rapid development of the server software.

Ambisonics Implementation Beyond basic audio I/O, SuperCollider is responsible for more involved forms of sound routing. One example of this is the task of encoder/decoder for more complex forms of diffusion such as Ambisonics. The **Ambisonic B-format encoder and decoder** implemented in ICAST are capable of receiving user input in the form of spherical coordinates, properly mixing the audio output by rotation of the Ambisonic field such that the sound appears to be rendered at the specified point on, inside or outside of an imaginary unit sphere.[6]

The **Ambisonic encoder** takes the arguments from the user that allow for a mono point source to be placed at a specific location in or around the unit sphere and encodes that point source into B-Format audio. The mono input is transformed into the four signals (X, Y, Z, and W) of B-format audio using equations 1–4, where A is the angular position (counter-clockwise) and B is the elevation. (Gerzon 1983)

$$X = \cos(A)\cos(B) * signal_{input} \quad (1)$$
$$Y = \sin(A)\cos(B) * signal_{input} \quad (2)$$
$$Z = \sin(B) * signal_{input} \quad (3)$$
$$W = 0.707 * signal_{input} \quad (4)$$

Here, equation 1 calculates the front↔back signal, equation 2 the left↔right signal, equation 3 the up↔down signal, and equation 4 the overall pressure signal.

These equations serve to simulate an Ambisonic microphone picking up a mono sound at the point represented by the angles A and B on the surface of a unit sphere. The calculations are slightly different when the radius ρ is added into the input arguments.

[6] We use the Ambisonic encoding and decoding UGENs for SuperCollider developed by Joshua Parmenter. `http://www2.realizedsound.net:8080/josh/papers.html`

The **Ambisonic decoder** takes the B-format signals and translates them into speaker feeds for a given speaker array. An example speaker array with which we have been experimenting places speakers in a cube configuration where the locations would be left front upper, right front upper, right back upper, left back upper, left front lower, right front lower, right back lower, left back lower; eight speakers in total. The decoder allows for the specification of such an array by taking in parameters such as speaker azimuth (ϕ), elevation (ξ) and distance as measured from the center of the listening area. The speaker feeds are decoded in a manner that satisfies the Makita and energy vector theories of sound localization.

$$LFU = W' + \frac{k\sqrt{2}}{\cos\xi * \cos\phi}X' + \frac{k\sqrt{2}}{\cos\xi * \sin\phi}Y' + \frac{1}{\sin\xi}Z' \quad (5)$$

Equation 5 is the calculation of the speaker feed for the left front upper speaker (assuming all the speakers were placed at the same distance from the center). The variable k is a frequency dependent constant, and W', X', Y', Z' are shelf-filtered versions of the original B-format signal.

$$\begin{pmatrix}\alpha_i\\ \beta_i\\ \gamma_i\end{pmatrix} = \frac{kmr}{\sqrt{2}}\left[\sum_{h=1}^{m}\begin{pmatrix} x_h^2 & x_h y_h & x_h z_h\\ x_h y_h & y_h^2 & y_h z_h\\ x_h z_h & y_h z_h & z_h^2\end{pmatrix}\right]^{-1}\begin{pmatrix}x_i\\ y_i\\ z_i\end{pmatrix} \quad (6)$$

$$S_i = W' + \alpha_i X' + \beta_i Y' + \gamma_i Z' \quad (7)$$

Equations 6 and 7 are the generalized calculations for an arbitrary speaker i with coordinates (x_i, y_i, z_i) (Gerzon 1983). Here, m is the number of speakers in the array, r is the distance of those speakers from the center of the array and S_i is the output signal to be routed to speaker i.

Using Parmenter's ugens, we can design Ambisonic environments from a range of speaker configurations. That is, we can place the speaker array in almost any configuration, noting the radial distance of the speaker, its angle from dead center, and its elevation angle, plug those values into ICAST and create a viable Ambisonic field. Ultimately, this will enable us to simultaneously diffuse sounds using multiple diffusion modes.

Other synthdefs Aside from its audio I/O and Ambisonic codec operations, SuperCollider performs to other tasks for ICAST. Several synthdefs were developed to handle the responsibility of loading sound files into a buffer and playing them through ICAST according to parameters and controls managed by user input devices (like the TASCAM US-2400). These allow the user to control things such as whether the playback of the buffer should loop, what position in the buffer playback should begin, how fast the buffer should playback, and routing information detailing where the buffer output is sent.

Another synthdef, monitor.sc, was developed to give ICAST users the ability to monitor channel amplitudes from the ICAST console. The audio output of any synthdef can be routed to a monitor which reports the RMS of its input to any client registered with the server for notifications..

SuperCollider and Max/MSP Issues Several issues came about in dealing with using SuperCollider and Max/MSP. One minor problem was with message passing between devices with OSC. The problem was not so much in the transmitting of messages to SuperCollider from Max, but, rather, the way that Max/MSP handles the return messages when querying the server about the status of a particular parameter within a node. For reasons still not fully understood, there seems to be a limit to the length of a parameter name that SC will report back to registered clients. It is not clear whether this issue is with SuperCollider, Max, OSC, or some combination thereof. The current work-around is to limit all of the parameter names in our synthdefs to six characters or less.

But these problems were nominal at best, and could be attributed to the issues with using OSC in Max. For the most part, issues of too much information, or too little bandwidth were easily resolved. Our experience is that both SuperCollider and Max/MSP are reasonably robust, giving us confidence that the ICAST system will remain functional, even after many hours of continuous use.

3.5 Deployment

Our first use of the ICAST system was in early December 2005, for a concert of computer music by LSU students. Although we only deployed 8.1 loudspeaker array, students found the diffusion system easy to learn and easy to use. In February 2006, we deployed a 16.1 channel configuration for a series of concerts as part of LSU's Festival of Contemporary Music. During these concerts, we processing both live sound and prerecorded sound in both acousmatic and ambisonic modes. In March, we will deploy the full complement of speakers (24.1) in a concert of electroacoustic music.

Figure 4 shows the speaker positioning that will be used for our first concert deployment of all 24 speakers. They are arranged so that a "circle 8" configuration is embedded within the acousmatic arrangement. The angled speakers upstage, center stage and along the side walls provide additional distance imaging. And we calculate the necessary Ambisonic coefficients based on the height, radial position and distance sound travels from each speaker to the center desk. We anticipate that we will need several concerts to fully optimize the system.

The concerts in February and March have been in the LSU School of Music's Recital Hall. Future concerts will experi-

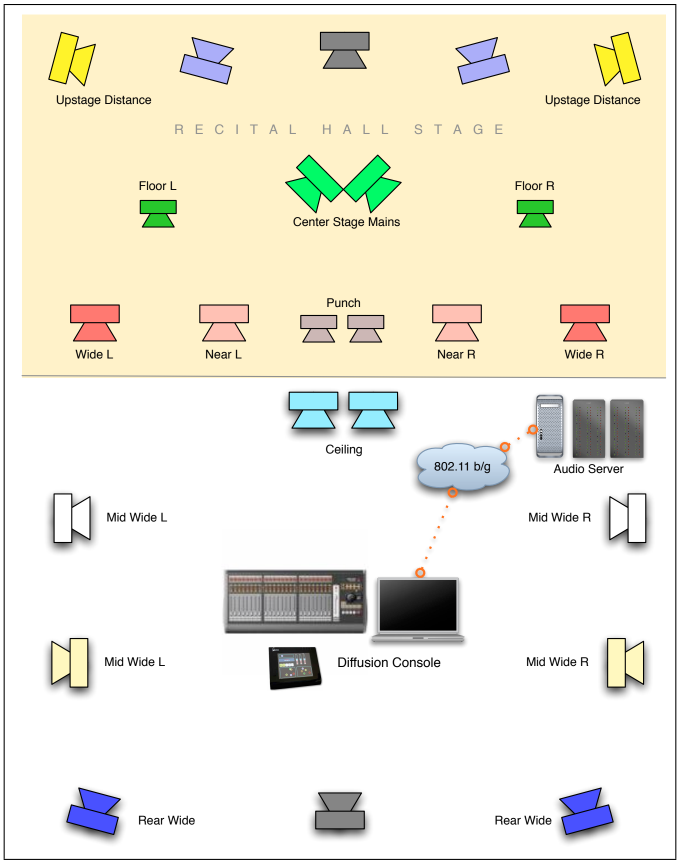

Figure 4: This diagram shows one possible configuration for the ICAST speakers.

ment with performance spaces at the new Shaw Center for the Arts in downtown Baton Rouge. Ultimately, we see ICAST as being a portable (although not trivially so) sound diffusion system that can be tuned to various halls in the region.

4 Future Directions

We are pursuing three primary directions for future development of the ICAST system. First, we are implementing some system updates to make the application more user friendly. This includes basic signal processing for each synthdef, so that we can implement standard forms of EQ for input and output channels; redesigning the Max interface so that it can automagically configure itself for different concert configurations[7]; autoloading of sound files into the system for playback.

Our second direction is the implementation of non-standard interface controls for manipulating diffusion. Several students are working on integrating the Lemur[8] multi-touch control surface into the ICAST system. Specifically, they are looking at how to move multiple channels of audio through the ambisonic diffusion space simultaneously. The *multiball* and *ring area* interface controls seem to provide the most interesting directions, and we are hopeful to integrate them soon.

Also, we are exploring the use of other non-traditional interface devices for controlling ICAST. Brygg Ullmer, an LSU faculty member in Computer Science, has been developing tangible devices and RFID-tagged objects for system navigation, file manipulation and data browsing of large data sets scattered across high-speed networks for the purposes of data visualization (Ullmer et al. 2005). We are beginning to explore how such devices might be adapted for use with the ICAST system, both in soundfile management, system configuration and manipulation of the Acousmatic soundfield.

Finally, we are looking at other metaphors for sound diffusion. In our preparation for this project, we theorized and proposed a new mode of sound diffusion, one we call *Sound Lighting*. This process takes its lead from theatrical lighting, where a series of lights are identified for a particular cue or scene. The cues are then sequenced through the theatrical performance, creating a lighting flow that sets the mood and helps tell the story.

In *Sound Lighting*, fader and signal processing settings of particular loudspeaker configurations are grouped and stored as vector fields, each field within the vector assigned to particular fader. One vector might represent distant sound from upstage. Another vector might represent a broad, wide diffusion. During performance, the diffusionist navigates the sound through various configuration vectors by using faders to interpolate between them. One could also expect that two-dimensional controllers, like Lemur, could also be applied in this fashion.

Our goal for this research year is to finish this diffusion mode, and to present a series of concerts which highlight the system's capabilities. We are also looking at generalizing the audio code so that we can use the system for data sonification applications, VR applications and cinematic applications.

5 Conclusions

Sound diffusion is the performance practice of electroacoustic music, and its importance as an art form is now becoming clear. Our experience in producing electroacoustic concerts has led us to understand that more comprehensive diffusion environments are needed to further performance op-

[7]That is, the system will adjust for different speaker deployments (16 vs. 24), as well as accommodate an arbitrary number of input channels

[8]`http://www.jazzmutant.com/`

portunities and develop a common set of technologies that can be deployed in other applications.

ICAST is an attempt to create a class of "diffusion instruments" that facilitate complex sound mixing through traditional and novel interfaces. We hope that our research will lead to powerful tools that enhance electroacoustic performance.

6 Acknowledgments

The authors wish to thank the Laboratory for Creative Arts & Technologies (LCAT) and the Center for Computation & Technology (CCT) at Louisiana State Univeristy for their support of this project. Funding for this project has come from direct allocations from LCAT as part of its initial infrastructure acquisition.

References

Austin, L. (2001). Sound diffusion in composition and performance practice II: An interview with Ambrose Field. *Computer Music Journal 21*(4), 21–30.

Chadabe, J. (1997). *Electric sound : the past and promise of electronic music*. Prentice Hall.

Clozier, C. (2001). The gmebaphone concept and the cybernéphone instrument. *Computer Music Journal 21*(4), 81–90.

Gerzon, M. (1983). *Decoders for Feeding Irregular Loudspearker Arrays*. United States Patent 4,414,430.

Harrison, J. (1999, 9). Diffusion: theories and practices, with particular reference to the beast system. *eContact 2.4*.

Malham, D. G. and A. Myatt (1995). 3d sound spatialization using ambisonic techniques. *Computer Music Journal 19*(4), 58–70.

McCartney, J. (1996). SuperCollider: a new real time synthesis language. In *Proceedings of the International Computer Music Conference*.

Moore, F. R. (1989). A general model for spatial processing of sounds. In C. Roads (Ed.), *The Music Machine*. MIT Press.

Roads, C. and J. Strawn (1985). *Foundations of Computer Music*. MIT Press.

Stampfl, P. and D. Dobler (2004). Enhancing three-dimensional vision with three-dimensional sound. In *SIGGRAPH Proceedings*.

Ullmer, B., S. D. Beck, E. Seidel, and S. Iyengar (2005). Development of "viz tangibles" and "viznet": Implementation for interactive visualization, simulation and collaboration. *NSF MRI Award CNS-0521559*.

Wright, M. and A. Freed (1997). OpenSoundControl: A new protocol for communicating with sound synthesizers. In *Proceedings of the International Computer Music Conference*.

Zicarelli, D. (1997). *Max/MSP Software*. San Francisco.