GroupLoop: A Collaborative, Network-Enabled Audio Feedback Instrument

David B. Ramsay
Responsive Environments
MIT Media Laboratory
20 Ames St.
Cambridge, MA 01239
dramsay@media.mit.edu

Joseph A. Paradiso
Responsive Environments
MIT Media Laboratory
20 Ames St.
Cambridge, MA 01239
joep@media.mit.edu

ABSTRACT

GroupLoop is a browser-based, collaborative audio feedback control system for musical performance. *GroupLoop* users send their microphone stream to other participants while simultaneously controlling the mix of other users' streams played through their speakers. Collaborations among users can yield complex feedback loops where feedback paths overlap and interact. Users are able to shape the feedback sounds in real-time by adjusting delay, EQ, and gain, as well as manipulating the acoustics of their portion of the audio feedback path. This paper outlines the basic principles underlying *GroupLoop*, describes its design and feature-set, and discusses observations of *GroupLoop* in performances. It concludes with a look at future research and refinement.

Author Keywords

Audio feedback, internet browser, WebRTC, laptop music, audio streaming, audio networks, collaborative instrument

1. INTRODUCTION

Audio feedback occurs when the output of a speaker is reamplified by the microphone that is driving it. In other words, feedback occurs whenever there is a loop by which an audio signal will repeatedly be combined with a delayed copy of itself.

Most people associate audio feedback with an undesirable, piercing tone. However, feedback has an illustrious history as a tool for musical expression. Pioneered in the 1960's by the likes of Pete Townshend, Jimi Hendrix, and Pink Floyd, feedback quickly became an integral part in the rock guitar lexicon. Outside popular music, composers like Steve Reich (Pendulum Music, 1966), laid the framework for innovative feedback architecture [1]. Recent works by David Lee Myers (i.e. "Feedback Chains") as well as Christian Carriere and Toshimaru Nakamura (icons of the 'no-input' mixer genre) drive the field forward [2].

Several musicians have designed instruments specifically for the creation of feedback music. Andrew McPherson is known for feedback designs based on pre-existing acoustic instruments [3]. Composers like Jeff Morris and Nicolas Collins have created platforms that inject a range of effects, manage the mixing of multiple acoustic paths, and, in some instances (i.e. Collins' "Pea Soup"), actively monitor and regulate feedback behavior [4][5].

Experimental network music also gained momentum starting in the 1950's with eary cybernetic artwork. As telecommunication infrastructure has improved, networked music performance (NMP) has become one of its dominant current incarnations. Laptop orchestras and high-speed National Research and Education Networks (NRENs) are now common platforms for experimentation in collaborative and emergent network music generation. The web browser has also seen growth in the collaborative music space, with applications such as Plink, Jam with Chrome, and Soundtrap.

These platforms and applications are not typically designed to create feedback music, however. The intersection of networked audio and feedback music has been explored in a handful of isolated work. Ritsch's "I am playing in one netroom" (2010) and Davis (et al.)'s "The Loop" (2012) are two notable examples of internet based feedback designs.

GroupLoop fits in the tradition of these feedback systems, with several notable innovations. *GroupLoop* is a browser-based feedback performance system that connects multiple acoustic spaces from anywhere in the world. It was designed to easily enable collaboration between multiple users across skill level and geography. *GroupLoop*'s decentralized architecture allows rapid growth in system complexity while maintaining its real-time configurability.

GroupLoop is available at `feedback.davidbramsay.com`.

2. PLAYING AND CONTROL

Grouploop's software connects to the local computer's default soundcard and forms streaming peer-to-peer audio connections with other *GroupLoop* users over the internet. Participants control their output by mixing of the incoming microphone streams. Any audio transmitted through the speakers is recaptured by a user's microphone and sent back out over the network to collaborators. The number of simultaneous users– and thus the number of available speakers and microphones– is only limited by the processing power of the host computers and network speed.

The networked design of *GroupLoop* affords players control over the creation, alteration, and interaction of multiple simultaneous feedback paths through multiple acoustic spaces. The real-time, distributed control of this evolving process– allowing the user to selectively turn on or off an incoming stream and adjust the volume accordingly– is the fundamental user interaction within the instrument.

Users can vary the sound of the instrument both before and during performance. Real-time controls on the outgoing audio include a highly-responsive graphic EQ and virtual knobs for delay, microphone gain, and EQ strength. Users can manipulate the feedback sound by increasing the gain (up to 20 dB) or adding up to two seconds of delay to

NIME'15, May 31-June 3, 2015, Louisiana State Univ., Baton Rouge, LA.

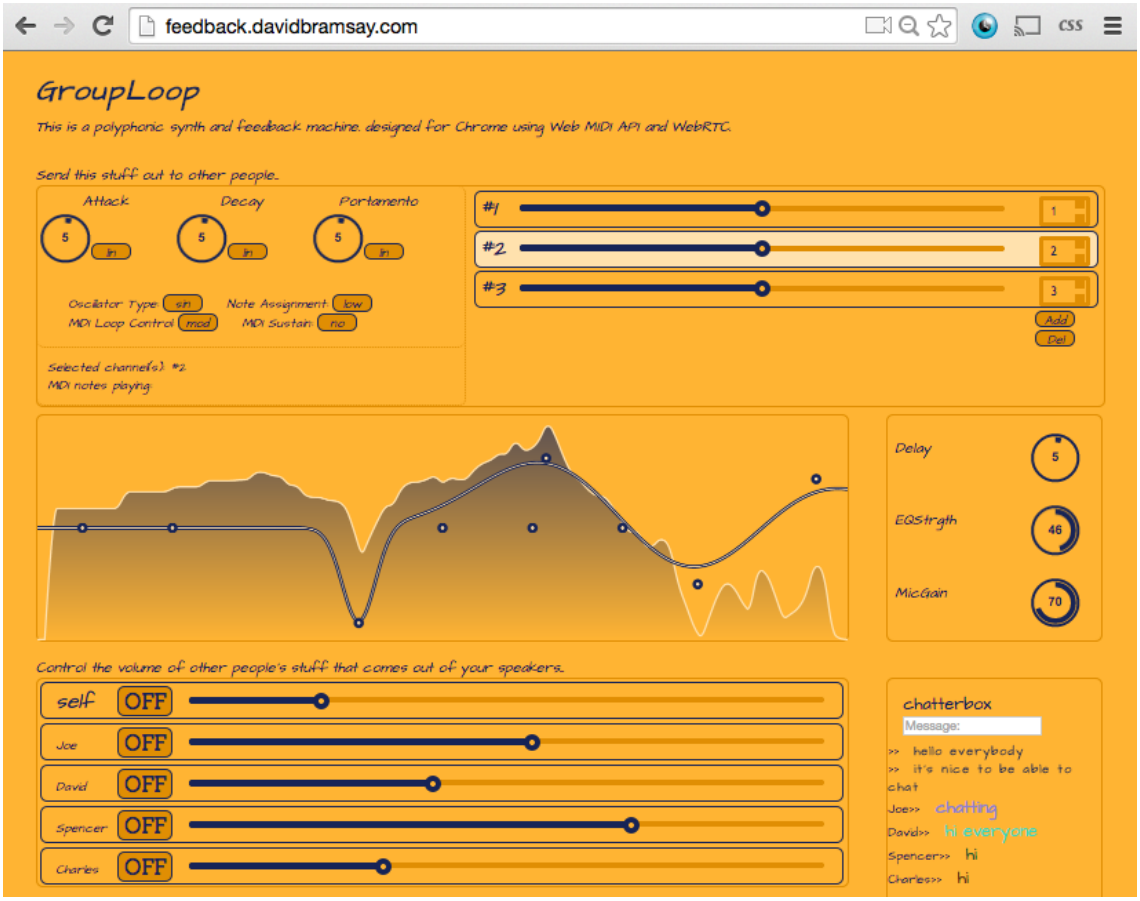

Figure 1: GroupLoop UI

their microphone stream. The EQ features nine adjustable-Q, resonant bandpass filters. The maximum gain of these filters is plus/minus 40 dB of gain, with the option to add an additional 80 dB.

Users can further manipulate their sound by altering microphone/speaker selection, external signal processing (i.e. limiters), system geometry, and environmental acoustics. Advanced users can go beyond the use of built-in computer microphone and speakers to set up complex and time-varying acoustic systems like those described in Reich and Morris. Moving, covering, touching, or otherwise modifying the acoustic elements (e.g. speakers, microphones) during a performance are simple but effective ways to interact with the instrument. In [6], the inventors of Laptap provide a thorough overview of acoustic interventions to 'play' feedback loops for a fixed laptop geometry.

Finally, *GroupLoop* includes a built-in MIDI synthesizer to inject source material into the feedback process. MIDI-enabled participants may also use their MIDI modulation wheel to control the volume of incoming audio streams, or automatically mute/play each stream in synchrony with synthesizer notes.

2.1 Source Material

There are four types of input signals that drive the feedback process: environmental sounds, recorded audio, embedded synthesis, and self-noise.

The most basic input is from players shouting, singing, clapping, or otherwise introducing self-made (i.e. environmental) sounds into their feedback paths using the microphone. Audio recordings can also be used by playing them on the *GroupLoop* computer through the system speakers. Recordings can be routed directly into the outgoing microphone streams with high fidelity using services like SoundFlower [7].

The embedded MIDI synthesizer, which takes advantage of the Web MIDI API (an experimental feature only available in Chrome) [8], is another option for input. The user can add/remove oscillators, modify their harmonic relationships, and change the volume, attack, decay, and portamento for each. Sine, square, triangle, and saw-tooth waves are available, and oscillators can be mapped to MIDI input notes in a variety of novel ways. For instance, each oscillator can be assigned– based on chord voicings– to only the lowest sounding note, the second lowest, the highest, etc. This output is directly mixed into the user's microphone stream, with an option to sustain the audible notes indefinitely.

With enough gain, *GroupLoop* will begin feeding back without an apparent input. In this case, the amplification of the noise floor of the system is enough to result in a growing feedback state with each loop iteration. This method of feedback is easily controlled with EQ.

GroupLoop's EQ gain scale was chosen to make it easy to achieve self-noise feedback across all frequencies on a 2013 MacBook Pro laptop. There is additional equalization upstream of the UI to bias the system towards a more even self-feedback response over frequency by boosting the bass and attenuating the high end.

2.2 Visual Design

The features described above are laid out in three visual sub-sections, as shown in Figure 1. The top two panels affect audio leaving the computer, while the last section controls incoming connections.

The top panel is hidden unless a MIDI device is plugged in. It reveals the *GroupLoop* synthesizer, divided into an oscillator bank on the right and control parameters on the left. The microphone/synthesizer output stream is visualized in the second section, with a real-time spectrum analyzer, a configurable EQ, and other associated controls.

The last block shows all available incoming streams with on/off and volume controls. Whenever a new user logs into the service, they automatically appear as an input in this section of the UI. A chat feature is included to help coordinate performances and encourage collaboration.

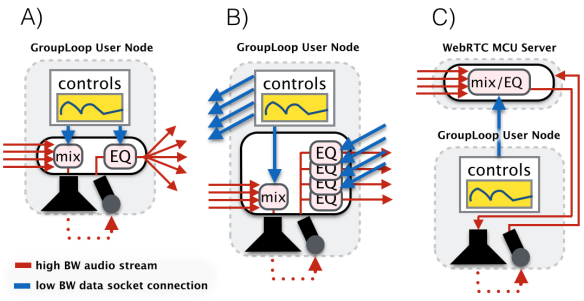

Figure 2: Potential GroupLoop networking topologies. (A) represents the current implementation, (B) shows a computationally expensive alternative, and (C) introduces a future, centralized server.

3. AUDIO NETWORKING

Networking is at the core of the *GroupLoop* platform. Because of this, *GroupLoop* was developed for the Google Chrome browser using Node and Javascript. It takes advantage of HTML5 and other advanced browser functionality, including Web MIDI (which is only implemented in Chrome), WebRTC, and Web Audio APIs. Webrtc.io, WebMIDIAPIShim, jQuery, jQuery Knob, and jQuery RangeSlider open source libraries are also used.

3.1 WebRTC and its Limitations

While there are recent examples of collaborative performance over private, high-speed NRENs, public internet latency still renders traditional real-time ensemble performance impractical [9][10]. NMP research typically focuses on strategies to either cope with this latency or incorporate it into a network-based effect (like reverb) [11]. In the context of feedback performance, however, 60-300ms of delay is tolerable and often desirable.

WebRTC is a relatively new W3C standard [12], with working desktop implementations in Google Chrome, Mozilla Firefox, and Opera. This API offers simple, browser-based, peer-to-peer media streaming. Connections based on the Opus codec and a customizable Session Description Protocol offer full-band (48k sampling rate) and sub-100ms latency audio performance, subject to network speed constraints. If the connection slows, Opus is capable of transitioning to a lower sampling rate in real-time, preserving the stream but temporarily reducing bandwidth [13].

Early adoption of any technology has certain drawbacks. WebRTC implementations in Firefox and Chrome differ substantially, constantly evolve, and lack refined documentation. One notable feature– the ability to process incoming network peer audio connections through the Web Audio API before playing them out of the speakers– is currently unsupported in Chrome [14].

3.2 Network Topologies and Control

Several of the *GroupLoop* design choices follow naturally from the Chrome WebRTC implementation. Control for equalization and delay is placed on the outgoing connection, as shown in (A) of Figure 2. To instead map the UI to affect the audible, incoming connections would require duplicating the audio processing of (A) on every upstream device, with control parameters sent remotely over a data socket link (B). The current implementation already approaches the limit of in-browser computation, making (B) impractical.

Grouploop creates a true peer-to-peer, fully connected mesh network, which provides the most configurable, lowest-latency topology. However, this also means that for *N* users, each device has *N-1* bi-directional full-band connections. Tests have shown that up to ten users can operate *GroupLoop* in the fully connected configuration without performance degradation. For larger networks, a central routing server, called an MCU (Multipoint Control Unit), could drastically reduce the number of streaming connections at the cost of increased latency [15]. Solutions currently exist that offer basic WebRTC MCU routing services. With additional effort to port Web Audio functionality to the MCU environment, audio processing might also be offloaded to the MCU (Figure 2-C).

4. PERFORMANCE

GroupLoop was debuted in December 2014 with a live performance by three trained musicians at the MIT Media Lab. Four laptops running *GroupLoop* were configured with external soundcards, speakers, and microphones, and placed around a large shared performance space. An additional machine was set up in a remote location.

This model of expert collaboration in a shared space was chosen to demonstrate advanced performance technique. To shape an audio stream without knowing where in the room or how loud it will be played demands careful listening and group coordination. The additional remote computer represents how a novice might collaborate. The extra node was exploited by the experts as an effects processor might be– selectively pulled into their feedback paths to change the sound quality. It is evocative of a many-to-few performance model, in which several novice users remotely collaborate with a handful of centralized, expert musicians.

Performance with *GroupLoop* is unrepeatable and difficult to control, and composition for the platform is a form of process music. In group settings, *GroupLoop* demands a high level of skill and understanding to play, with narrow margins for error. It remains accessible, however, for novice users interested in simpler contributions.

5. FUTURE WORK

In its current design, *GroupLoop* lends itself to two system topologies: (1) experienced players using the platform in a shared space together collaboratively, and/or (2) novices around the world setting up simple remote systems that the experts may exploit.

Iterations of *GroupLoop* that granted MIDI-enabled users additional control over their output stream were sonically limited and unintuitive to use. Fundamental changes in the balance of control or network topology are more likely to generate compelling new instrument designs. One example is a master/slave topology- in which one central user has

Figure 3: GroupLoop in use during a three musician live performance. Notice the external speaker and microphone feedback elements, as well as the MIDI controller.

complete control over a star or mesh network. Performance with this design could include the ability to send different streams out through each feedback loop, splitting individual notes in a synthesizer chord (or other inputs) between downstream users.

These concepts generally share the assumption of a centralized performance space with remote contributors. One impediment to this design is the lack of a monitoring solution for remote participants. Monitoring requires an additional audio stream from the performance space to each user and significant additional setup. It represents a tradeoff between flexibility and ease of use, and for *GroupLoop* a low barrier to entry was paramount. However, for performances featuring a small set of experienced remote users, monitoring would be important.

Furthermore, this system is suggestive of simultaneous, linked, remote performances. To enable this type of collaboration, a solution that tightly couples the user controls to the local aural experience is required. Updates to Chrome's WebRTC implementation may make this an easy modification in the near future– otherwise, a change in platform or a sophisticated control paradigm will be required.

There are other areas for improvement in the core platform. A more sophisticated visualization of the network would be useful to manage complex loops in real-time. An MCU server would enable support for more simultaneous connections with a small increase in latency. Additional signal processing for modifying and controlling the feedback path could also be beneficial, although the current design is close to the limit of real-time browser computation. Offloading part of the processing to a powerful MCU could address this limitation, though it would require a significant investment in custom software architecture.

6. CONCLUSIONS

GroupLoop was created to enable the collaboration of users across all skill levels and geographies. Performance with the platform suggests success, as it is simple enough to start by opening a browser on a laptop, but extensible enough to invite complex acoustic design and virtuosic mastery. As an instrument, it presents the player with challenges that are both technical and artistic. It is capable of diverse and unexpected sounds, immeasurable reconfigurability, and in some cases, unrepeatable complexity. *GroupLoop* encourages a high level of collaboration, and leverages new technologies to become one of the first full-band, real-time collaborative music platforms available on the internet.

GroupLoop creates new topologies for collaboration in performance, and invites thoughtful reflection on future topologies for real-time music collaboration over any distance. Technology is quickly driving toward ubiquitous, internet-based collaboration. *GroupLoop* represents a step towards that connected, musical future.

7. ACKNOWLEDGMENTS

The authors would like to acknowledge Tod Machover and the participants of the MIT Media Lab 'Future of Music' class– particularly Spencer Russell for his continued guidance. Special thanks also goes to the *GroupLoop* performers: Charles Holbrow and the aforementioned Mr. Russell.

8. REFERENCES

[1] Steve Reich. *Writings About Music*. New York University Press, 1974.

[2] Todd Jenkins. *Free Jazz and Free Improvisation: An Encyclopedia, Volume 2*. Greenwood Press, 2004.

[3] Andrew McPherson. The magnetic resonator piano: Electronic augmentation of an acoustic grand piano. *Journal of New Music Research*, 39(3):182–202, 2010.

[4] Jeffrey M Morris. Feedback instruments: Generating musical sounds, gestures, and textures in real time with complex feedback systems. In *Int. Computer Music Conference, Copenhagen, Denmark*, pages 469–476, 2007.

[5] Nicolas Collins. The history of *Pea Soup*. *www.nicolascollins.com*, 2011.

[6] Dae Ryong Hong and Woon Seung Yeo. *Laptap: Laptop Computer as a Musical Instrument using Audio Feedback; Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 233–236. 2013.

[7] Soundflower, http://rogueamoeba.com/freebies, 2015.

[8] J. Kalliokoski and C. Wilson, editors. *Web MIDI API*. W3C Working Draft (work in progress), http://www.w3.org/TR/webmidi, 2013.

[9] Carlo Drioli, Claudio Allocchio, and Nicola Buso. Networked performances and natural interaction via LOLA: Low latency high quality A/V streaming system. *Information Technologies for Performing Arts, Media Access, and Entertainment*, 7990:240–250, 2013.

[10] Nathan Schuett. The effects of latency on ensemble performance. Undergraduate honor's thesis, CCRMA, Stanford University, 2002.

[11] J.P. Cáceres and A. Renaud. Playing the network: the use of time delays as musical devices. In *Proceedings of International Computer Music Conference*, pages 244–250, 2008.

[12] A. Bergkvist, D. Burnett, C. Jennings, and A. Narayanan, editors. *WebRTC 1.0: Real-time Communication Between Browsers*. W3C Working Draft (work in progress), http://www.w3.org/TR/webrtc, 2013.

[13] J. M. Valin, editor. *Definition of the Opus Audio Codec*. Internet Engineering Task Force (IETF), http://tools.ietf.org/html/rfc6716, 2012.

[14] Issue 241543: createMediaStreamSource does not work for remote peer's stream, 2015. Chromium Bug Tracker, code.google.com/p/chromium/issues/detail?id=241543.

[15] Justin Uberti and Sam Dutton. Real-time communication with WebRTC. In *Google I/O*, 2013.