

---

# Interactive Network Performance: a dream worth dreaming?

---

AJAY KAPUR<sup>†</sup>, GE WANG, PHILIP DAVIDSON and PERRY R. COOK<sup>‡</sup>

Department of Computer Science (also <sup>‡</sup>Music), Princeton University, Princeton, New Jersey, USA

<sup>†</sup>Music Intelligence and Sound Technology Interdisciplinary Center (MISTIC), University of Victoria, Victoria, British Columbia, CANADA

<sup>†</sup>E-mail: [ajay@ece.uvic.ca](mailto:ajay@ece.uvic.ca)

**This paper questions and examines the validity and future of interactive network performance. The history of research in the area is described as well as experiments with our own system. Our custom-built networked framework, known as *GIGAPOPR*, transfers high-quality audio, video and MIDI data over a network connection to enable live musical performances to occur in two or more distinct locations. One of our first sensor-augmented Indian instruments, The *Electronic Dholak (EDholak)* is a multi-player networked percussion controller that is modelled after the traditional Indian Dholak. The *EDholaks* trigger sound, including samples and physical models, and visualisation, using our custom-built networked visualisation software, known as *veldt*.**

## 1. INTRODUCTION

Is the concept of musicians in multiple locations around the world performing together in real time using high speed Internet, with no latency, in front of live audiences a dream worth dreaming? Is there a valid point in researchers developing novel systems for networked performances, often spending large amounts of grant money to see this dream comes true? Is the music created using these systems worthy of being listened to, or should the performances be called 'live music'? Are the performers really interacting with each other over these long distances?

These are questions being asked by researchers who collaborate to create 'teleconcerts' or 'remote media events' as an application development project for their expensive Internet2 (USA) and CaNet (Canada) lines. Dreamers at Stanford University's CCRMA, McGill University, New York University, University of Southern California, Rensselaer Polytechnic Institute, the Electrotechnical Laboratory in Japan, and many other research facilities across the world have questioned and solved different pieces of the networked performance puzzle. Standing on the shoulders of these innovators, our team has created a new system for live network performance, to help answer some of the questions for ourselves.

In this paper we will present:

- The background to the development of networked media systems

- The design details of the *GIGAPOPR* networked media software
- The creation of the real-time *EDholak* multiplayer network controller
- The creation of *veldt*, a real-time networked visual feedback software that reacts to *Edholaks*, and other MIDI devices, in multiple locations
- The *Gigapop Ritual*, a live networked concert between Princeton University and McGill University, using *GIGAPOPR*, *EDholak* and *veldt*
- Concluding remarks and future applications of networked media systems

We will show our full methodology and design, as well as demonstrate that the implementation of high-quality, low-latency, live networked media performance can be straightforward and relatively inexpensive. Further, we evaluate the validity of the aesthetic of the network performance and whether the dream is indeed worth dreaming.

## 2. BACKGROUND

In the mid-1990s, a team at the Chukyo University of Toyota in Japan performed experiments using ISDN (128 kbps) to connect two or more concert venues with teleconferencing technology to allow musicians in remote locations to maintain 'musical eye-to-eye contact'. Later, in 1997, the team developed a system to use a low-bandwidth Internet connection, consisting of an Internet server relay which redirected musical messages, and musical synthesis software clients at three different venues. Each unique site controlled frequency values of a single oscillator, and performers on stage transmitted controller data to change their frequency in response to another site's change (Tanaka 2000).

In 1997, at the USA/Japan Inter-College Computer Music Festival in Tokyo, Japan, a team at Electro-technical Laboratory in Tsukuba, Japan, presented their work on a Remote Music Control Protocol (RMCP) that integrated MIDI and UDP protocol to allow users at separate workstations to play as an ensemble. This system also had visualisation feedback software, which reacted to what a user performed (Goto, Neyama and Muraoka 2000). The University of

San Diego and the University of Southern California collaborated in the Global Visual Music Project with the pieces *Lemma 1* and *Lemma 2* with an improvisatory jam session between Greece and the United States (Puckette, Sorensen and Steiger 1997) presented at the International Computer Music Conference in 1998. Presented all over the world in different versions, MIT's *Brain Opera* allowed observers to participate and contribute content to the performance via the Web (Paradiso 1999).

In September 1999, at the AES conference in New York, the Society's Technical Committee on Network Audio Systems demonstrated a live swing band originating at McGill University in Montreal, streamed over a multi-channel audio and simultaneous video connection (combination of UDP and TCP protocol) to a theatre at New York University, where on stage a dancer reacted to the high quality 48 kHz, 16 bit, surround sound music (Xu, Woszczyk, Settler, Pennycook, Rowe, Galanter, Bary, Martin, Corey and Cooperstock 2000). This system was later used to network concerts between Montreal and the University of Southern California, in Los Angeles and later Montreal and Japan.

In spring 2000, a team at Stanford University's CCRMA presented a networked concert between two multi-channel venues on campus, both with live audiences using the campus intranet and TCP protocol, to test whether an accurate sound image of the remote space could be projected locally. That summer, using the same system at Banff Center in Canada, ten-channel concert feeds from two concert halls were transported to a mixing room, and mixed down in real time (Chafe, Wilson, Leistikow, Chisholm and Scavone 2000). These systems use *SoundWIRE*, software that evaluates the reliability of a network by creating an 'acoustic ping' between the two host computers (Chafe and Leistikow 2001; Chafe, Wilson and Walling 2002). Later in 2004 this system was used to network three geographically distinct locations (California, Montana and Victoria) in a project entitled 'Distributed MahaVishnu Orchestra'.

The Integrated Media Systems Center at the University of Southern California (USC) has developed YIMA, an end-to-end architecture for real-time storage and playback of high-quality multi-channel audio and video streams over IP, as part of their Remote Media Immersion project. In October 2002, the team successfully broadcasted (16 channels of 24-bit 48 kHz samples per second audio and MPEG-2 720p formatted video at 45 Mb/s) a concert by the New World Symphony in Arlington, Virginia to an on-campus venue in Los Angeles, California. In September 2003, the system was tested internationally with a transmission to Inha University in South Korea (Shanhabi, Zimmermann, Fu and Yao 2002).

Other projects in the last few years have confronted and exploited different aspects of networked music.

*The Technophobe and the Madman* was an Internet2-distributed musical performance collaboration between New York University and Rensselaer Polytechnic Institute (Rowe and Rolnick 2004). *FMOL* (Jorda 1999) is a Virtual Music Instrument that was used between Dresden, Germany and Barcelona, Spain in 2001 (Jorda and Barbosa 2001). *PeerSynth* is a framework developed for peer-to-peer networked performance that makes use of latency as a parameter of synthesis (Stelkens 2003). *SoundMesh* is an application designed to mix audio files in a live Internet2 improvisation (Helmuth 2000). The *Auricle* website is a kind of audio analysis/synthesis enhanced chat room (Freeman *et al.* 2004). Most of these do not attempt to mimic live performance over distance directly.

For more information, the authors direct readers to Barbosa (2003), Föllmer (2002) and Weinberg (2002), which survey network systems for music and sonic art. Also, another innovative article is a 1998 AES white paper (Bargar *et al.* 1998) which forecasted visions of network media performance that have influenced most of the research presented.

### 3. GIGAPOPR: NETWORKED MEDIA PERFORMANCE FRAMEWORK

GIGAPOPR is a framework for low-latency, bi-directional network media performance over a high-bandwidth connection. It transmits multi-channel uncompressed audio, uncompressed video, and MIDI data among an arbitrary number of nodes. GIGAPOPR served as the software framework for the *Gigapop Ritual*, discussed in detail below.

#### 3.1. Challenges in design

##### 3.1.1. Latency

GIGAPOPR was designed to enable performers at geographically remote locations the ability to cooperate and interact with each other – recreating, as much as possible, the experience of playing together at the same place. Thus, one-way latency and round-trip latency both have critical effects on the quality of the interaction. Experiments conducted by CCRMA on quantifying the effects of latency in network performance show that humans perform best at round-trip bi-directional audio latency between 20 and 30 milliseconds (Gurevich, Chafe, Leslie and Tyan 2004). This was the toughest challenge our team had to face in building the framework.

##### 3.1.2. Network porridge

We define *network porridge* as any prolonged and perceptually significant audio artefact caused by some aspect of the network transmission. Network porridge, like the name suggests, is highly crackly and poppy audio resulting from one or more audio frames

failing to reach the destination machine in time. One common cause of network porridge is inconsistent delay introduced by the network during transmission, as a result of dropped or delayed packets (see section 3.1.4 below). Another cause may be contention between the network interface card (NIC), soundcard, and/or the CPU on the sending or receiving machine. For example, if the sending machine tries to send a large single chunk of data over the network (such as an entire frame of uncompressed video), it may 'tie up' the NIC and delay the transmission of several frames of audio, even when there is ample bandwidth to support both.

### 3.1.3. Compensation for different sound card clock speeds

Most sound cards have an onboard clocking mechanism, which times the capture/playback of audio data. However, it is often the case that the sound card on one machine may have a clock that is slightly faster or slower than another, even if both sound cards are the same model. If the difference in clock speeds of the sending and receiving machines is great enough, then eventual clicks (or porridge) may be introduced.

### 3.1.4. Robustness

The Internet is inherently a best-effort transmission system. Packets can be lost, duplicated, and/or reordered between the end hosts. Transmission control protocols (such as TCP) alleviate this issue by tracking and acknowledging packet delivery, and re-transmitting potentially lost packets. However, since audio data in a live-networked performance must take place in a highly timely manner, packet re-transmission is impractical. Therefore, a system should respond robustly and reasonably to potential network problems.

## 3.2. Design and implementation

### 3.2.1. Simplicity

The design and implementation of GIGAPOPR is straightforward, with only a few considerations and optimisations for low-latency, high-volume throughput. The framework is divided into three subgroups of applications, one each for audio, MIDI and video. Each group of applications is designed to run in a separate, autonomous process space. The challenge is finding a way to utilise the potential of the network in a real-time fashion.

### 3.2.2. Flow control and sequencing

All data packets are transmitted using the GIGAPOPR protocol over UDP. UDP provides

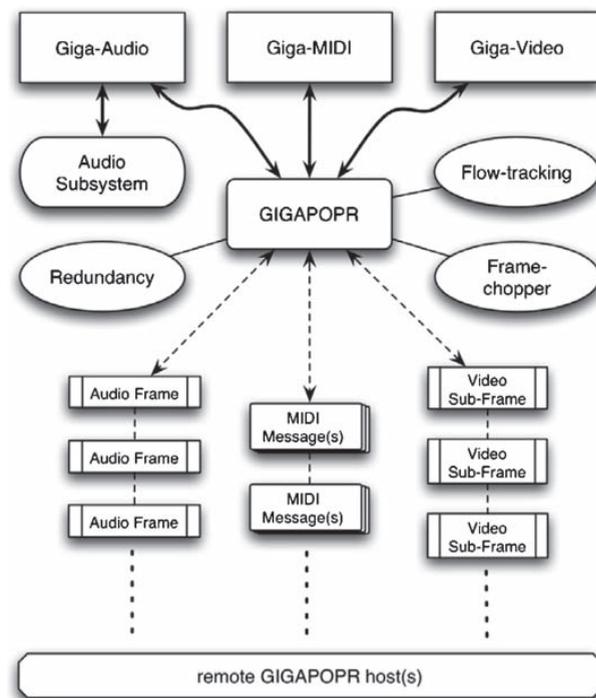


Figure 1. Flow control and sequencing of GIGAPOPR.

efficient and error-checked delivery of packets but is without flow control or congestion control. For our purposes, this is desirable since the system cannot afford to wait for re-transmission of lost audio packets (TCP-esque re-transmission is partly based on timeouts). If a packet is lost, then either the previous frame or silence is played. Furthermore, if the network is congested, there is little that an end-to-end connection can do. In this respect, we hope for the best from the bandwidth ceiling of a high-performance network. In our experience running over Internet2 and CA2Net, this was not a significant problem.

A sequence number is sent in the header of every GIGAPOPR audio packet. This simple sequence numbering scheme enforces ordering of incoming packets, allows the receiver to detect when packets were lost, and also makes possible redundant transmission of data. For example, it is possible for GIGAPOPR to send copies of each frame of data to increase the chance of at least one of the packets reaching the destination. Sequence numbering for video is more involved since it sends sub-frames.

### 3.3.3. giga\_audio

*giga\_audio* is a client/server application for capturing audio at one host and sending it with low latency to a remote host for playback. The mechanism is very straightforward. The capturer/sender application reads in frames of audio from the A/D converter and performs some minimal transformations on the

data (type-casting/Endian-adjustment) and encloses the data in a packet and sends it out using the transmission module. The size of the audio frame is adjustable. As is to be expected, larger frames will contribute to overall latency, while smaller frames may incur extra network overhead that can lead to dropped packets. For our performance, we used 48,000 Hz, stereo, with buffer sizes of 512 sample frames.

Additionally, redundant copies of each frame can be sent. The receiver/playback application receives the packets, performs simple sequence number checks (discarding out-of-date packets, and updating the next packet sequence number to expect) and also manages redundancy, if it is in use. It then pulls out the frames from each packet and sends them to the DAC for playback.

At the time of the performance, *giga\_audio* was implemented using the Synthesis ToolKit (STK) and RtAudio for capture/playback and over a custom transmission module over UDP.

#### 3.3.4. *giga\_midi*

*giga\_midi* is the MIDI counterpart of *giga\_audio*. The ‘midi in’/sender host sends one or more MIDI messages in a single packet to the receiver/‘midi out’ host. The MIDI data receiver can be mapped to onboard or external MIDI devices. *giga\_midi* was implemented over a custom module written with ALSA and also sent over UDP.

#### 3.3.5. *giga\_video*

The *giga\_video* application follows the client/server model used by *giga\_audio*. The video capture/sender application grabs video frames from any video source and sends it over UDP to the receiver/video playback application.

The design favours the timely transmission of audio and MIDI over that of video. Each video frame is actually sent in separate chunks, and sent with a small intentional delay between each one. This is to avoid tying up the NIC for a single large transmission, which might delay one or more audio packets from being sent on time. In GIGAPOPR, uncompressed  $480 \times 320$  video frames are segmented into 30–40 equal-sized chunks and sent separately.

#### 3.3.6. Configuration

We ran Linux (Redhat 9) with ALSA and the Planet-CCRMA<sup>1</sup> low-latency kernel. The audio/MIDI data were transmitted between two machines: Pentium 4/

2.8 GHz CPU/1 GB of RAM. The video transmission employed two additional machines: Pentium 3/400 MHz/128 MB of RAM. The real-time graphical feedback ran on a fifth machine: Pentium 3/800 MHz/512 MB of RAM, with GeForce 3 graphics card. Finally, a Pentium 3 laptop controlled additional devices on-stage.

### 3.4. Performance and optimisations

Perhaps the most striking reflection from our implementation of GIGAPOPR is that on today’s (and tomorrow’s) high-performance networks, it really doesn’t take much to get a high-quality bi-directional system up and running. For the most part, it suffices to have competence in implementing network and audio processing interfaces without introducing significant additional latency, and to know the right knobs to tweak. In this section, we discuss some factors that can greatly affect overall latency, as well as suggestions from our experience to reduce latency.

Several factors contribute to the overall audio latency of the system: (i) network latency between the source and destination hosts, (ii) end-host latency that involves buffering, context switching, processing, sending data to NIC, network stack processing, and the actual transmission time on the NIC’s hardware, and (iii) hardware latency of sound cards and the host machine itself.

The network between the end hosts is the least controllable aspect of the system, in today’s best-effort, end-to-end Internet. There is no direct way to even influence the routing of packets, or to avoid or respond to congestion. Until more programmable, dynamically routable networks become mainstream, we cross our fingers and leave these aspects to the underlying protocols and existing routing algorithms.

As for the end-host latency, we do have both direct and indirect control. Starting with the underlying operating system, it can be beneficial to install low-latency kernel patches (if running Linux) such as the one packaged with Planet-CCRMA. On MacOS X, setting the scheduling policies to round-robin for audio and network processing threads while keeping the rest as default first-in-first-out can significantly improve stability and latency for lower buffer sizes. Boosting process priority on both systems can also be helpful.

Finally, machine hardware and soundcard quality can have a big impact on latency and stability. For the machine itself, good bus performance is crucial, as audio I/O, network I/O, and often (depending on the architecture) memory operations may all contend for and share the bus. Yes, faster machines with more memory are good, too. Lastly, soundcard latency can vary vastly from model to model and across vendors. It is worthwhile to ensure all hosts have low-latency

<sup>1</sup><http://ccrma.stanford.edu/planetccrma/software/> (February 2005).

soundcards with appropriate configurations and settings.

At the time of the performance, we clocked between 120 and 160 ms round-trip latency between Princeton, NJ, and Montreal, Canada. We were able to perform using 120 ms latency, and did not implement all of the ‘performance tips’ mentioned above – many of them came out of subsequent experiments. We are optimistic that we can do better on today’s improving networks and from experiences we have gained since.

#### 4. THE ELECTRONIC DHOLAK CONTROLLER

##### 4.1. The traditional dholak of India

The dholak is a barrel-shaped hand drum originating in Northern India. It has two membranes on either side of the barrel, creating higher tones on the smaller end, and lower tones on the larger end (Kothari 1968). The smaller side has a simple, single-layer membrane, whereas the larger side has *Dholak masala* (a composition of tar, clay and sand) attached to the inside of the single-layer membrane, to lower the pitch and produce a well-defined tone. The dholak can be tuned in two ways depending on the type of drum. The traditional dholak is laced with rope, so tuning is controlled by adjusting a series of metal rings that determine the tightness of the rope. Modern dholaks have metal turnbuckles which are easily adjusted for desired tone. The dholak is widely used in the folk music of villages in India. It is common for folk musicians to build dholaks themselves from commonly available material. They then use the drums in musical rituals and special functions such as weddings, engagements and births (Sharma 1997).

Two musicians play the dholak. The first musician strikes the two membranes with their left and right hands. There are two basic playing techniques; the open hand method is for louder playing, while the controlled finger method is for articulate playing. There are a few different positions to play the dholak, but the most popular is squatting with the drum in front, the bass head on the left, and the treble head on the right. The second musician sits on the other side of the drum, facing the first musician. This second performer strikes the barrel with a hard object, such as a spoon or stick, giving rhythmic hits similar to a woodblock sound (Bagchee 1998).

##### 4.2. The electronic dholak controller

The design of the *Electronic Dholak* (Kapur, Davidson, Cook, Driessen and Schloss 2004) is inspired by the collaborative nature of the traditional drum. Two musicians play the *EDholak*, the first striking both heads of the double-sided drum, and the second keeping time with a ‘*Digital Spoon*’ and manipulating the

sounds of the first player with custom-built controls on the barrel of the drum and in software. We further explored multiplayer controllers by networking three drummers playing two *EDholaks* at two geographically diverse sites.

Finger strikes are captured by five piezo sensors (three for the right hand and two for the left hand) which are stuck directly on the *EDholak*’s drum skins. Sensors are placed in positions that correlate to traditional Indian drumming. The left drum-skin head captures *Ga* and *Ka* strokes, while the right hand drum-skin captures *Na*, *Ta* and *Ti* strokes (Kapur, Essl, Davidson and Cook 2003).

The *Digital Spoon* has a piezo sensor attached to the back of a flat wooden spoon. There is neoprene

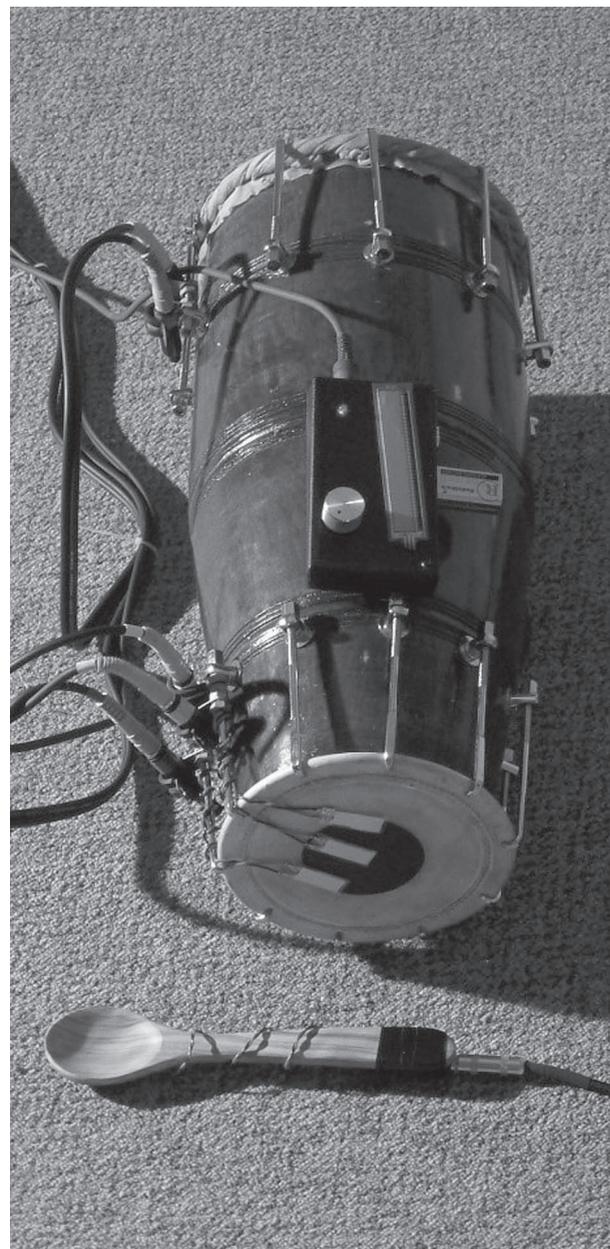


Figure 2. The Electronic Dholak controller.

padding covering the piezo to keep the striking of the *Digital Spoon* acoustically quiet. The spoon player has the option of striking anywhere on the drum, or floor, triggering an audio/visual response, or striking on a linear force sensing resistor (FSR) on the *EDholak Controller Box*, which augments the audio/visual of the spoon strike and the audio/visual instances of all *EDholak* finger strikes. The *Controller Box* has a linear FSR and a knob that the spoon player can use with his left hand to augment all sounds/graphic instances.

All piezo triggers are converted to MIDI by the *Alesis D4<sup>2</sup>* 8-channel drum trigger box. The *Controller Box* is built using a Parallax Basic Stamp that converts all sensor data to MIDI. When two *EDholaks* are used in distinct locations, piezo-generated MIDI signals are transferred using *GIGAPOPR* and then processed and then merged together by an *Alesis D4*.

All MIDI messages are funnelled to the *EDholak MIDI Control Software* written for Windows. This software is used by the spoon player to control many parameters of the performance. A user can toggle between a networked performance (two *EDholaks* sending MIDI messages) or just one. The software is custom built to communicate with the *Roland Handsonic*.<sup>3</sup> The user can pre-program patches which they wish to use in performance, in order of occurrence, and simply use the mouse to switch between



Figure 4. EDholak: a two-person electronic drum.

them during a concert. The software also maps the *Control Box* sensors (Knob and FSR) to different MIDI Control Changes such as pitch, sweep, colour, pan and volume, augmenting sounds of piezo MIDI signals going to the *HandSonic*. For example, the performers can start out by playing traditional dholak sound samples while the spoon player selects a frequency sweep effect which morphs the samples to new expressive rhythmic sounds with the *Controller Box* and *Digital Spoon*. All MIDI messages from the software get transmitted to *veldt* to trigger visual events.

## 5. VELDT: NETWORKED VISUAL FEEDBACK SOFTWARE

The MIDI messages generated by *EDholak* drummers and spoon players are routed to a graphics computer running the *veldt* software, which synthesises visuals in response to the patterns of drum triggers and other controller messages. *veldt* is an application which was designed from the ground up for the purpose of visual expression and performance. It receives MIDI (Music Instrument Digital Interface) messages from digital musical interfaces and maps them to a system of reactive events in order to generate live visuals, which are rendered in real time using the OpenGL2 graphics language. Mappings are flexible: sets of mappings may be arranged and modified during the design and rehearsal process, and triggered by control events during different movements of a performance, and arbitrary text, images, video, and geometric models may be used as source material.

We display a real-time composition of these media sources over geometric elements which are generated and modified according to the parameters of the current mapping. In addition to control events received from the performer, a physical simulation environment is incorporated to allow for a variety of secondary motion effects. This visually (and contextually) rich combination of source material over physically reactive structural elements allows for a response that is dynamically generated and artistically controlled.

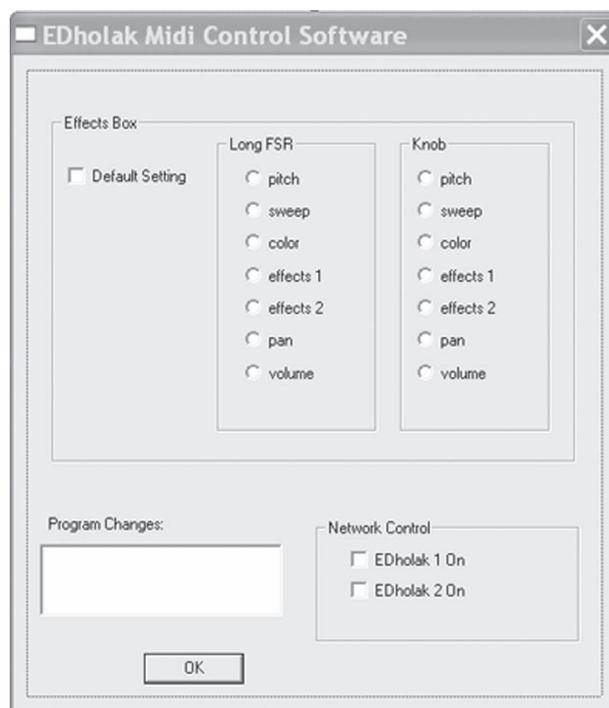


Figure 3. The Electronic Dholak MIDI control software.

<sup>2</sup>*Alesis D4 Reference Manual*, available at <http://www.alesis.com> (July 2004).

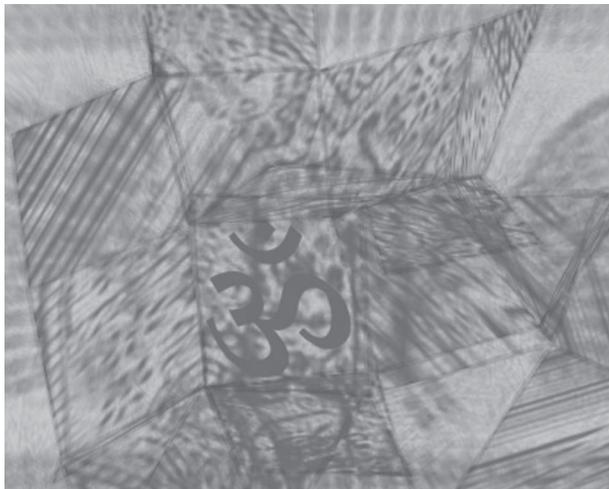
<sup>3</sup>*Roland Handsonic HPD-15 Reference Manual*, available at <http://www.roland.com> (July 2004).



**Figure 5.** Example screenshot of structure evolved from a drumming sequence generated by *veldt*.



**Figure 6.** Layering text elements (Hindi) over several sparse structures using *veldt*.



**Figure 7.** A more cohesive structure generated by a variation on the rule set.

While the parameters that govern the overall response of the system to the drum controllers may be modified through cues such as MIDI program change messages, *veldt* allows an additional visual performer to control the finer aspects of the performance.

## 6. THE GIGAPOP RITUAL

The *Gigapop Ritual*<sup>4</sup> was a live network performance between McGill University in Montreal, Canada, and Princeton University in New Jersey, USA. This live collaborative musical performance, weaving cyber electronics and Indian classical tradition, involved high-bandwidth, bi-directional real-time streaming of audio, video, and controller data from multiple sources and players at both sites, using the GIGAPOPR framework. In composing the piece we took into account a saying by Atau Tanaka: 'Latency is the acoustics of the Internet' (Tanaka 2000). We composed a piece that was appropriate for this aesthetic.

### 6.1. The composition

We composed the piece to explore multiple areas of performance over a network, using the traditional structure for North Indian classical music, as well as taking into account the framework of the network itself. The first section, known as *Alap*, was a slow call and response section between two melody-making instruments (sitar in McGill, Electric Violin in Princeton). These two performers left space for one another to interact to different improvised themes and phrases. The second section, known as *Gat*, was a pre-composed melody (based on Raga Jog and Jai Jai Vanti), over a structured eight-beat rhythmic cycle known as *Kherva* (performed on tabla in Princeton). The challenge and solution in performing a melody over the network was to have a leading side, and a following side. The round-trip latency of 120 ms was about the same as the echo one would hear from the back wall of a 60 foot room. Playing with other performers removed by 60 feet is somewhat common (in marching bands, antiphonal choirs, and other musical settings), and this experience was made only slightly more challenging by the large amount of equipment to be set up and tested. The performers in Princeton were the leaders, and once the data arrived in McGill, the Canadian performers simply played along, reacting to what they heard. The third section was a free-form improvisation where musicians explored the network performance space. Performers used their custom-built digital interfaces to create diverse computer-generated sounds, while still focusing on interacting with the performer on the other side of the network.

<sup>4</sup>See <http://gigapop.cs.princeton.edu/> for video and pictures of performance

Each performer left enough space for others to react, and no one played anything unless it was a response to a ‘call’ from another musician. Thus we were able to create a spiritual tie using the two host computers at two geographical locations, connecting performers in the Pollack Concert Hall of McGill University with performers in the Princeton Computer Science DisplayWall room for a 2003 New Interfaces for Musical Expression (NIME) Conference performance.

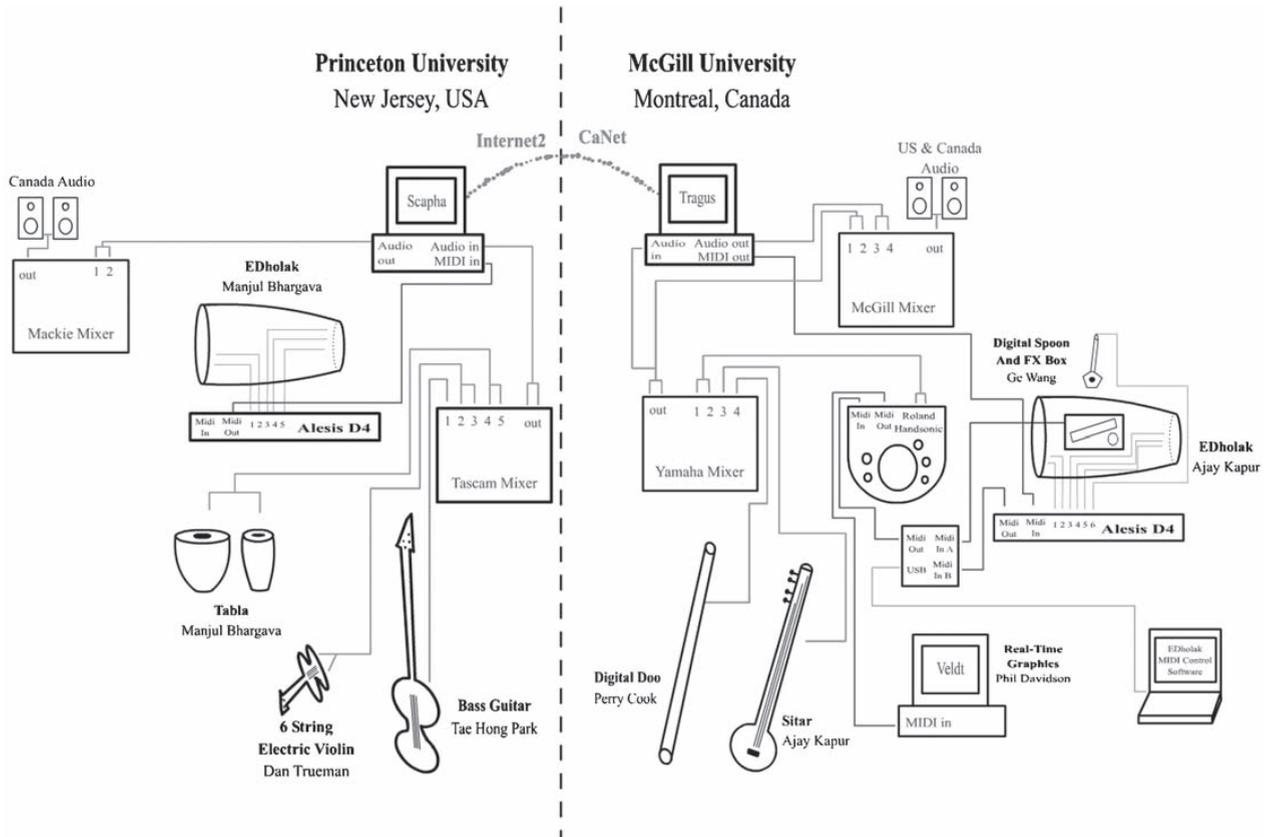
**6.2. veldt visual representation**

Our intent was to create an environment in which the actions of both drummers were visible and distinguishable. Our solution for this concert was to allow two players to interact through a sculptural metaphor. Using a dynamic geometry representation to allow modifications to the structures in real time, the two performers interacted through a series of operations to create a visual artefact of their drum patterns. Their strikes were dynamically mapped to a series of geometric operations that generated, deleted, deformed or detached elements of the structure and generated unique artefacts from the rhythms they played. In figures 5, 6 and 7 we see structures that have evolved



**Figure 9.** *Gigapop Ritual* live performance at McGill University with left screen showing live feed from Princeton University and right screen showing real-time visual feedback of *veldt*.

under different mapping rules. In figure 6, for example, we chose a mapping that created smaller, separate elements rather than building from a central structure as in figure 5. In figure 7, we chose rules which resulted in a solid, sheet-like structure. To add a convincing physical response to the addition and alteration of new elements, we used a mass-spring model to apply and



**Figure 8.** Diagram of *Gigapop Ritual* setup.

distribute forces as the structures developed. In these figures, the actions of the drummer bend and distort the figure, while secondary forces try to smooth and straighten the figure, like a plucked string which vibrates to rest.

To represent the shared performance space, we experimented with several different forms of visual ‘interaction’ between the signals received from two performance spaces. To begin, we assigned the two drummers to separate visual spaces: one drum would excite patterns as in the ETabla performance, while the second was assigned to build structures. We then assigned both performers as builders, so that their rhythms would build upon one another. In the next, one performer’s strikes would build while the strikes of the second would rattle or delete those structures.

## 7. CONCLUDING REMARKS AND THOUGHTS

The promise of interactive, multi-performer, networked performances, including audience participation, has been with us for quite a long time now. New research agendas have been born to technically enable these types of performances. Some projects have begun to look at the social aspects of this area as well. Our paper served to report about specific systems, a composition, and a performance. Moreover, we asked questions as to the motivations, reasons, necessity and validity, both artistic and aesthetic, of investing the significant time and money in order to perform in more than one place at once.

An interesting thing we discovered about networked audio/music is that it isn’t as technically difficult as it has been. The recent availability of Internet2, CA2Net and other optically based gigabit networks has made creating systems such as SoundWire and Gigapop rather simple. Honestly speaking, a good programmer with a standard networking textbook could implement our system. Performance system tweaking required quite a bit of experimentation, but when it came down to the performance itself, it worked fine. If it didn’t work, the failure would have been because some astronomer decided to ftp a terabyte of data, or the dining hall closing at some university between Princeton and McGill prompting 200 students to suddenly rush back to their dorm rooms and start downloading movies using BitTorrent, or some similar reason. The promise of guaranteed quality standards on our networks went away with the demise of ATM (in the US and Canada), so it seems that we are ‘stuck’ with very high bandwidth, but no guarantees against porridge.

One aspect of future systems, especially those based on our existing infrastructures, might include components of handshaking, where the multiple sites each announce and probe each other as to the available capabilities. In this way, networked audio might behave much as instant messaging, where each site

gives and receives what it can technically to the performance. Some sites might only send gestural/sensor data, minimal audio, and very low quality (or no) video, and synthesise a local audio performance based on minimal data from the other sites. Others might be able to provide and consume full-bandwidth uncompressed video, audio, and sensor data. The aesthetic issues surrounding these sorts of inhomogeneous, highly asymmetric systems are quite interesting for future research and study.

### 7.1. Good things about networked music performance

There are some good aspects to doing research in real-time networked sound, one of them being that sound is an excellent test-bed for testing network hardware and software. Latency and continuous quality of service is more important for sound than even for video. We can all tolerate a dropped or repeated video frame now and then, but not choppy audio. So in this way, sound and music are good for networking research and systems building, but this does not imply that networking is good for sound and music (except perhaps for funding opportunities).

Areas that will clearly become useful, once systems become commonplace and affordable, include applications in pedagogy, such as remote instruction, rehearsal, etc. The ability to rehearse remotely is also interesting for professionals in some cases. There are many cases of unique instruments that cannot be moved easily. Opting to do a rehearsal remotely from a studio (or one’s home) rather than flying to the ‘gig’ and back, sounds attractive, if the quality is good enough.

Another aspect that Tanaka and others have mentioned is that the network and systems will breed new aesthetics. So new forms of art and interaction that don’t fit the traditional performance, improvisatory, audience, etc. moulds might emerge, giving networked performance its own unique space in art.

### 7.2. Not so good things about networked music performance

Technically, as we have stated, existing networks do not provide guarantees of quality (delay or bandwidth), and we are fairly certain that for some time to come, any such guarantees would be very expensive to have if available. Internet2/CA2Net are expensive themselves, and available only to academics with lots of serious research to do. To think that the physics department will buy the music department a new gigabit router, and pay to rewire the concert halls with fibre, seems like pipe dreaming. So expense is still a serious issue for all but a few.

One concern we have is the loss of the identity of the ‘band’ itself, that is, the interaction of a finite number of players, each with their unique role, playing together

on a single stage. Of course this is to be considered a 'feature' as well as a potential bug, but is cause for concern, given the long history of musical performance in the more traditional moulds.

This tradition provides important grounding for audiences, who should also be considered in the future of new music performance. Contemporary composers and musicians have historically inflicted quite a bit of grief on their audiences (Babbitt 1958). In this tradition, we suppose that having a robot playing an instrument on stage in a concert location, along with grainy video of human players in a remote location, could be amusing or even aesthetically pleasing. But once the curiosity has worn off, the music and performance must stand on its own.

Related to this is the loss of society within the 'band', that is the interactions that go on between band members, both on and off stage. Waiting backstage to go on, and important aspects of socialisation after a performance, are not the same over a network. Being able to go out for a drink in Paris after a performance can be more important and memorable than the actual performance itself. And, that drink and performance in Paris can make the long airplane flight worth it as well.

### 7.3. A dream worth dreaming

Networked Media is a dream worth dreaming. The work completed by researchers so far comprises steps in the right direction on a path to a very uncertain destination. GIGAPOPR, the Edholak, and *veldt* are small pieces of a much bigger puzzle. Applications must be constructed, and allowed to evolve naturally, that can take advantage of the 'sound without space'.

Someday, musicians might be faced with a common decision of whether to sit at home in their fuzzy pyjamas and play concerts with others, or to travel to the site of the performance. The authors wonder if networked performances will be granted an artistic status as legitimate as more traditional musical endeavours.

### ACKNOWLEDGEMENTS

We would like to thank Dan Trueman, Tae Hong Park and Manjul Bharagava for their performance in the *Gigapop Ritual* at Princeton. We would also like to thank Ari Lazier, Tom Briggs and Mary Roberts for their technical support at Princeton before and during the performance. This concert would not have been possible without Adam Tindale, Richard McKenzie and the rest of the NIME 03 organisers at McGill University. We would also like to thank Chris Chafe and Gary Scavone for helping us get our feet wet. Also thanks go to Fernando Lopez-Lezcano for creating Planet CCRMA Redhat package manager, and to Szymon Rusinkiewicz for raw video capture code in Linux.

### REFERENCES

- Babbitt, M. 1958. Who cares if you listen? *High Fidelity*, February.
- Bagchee, S. 1998. *NAD: Understanding Raga Music*. Mumbai, India: Ceshwar Business Publications Inc.
- Barbosa, A. 2003. Displaced Soundscapes: a survey of networked systems for music and sonic art creation. *Leonardo Music Journal* 13: 53–9.
- Barger, R., Church, S., Fukada, A., Grunke, J., Keisler, D., Moses, B., Novak, B., Pennycook, B., Settel, Z., Strawn, J., Wisner, P., and Woszczyk, W. 1998. *AES White Paper: Networking Audio and Music Using Internet2 and Next Generation Internet Capabilities*. New York: Audio Engineering Society.
- Chafe, C., and Leistikow, R. 2001. Levels of temporal resolution in sonification of network performance. In *Proc. of the Int. Conf. On Auditory Display*. Helsinki, Finland.
- Chafe, C., Wilson, S., Leistikow, R., Chisholm, D., and Scavone, G. 2000. A simplified approach to high quality music and sound over IP. In *Proc. of the Int. Conf. On Digital Audio Effects*. Verona, Italy.
- Chafe, C., Wilson, S., and Walling, D. 2002. Physical model synthesis with application to Internet acoustics. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing*. IEEE.
- Föllmer, G. 2002. *Musikmachen im Netz Elektronische, ästhetische und soziale Strukturen einer partizipativen Musik*. Ph.D. thesis, Martin-Luther-Universität Halle-Wittenberg.
- Freeman, J., Ramakrishnan, C., Varnik, K., Neuhau, M., Burk, P., and Birchfield, D. 2004. Adaptive high-level classification of vocal gestures within a networked sound instrument. *Proc. of the Int. Computer Music Conf.*, pp. 668–71. Miami, Florida.
- Gurevich, M., Chafe, C., Leslie, G., and Taylor, S. 2004. Simulation of Networked Ensemble Performance with Varying Time Delays: characterization of ensemble accuracy. In *Proc. of the Int. Computer Music Conf.* Miami, Florida.
- Helmuth, M. 2000. Sound exchange and performance on Internet2. In *Proc. of the Int. Computer Music Conf.* San Francisco.
- Jorda, S. 1999. Faust Music On Line (FMOL): an approach to real-time collective composition on the Internet. *Leonardo Music Journal* 9: 5–12.
- Jorda, S., and Barbosa, A. 2001. Computer Supported Cooperative Music: overview of research work and projects at the Audiovisual Institute – UPF. *Proc. of MOSART Workshop on Current Research Directions in Computer Music*, pp. 92–7.
- Kapur, A., Davidson, P., Cook, P. R., Driessen, P., and Schloss, W. A. 2004. Digitizing North Indian performance. In *Proc. of the Int. Computer Music Conf.* Miami, Florida.
- Kapur, A., Essl, G., Davidson, P., and Cook, P. R. 2003. The Electronic Tabla Controller. *Journal of New Music Research* 32(4): 351–60.
- Kothari, K. S. 1968. *Indian Folk Musical Instruments*, pp. 43–4. New Delhi, India: Sangeet Natak Akademi.
- Goto, M., Neyama, R., and Muraoka, Y. 1997. RMCP: Remote Music Control Protocol – design and

- applications. *Proc. of the 1997 Int. Computer Music Conf.*, pp. 446–9. September.
- 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).
- Puckette, M., Sorensen, V., and Steiger, R. 1997. *Lemma 1*, performed live at Milos Jazz Club at the International Computer Music Conference, Thessaloniki, Greece.
- Rowe, R., and Rolnick, N. 2004. The Technophobe and the Madman: an Internet2 distributed musical. In *Proc. of the Int. Computer Music Conf.* Miami, Florida, November.
- Shanhabi, C., Zimmermann, R., Fu, K., and Yao, S. D. 2002. Yima: a second generation continuous media server. Published in the *IEEE Computer Magazine*, June, pp. 56–64.
- Sharma, S. 1997. *Comparative Study of Evolution of Music in India & the West*. New Delhi, India: Pratibha Prakashan.
- Stelkens, J. 2003. peerSynth: a P2P multi-user software synthesizer with new techniques for integrating latency in real-time collaboration. In *Proc. of the Int. Computer Music Conf.* Singapore.
- Tanaka, A. 2000. Interfacing Material Space and Immaterial Space: network music projects, 2000. In *The Journal of the Institute of Artificial Intelligence of Chukyo University*, Winter. Toyota, Japan.
- Weinberg, G. 2002. *Interconnected Musical Networks: Bringing Expression and Thoughtfulness to Collaborative Music Making*. Ph.D. dissertation. MIT Media Lab, Cambridge, MA.
- Xu, A., Woszczyk, W., Settel, Z., Pennycook, B., Rowe, R., Galanter, P., Bary, J., Martin, G., Corey, J., and Cooperstock, J. 2000. Real time streaming of multi-channel audio data through the Internet. In *Journal of the Audio Engineering Society*, July/August.