Virtual musical performance and improvisation on Internet2

MARA HELMUTH

College-Conservatory of Music Center for Computer Music, 282 Memorial Hall, University of Cincinnati, Cincinnati, OH 45221-0003, USA
E-mail: mara.helmuth@uc.edu

Several projects involving audio and video transmission between the University of Cincinnati and other sites over the high bandwidth Internet2 have ranged from the simple transmission of video material, to more complex interactive improvisation. A performance of *Clotho, the life of Camille Claudel*, a musical monodrama collaboration, was streamed in September 2000 over Internet2 to a widely distributed audience. The second project involved my real-time contribution from Cincinnati to a performance happening at Yale University of *The Ankle Diver*, with music by Matthew Suttor. A network improvisation project with the *Soundmesh* (formerly *Internet Sound Exchange*) software resulted in a number of improvisations involving the University of Cincinnati, Columbia University, Yale University and the Fall Internet2 Meeting at the University of Southern California. High-quality audio and flexibility of sound sources and processing is a strength of these improvisations.

1. INTRODUCTION

The Internet has brought vast resources in information, media and communication options to our personal computers in the last decade. The standard Internet1 allows compressed audio and video to be streamed, but there are often issues with image size and resolution, as well as audio quality. If the Internet is busy, the transmission rates also will cause media to pause or gap while playing. The recently developed high-bandwidth Internet2 allows for high quality audio and video to be transmitted in real time between remote hosts. The development of high-bandwidth network architecture infrastructures provides a new platform for live electronic music exchange.

Long before Internet2, John Bischoff, Rich Gold and Jim Horton (Bischoff 1978; Kahn 2004) founded the League of Automatic Music Composers for performing network improvisations in 1978. In more recent work in 1998, Seionshin Yamagishi and Kohji Setoh used Max to generate MIDI data, encoded to Real format and streamed through the Real Server and using David Zicarelli's The W Protocol (Yamagishi 1998). Keiji Hirata and Tasuya Aoyagi created a compositional system with a collaborative intelligent system using MIDI and the P^3 compositional tool for jazz musicians (Hirata 1999). Sergi Jorda's Faust Music On Line (FMOL) project, developed in 1997, stimulated collaborative composition using a real-time synthesis engine on the Internet (Jorda 1999). Phil Burk's TransJam Server allows musicians to perform together over the Internet using the JSyn Audio Synthesis API. Using either MIDI or real-time synthesis, these more recent systems explore remote collaboration or composition over Internet1. Tod Machover's Brain Opera (http://brainop.media.mit.edu/first-page.html) allowed participants to take part in an Internet event using Java applets for sound (Paradiso 1999). *Cathedral*, a web interactive work conceived by William Duckworth (De Ritis 1999) explores a nonlinear time domain and also incorporates listener interaction with MIDI-generated sound. These projects explore listener involvement in the compositional process and challenge the traditional composer/performer/audience paradigms. They also exploit various real-time synthesis methods or MIDI generation of sound to accommodate the lower bandwidth of Internet1.

Musical exchanges over Internet2 have linked audiences and performers distributed between educational institutions in the United States, to create multi-locational performances. The higher bandwidth allows better audio and visual quality transmission, allowing the use of uncompressed audio. Performers may interact through audio/visual displays of their musical partners. The nature of their exchange is different than when they are in the same room, not only because the short delay affects synchronisation, but also because, often having less immediate visual cues, performers must rely more on the music. Several projects described here were early explorations of remote performance and collaboration on Internet2, including (i) the distribution of a performance over the Internet in real time, (ii) collaboration over the Internet in programming composition and performance, and (iii) remote improvisation. *Clotho: the life of Camille Claudel*, an interactive drama collaboration between Dagmar Birke, Audrey Luna, Allen Otte and myself was streamed over the Internet live in September 2000. In 2002 I contributed to a performance of Matthew Suttor's music in *The Ankle Diver*, a dance-theatre production at Yale University. In an ongoing software project, Brad Garton, Douglas Scott and I

Organised Sound **10**(3): 201–207 © 2005 Cambridge University Press. Printed in the United Kingdom. doi:10.1017/S1355771805000944

https://doi.org/10.1017/S1355771805000944 Published online by Cambridge University Press

expanded the RTcmix music programming language to allow for score and sound sample transfer between hosts on the Internet through socket programming. Several improvisations using this software and the Soundmesh interfaces between Yale, Cincinnati, Columbia and an Internet2 meeting at USC gave participants unique interactive musical experiences. These projects provided opportunities for creative exchange over the Internet, and explored the nature of performance, improvisation and composition through the unique qualities of interaction inherent in each situation.

2. INTERNET2

Internet2 (http://www.internet2.edu/), a not-for-profit consortium of over two hundred universities, with some corporate partners, and the United States government, has deployed advanced network applications and technologies for enhanced research and collaboration since the late 1990s. Internet2's United States government partnerships include the Next Generation Internet initiative, with the National Science Foundation, the Department of Energy, and the National Aeronoautics & Space Administration. The Abilene advanced backbone network allows for 10 Gbps (gigabits per second) connectivity, although specific site connections may be slower. When the improvisation application project began, the University of Cincinnati had approximately a 155 Mbps bandwidth connection to the Abilene network. The connection recently increased to 540 Mbps, of which 100 Mbps is reserved for Internet1, and the rest for Internet2 (Morgan 2005). Some of the issues with transmission are latency, which can often be a 500 to 1,000 ms delay from a sending site to a receiving site across the United States, and jitter, the break-up of the smooth flow of signal.

3. HUMAN COMMUNICATION, PERFORMANCE AND IMPROVISATION CONTEXTS

Spontaneity and interactivity in music making has often been sacrificed in computer music genres in exchange for expanded sound worlds and exquisite timbral control. In the extreme case of fixed format, or 'tape' music, all sound is predetermined ahead of time and only the space and sound system determine variation in performance. Since the composer may take as long as s/he needs to create each portion of the work, craft, innovation and reflection play a large part in the compositional process, while processing time of algorithms is less significant. In real-time performance situations, acoustic and computer performers may improvise, allowing immediate exchanges of ideas, but this demands a powerful processor and efficient software strategies to operate in real time. This shift from 'tape' to real time in performance due to technological breakthrough is analogous to another shift in moving the actual 'stage' or recording studio onto the Internet, where performers in remote locations interact live. The nature of their exchange is different from when one sits in the same room, as one has more limited means of communication and must rely on the music rather than nods or looks to indicate changes. Perhaps this can be a good thing, encouraging more careful listening. Also, since one may be less familiar with a remote composer's work than the composer in the next office, when participating in remote improvisation one must be prepared to respond musically to a wider variety of inputs. If one receives unexpectedly different sounds from composers spread around the country, one may find uniquely appropriate ways to respond experimentally from one's repertoire of techniques and soundfiles.

4. INTERNET-STREAMED PERFORMANCE

The distribution of a non-interactive performance over high-bandwidth Internet is enhanced in two ways over previous modes: first, people in many locations can experience the performance, and second, the quality of transmission is higher than the highly compressed video possible on Internet1. This enhanced transmission connects audiences through a shared live performance. *Clotho: the Life of Camille Claudel*, an interactive drama for soprano, percussion and fixed-format audio, reveals the development and struggle for recognition of this remarkable sculptor, her love for Auguste Rodin, the destruction of her hypersensitive personality, and her decades in a mental asylum. *Clotho* is the name of one of her most famous works, a representation of one of the three Greek goddesses of fate. The drama was a collaboration between Dagmar Birke, contributing concept, text and direction, singer Audrey Luna, percussionist-composer Allen Otte and I, who composed the computer music. Tom Streeter streamed the riveting performance on Internet2 on 28 September 2000 from the Cohen Family Studio Theater at the University of Cincinnati College-Conservatory of Music (CCM). In 2000 there were four performances of the one-hour work *Clotho*, which after the significant artistic efforts by a number of people, were only seen by approximately two hundred people each of two nights in the Miami University concert hall, and one hundred and twenty-five people per night in the two CCM performances. While this was a larger audience than many new music concerts, on Internet2 the work was also heard by many more people across the United States. A web page with documentation of the project and downloadable video file resides at http://stremedia.uc.edu/clotho/, and figures 1 and 2 are stills taken from a

https://doi.org/10.1017/S1355771805000944 Published online by Cambridge University Press



Figure 1. Audrey Luna portrays Camille Claudel in *Clotho* in September 2000, in a performance at Cincinnati Conservatory's Studio Theater.



Figure 2. Allen Otte, percussionist-composer, bows fishing line in *Clotho.*

video of that performance. After the performance, downloads of the broadcast reached others, including a former student as far away as Singapore. Such distribution may be particularly important for non-commercial projects which may find a larger audience if widely distributed.

5. REMOTE INTERACTIVE COLLABORATION

While the distribution of performance over the Internet is significant, more challenging is actual collaboration in performance in real time over the Internet. From my office in Cincinnati, I participated in a performance of *The Ankle Diver*, a multimedia dance-theatre piece, written by Tim Acito, with music composed by Matthew Suttor and performed at the New Theater, Green Hall, at Yale University in December 2002. I received a stream of the ongoing performance on my computer, and, initiating events early to compensate for the latency, triggered some sounds in RTcmix to contribute to the music happening in Connecticut. I felt very engaged with the performance through my video connection, and experienced the enthusiasm of the audience at the end of the work as if I were there. However, as soon as the connection dropped, and my text communication with Matthew in a chat session ended, I realised I was alone in my office on a Saturday night. It was quite dark, as the sun had set during the performance. This sudden perceived environment shift from virtual performance to reality was a little shocking. The project with Suttor was one of my earlier attempts at using RTcmix across the Internet, and also involved programming some of the RTcmix socket sample transfer code, used later in the improvisation applications.

Matthew was interested in how well the synchronisation would work with the delay: 'some of the cues given over to your control depended upon sensitive and exact timing – this was a crucial test for our collaboration – and it worked splendidly!' He also commented, 'I find the concept of multi-site performance to be very exciting but also very familiar – it is after all a reflection of the way we work and interact in many situations in our lives today. Why shouldn't performance reflect this also?' (Suttor 2005). The demands of musical performance are more challenging to enact across physical locations, but the reward is to interact musically with those one finds most interesting, not just those who are nearby.

6. CMIX, RTCMIX AND SOUNDMESH

Paul Lansky's computer music language CMIX, like many others such as CSound and Cmusic, was written to generate fixed-format, or 'tape' compositions. In the 1990s many previously non-real-time oriented music programming languages began to incorporate the ability to synthesise in real time. At the 1997 International Computer Music Conference in Thessaloniki, Greece, Brad Garton demonstrated a version of



Figure 3. *The Ankle Diver*, produced at Yale University in 2002.

https://doi.org/10.1017/S1355771805000944 Published online by Cambridge University Press

RTcmix (Real Time Cmix), which could not only operate in real time, due to an event scheduler and faster processing of machines, but also could send 'score' or data files to other computers on the Internet to be performed. The transmission of scorefiles was accomplished with UNIX socket programming, allowing one host to transmit to any Internet IP address or hostname, on a particular port or socket on which the receiver was running a listen process. The computer with the appropriate port open, and a process listening to that port within RTcmix, would receive and play the scorefile. This feature was implemented primarily to distribute the interface processing to a different computer than the one doing synthesis, so that the musician could interact fluidly with the interface with a computer devoted to displaying and refreshing the GUI (interface) and algorithms, while a separate synthesis computer would devote its processing resources to audio. This ability to send scorefiles to remote hosts also could be used to send musical events across the Internet to any host, even if not in the same studio, creating a separation between the performer and the environment in which the performance was perceived by listeners. Remote playback also has humorous applications, such as starting up a playing process on a friend's computer, while you happen to be out of town. Musical exchange may be one of the strongest benefits, when being in the same location with a collaborator is impossible. Collaborators can still send sound back and forth, as long as they are running RTcmix, listening on the appropriate port. This exchange was accomplished in RTcmix by running a scorefile with command line arguments, and starting up the listener program in a UNIX terminal application (rather than a GUI) to send and receive sound.

While this new means of performing or improvising remotely was intriguing, I was concerned about a limitation: only algorithmic synthesis was available, not the use of recorded sounds. Musicians interacting in different locations might want to send sounds from their latest pieces back and forth, in new permutations and alterations. After hearing about the faster connection bandwidth of Internet2, Brad Garton and I discussed adding the ability to send and receive processed samples in real time. Brad programmed a prototype addition to RTcmix on IRIX, with a listener/player utility, and I expanded on this to make usage more robust, friendly and powerful. I ported my own version of RTcmix, trying to stay current with the Garton/Topper/John Gibson updates, to Linux (2000) and OSX (2002). This extension to RTcmix allowed the transmission of uncompressed sound from one host to another over requested sockets. Because improvising with text scorefiles and command line syntax was clumsy, I created the interfaces Internet Sound Exchange (ISX) in 2000 and Soundmesh in 2002. These interfaces allowed students and remote collaborators to send and receive sound during our improvisations. Participants could start up the socket listening program on various ports to receive sound, and could send selected sounds algorithmically modified as they chose.

In its most fully featured form, Soundmesh allowed one to load up to fifteen soundfiles, specify a remote host by IP address or hostname, initiate playing by hitting a button, and algorithmically window the sound, controlling window duration, amplitude and rate. On the receive side, one can turn on and off listening to various ports. Outside of Soundmesh one can also run RTcmix scores on the command line with processing limited only by the computational power of the sending machine. It is best to agree with participants which ports one will use ahead of time, so the particpant has more control over the resulting mix. Since processor power is the limiting factor, it is best to have a number of machines devoted to different tasks, perhaps some sending and some receiving sound. Quite complex host configurations can produce a situation where each participant chooses his/her sound input source hosts and destinations, experiencing a unique and changing set of interactions.

Matthew Suttor's class at Yale and my students in Cincinnati have had periodic improvisations in which we throw sounds from our latest compositions into a communal mix. Often people comment on the high-quality sound, and the unique situation of improvising with people a thousand miles away, with no visual cues. We might speak on the phone, or open a telnet/talk (chat) window, but otherwise depending on the music to know how to respond. I demonstrated this application to an audience at the University of Cincinnati's Internet2 meeting in 2000, interacting with Brad Garton at Columbia, and presented it at the 2000 International Computer Music Conference in Berlin. At a later demonstration at the 2001 Society of Electroacoustic Music in the United States conference, connections were defeated by a university firewall. Subsequent presentations were preceded by communication with network administrators to avoid this problem. I demonstrated Soundmesh on the Interfaced Cultures festival at Yale in May 2002, with graduate students Christopher Watts and others in Cincinnati sending sound to Matthew and me at Yale, and then at the 2002 Fall Internet2 Meeting at the University of Southern California, interacting with CCM students Maria Martin, Margaret Schedel and others, and Brad and Douglas Repetto at Columbia University. In these demonstrations, listeners and participants were often intrigued by an exchange with invisible musicians sending high-quality uncompressed sound from far away rather spontaneously. I set up a web page to encourage participation by others in our exchanges at

https://doi.org/10.1017/S1355771805000944 Published online by Cambridge University Press

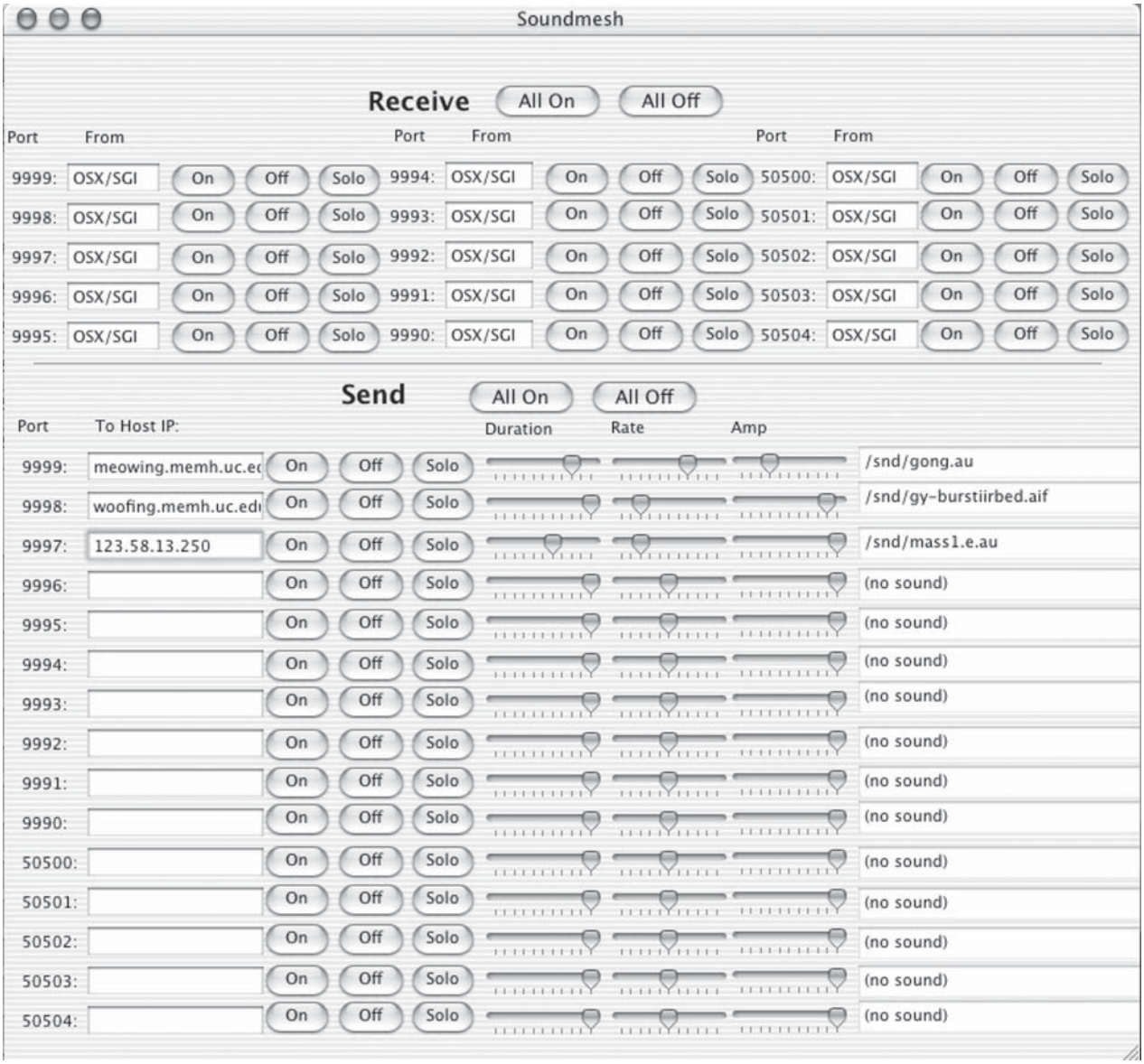


Figure 4. The Soundmesh interface in 2002. The receive panel allows a musician to listen on various ports; the send panel allows a musician to send a soundfile, with optional windowing, to another host.

http://meowing.memh.uc.edu/~mara/soundmesh/. For the most recent exchange, all of the RTcmix 'instruments' were recompiled with the net code, including those that take audio input, so we were able to process sound live with a microphone and send it to the remote improvisors, as well as work with soundfiles.

The most complex host configuration involved in a Soundmesh improvisation was probably between one machine at USC, three machines in Cincinnati, and several machines at Columbia University in 2002. While it can be satisfying to exchange sound with one or two others, theoretically any number of machines could participate, although they may not be powerful enough to interact with all of the other machines at the same time.

I often found the improvisations stimulating, due to the unexpected nature of the exchanges, and the challenge in responding musically to that input. Margaret Schedel described her experience: 'It is both exciting and disconcerting to improvise with Soundmesh – you lose control over your own computer, but the juxtaposition of disparate sound sources can be unexpectedly satisfying. The high audio quality is especially impressive when working with people half a continent away'. The 'losing control' aspect is probably due to the computer's processor power suddenly being taken up playing a remote improvisor's sound unexpectedly. Brad Garton said, 'What was really interesting was the simultaneous sense of "connection" and "disconnection" I felt while participating in the network performance. Here I was, sitting in my office at

https://doi.org/10.1017/S1355771805000944 Published online by Cambridge University Press

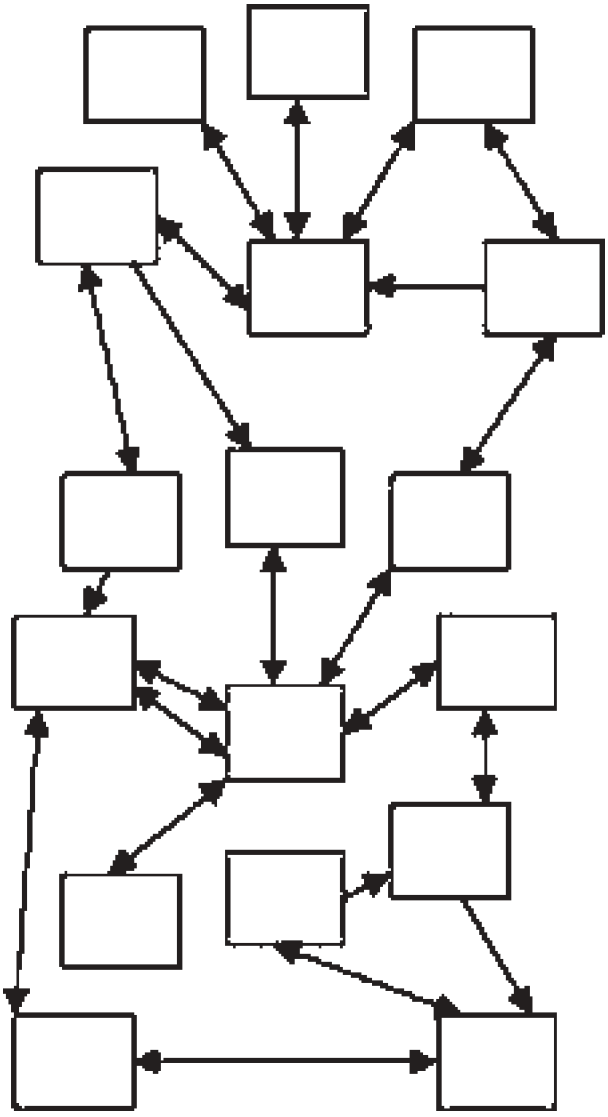


Figure 5. A possible complex host configuration for a Soundmesh improvisation.

Columbia, but at the same time performing in front of a large audience in an auditorium in Cincinnati. I felt that I was experiencing a new performance paradigm, a new kind of performer-audience interaction. As the technology develops to allow a larger bandwidth of exchange, and as savvy performers begin to explore this new system, I am certain that Paganinis and Presleys will arise, developing whole new ways of engaging with an audience. I hope that these will ultimately be human empowering, and not a technologically fuelled reinforcement of tired old performance cliches of the past'. With new instruments that are software/hardware configurations, the performer can develop interactive strategies particularly appropriate for certain digital signal processing techniques or their own physical skills. For example, a Cincinnati composer with an agile left foot may use it to control timbral spectral shifts on incoming sound from a California studio, and send it on to New York with a raising of a slider on the GUI. Since the sound worlds and performance gestures differ from classical techniques, the language of expression will be continually reinvented. Using the power of the Internet to distribute this music, evolving nuances of expression may spread and be understood quickly.

These 'human-empowering' ways of engaging with an audience, may result from more than the uniqueness of a first-time experience. In each situation I was aware that I was able to interact with people I had chosen, who would be most interested in this particular type of interaction. I was not limited by location or being in the same room/community/state, but could play with anyone near an Internet2 connection. Also, participants could select their own unique source sounds, which may have been shaped over long periods of time while composing, rather than depending on MIDI-generated sounds. Finally, since Soundmesh on Internet2 has a latency in the transmission time, one is forced to give up timed rhythmic synchronisation with other performers, and instead work with interactions of layers of sounds. These layers may be very freely composed, and the coordination in time relaxed which tends to create 'ambient' or at least timbrally oriented material, with gestures that unfold from combining layers. This is closer to the aesthetic of some fixed-format 'tape' music than instrumental performance, yet the spontaneity of the real-time responses generates a completely different performance situation. One works with blocks of recorded or script-generated material, which can be altered and preconfigured to allow on-the-spot choices in reponse to incoming sound. A complex configuration created before performance could allow as much control as in fixed-format music, as the available soundfiles and algorithms could be complex, but easily triggered within the GUI. Setting up families of related material, something many composers might already have after composing a piece, allows one to perform patterns and variations on previously used material. Creating a recognisable 'voice' from these related materials may enhance intelligibility in the absence of physical presence and consequently encourage meaningful interaction.

7. RECENT RTCMIX INTERNET DEVELOPMENT

Ivica Ico Bukvic ported Soundmesh and the underlying revised code to Linux with additions including a status bar with real-time updates on processes, compatibility with Perl and Python RTcmix scripting as well as MINC, and process control (Bukvic 2005). A status bar informs the participant of process updates. He also plans to add a configurable number of I/O streams, signal level monitoring, and real-time controls, as will be found in RTcmix 4.0.

https://doi.org/10.1017/S1355771805000944 Published online by Cambridge University Press

In 2004 I combined the OSX and Linux versions into code compatible with RTcmix 3.x, and John Gibson integrated it into the next version of RTcmix, 3.6. Douglas Scott was immediately inspired to encapsulate the code into a C++ object for RTcmix version 3.7. The listening NetAudioDevice can determine soundfile format, number of channels, and other parameters of the data, and has better error handling (Scott 2005).

8. CONCLUSION

Collaborative virtual performance connection projects challenge some traditional modes of performance: that performers must engage with each other and the audience in a particular space, that precise rhythmic synchronisation is a necessity for musical communication, and that familiarity with the performed music creates a better result. In each of these situations the space of performance and listening was multi-locational, connecting audiences through a shared experience. Enhanced audio improvisation with RTcmix and Soundmesh gave the participants opportunities to interact with remote participants without sacrificing audio quality or the use of recorded and processed sound. Also, the unexpectedness of unique, personal materials from a remote composer actually seemed to stimulate more interesting improvisations. The increased choices available to musicians and listeners on Internet2 can promote unique and provocative interactions.

REFERENCES

Bischoff, J., Gold, R., and Horton, J. 1978. Music for an interactive network of computers. *Computer Music Journal* **2**(3): 24–9. Reprinted in *The Foundations of Computer Music*, pp. 588– 600. Cambridge, MA: MIT Press, 1985.

Bukvic, I. I. 2005. Re: improvisations. . . E-mail to Mara Helmuth, 30 March 2005.

Burk, P. 2000. Jammin' on the Web – a new client/server architecture for multi-user musical performance. In *Proc. of the 2000 Int. Computer Music Conf*. San Francisco: International Computer Music Association.

Cameron, D. 1999. *Internet2: the Future of the Internet and other Next-generation Initiatives*. Charleston, SC: Computer Technology Research Corp.

De Ritis, A. 1999. *Cathedral*: an interactive work for the Web. *Proc. of the 1999 Int. Computer Music Conf.*, pp. 224–7. San Francisco: International Computer Music Association.

Garton, B. 2005. Re: improvisations. . . E-mail to Mara Helmuth, 31 March 2005.

Garton, B., and Topper, D. 1997. RTCmix – using CMIX in real time. *Proc. of the 1997 Int. Computer Music Conf.*, pp. 399–402. San Francisco: International Computer Music Association.

Helmuth, M. 2000. Sound exchange and performance on Internet2. *Proc. of the 2000 Int. Computer Music Conf.*, pp. 121–4. San Francisco: International Computer Music Association.

Hirata, K., and Aoyagi, T. 1999. Musically intelligent agent for composition and interactive performance. *Proc. of the 1999 Int. Computer Music Conf.*, pp. 167–70. San Francisco: International Computer Music Association.

Hub, The. 1989. *Computer Network Music.* Berkeley, CA: Artifact Recordings.

Jorda, S. 1999 Faust Music On Line: an approach to real-time collective composition on the Internet. *Computer Music Journal* **9**(1): 5–12(8)

Morgan, P. 2005. Re: net 2 speed. E-mail to Mara Helmuth, 31 March 2005.

Paradiso, J. A. 1999. The Brain Opera Technology: new instruments and gestural sensors for musical interaction and performance. *Journal of New Music Research* **28**(2): 130–49(20). Routledge.

Scott, D. 2005. Re: internet stuff. E-mail to Mara Helmuth, 28 March 2005.

Suttor, M. 2005. Re: net2, soundmesh, your theater piece. E-mail to Mara Helmuth, 31 March 2005.

Yamagishi, S., and Setoh, K. 1998. 'Variations for WWW': network music by MAX and the WWW. In *Proc. of the 1998 Int. Computer Music Conf.* San Francisco: International Computer Music Association.

https://doi.org/10.1017/S1355771805000944 Published online by Cambridge University Press