

Repertoire Remix: Integrating Musical Preferences from Remote Audience in A Live Networked Improvisation

Akito van Troyer¹

¹ MIT Media Lab – Massachusetts Institute of Technology
Cambridge, U.S.A.

akito@media.mit.edu

Abstract. *Repertoire Remix enables remote audience members to dynamically suggest their musical preferences for live web-streaming musical improvisation sessions. The semantic web interface encourages remote participants to collaboratively use “stirring” mouse gestures to influence the size of graphical bubbles that contain composers’ names. The accumulated weight is then interpreted by musicians to improvise. This paper documents the first pilot run of the Repertoire Remix system, explores challenges in designing a real-time shared music-style arranging system for networked live improvisation, and interprets the resulting performance by assessing the participants’ mouse gestures collected during the pilot run.*

1. Repertoire Remix in the Context of *Festival City*

Festival City is a symphonic work commissioned by the Edinburgh International Festival (EIF) that premiered in August 2013 [Edinburgh International Festival 2013]. The main objective of this work was to have a composer, Tod Machover, and people in the Scottish capital, Edinburgh, to collaborate in composing a final piece of music in one of the concert series taking place in EIF. In realizing this objective, our first step was to develop a number of web-based music composition applications that enabled people from Edinburgh to remotely and asynchronously collaborate in music-making on the web [van Troyer 2013]. These composition applications brought together an unprecedented number of people from diverse backgrounds to contribute in pursuing our main objective and also established a unique model for creating intricate collaborations between experts and non-experts in composing music.

While these web-based composition environments achieved our main objective, we wanted to further experiment with the *Festival City* project to engage our audience more intimately by providing them opportunities to remotely interact with the composer and musicians in a live context. To accomplish this goal, we created Repertoire Remix, a web browser-based application, that can creatively engage remote audience members to become part of the real-time improvisational performance process [Guardian 2013]. Repertoire Remix does not require website visitors to have specialized musical training. It uses a visual interface consisting of ten morphing graphical bubbles that represent famous composers and their music styles (See Figure 1.), and website visitors simply “stir” their mouse in the interface to suggest their musical preference to performers. The composer bubbles then change size according to the mouse gestures, which then further influence how the performers improvise. While remote participants influence the composer bubbles, the director, a person who oversees and moderates the entire performance session, controls

a second interface with sliders and buttons that determines how the composer bubbles overlap, bounce against each other, or fuse together. The aim of having a director is to give the performers more ideas about how to combine and transform their improvisation.

Repertoire Remix is also a system that collects various real-time data including MIDI data events, video, audio, and remote participants' mouse gestures. We collected these data to facilitate our original objective to utilize such data in the final piece of music that can be performed at EIF. For our first pilot run, the solo piano improvisation session, performed by Tae Kim and directed by Tod Machover, was recorded and later incorporated into the final piece of music that then got performed at EIF [MIT Media Lab 2013].

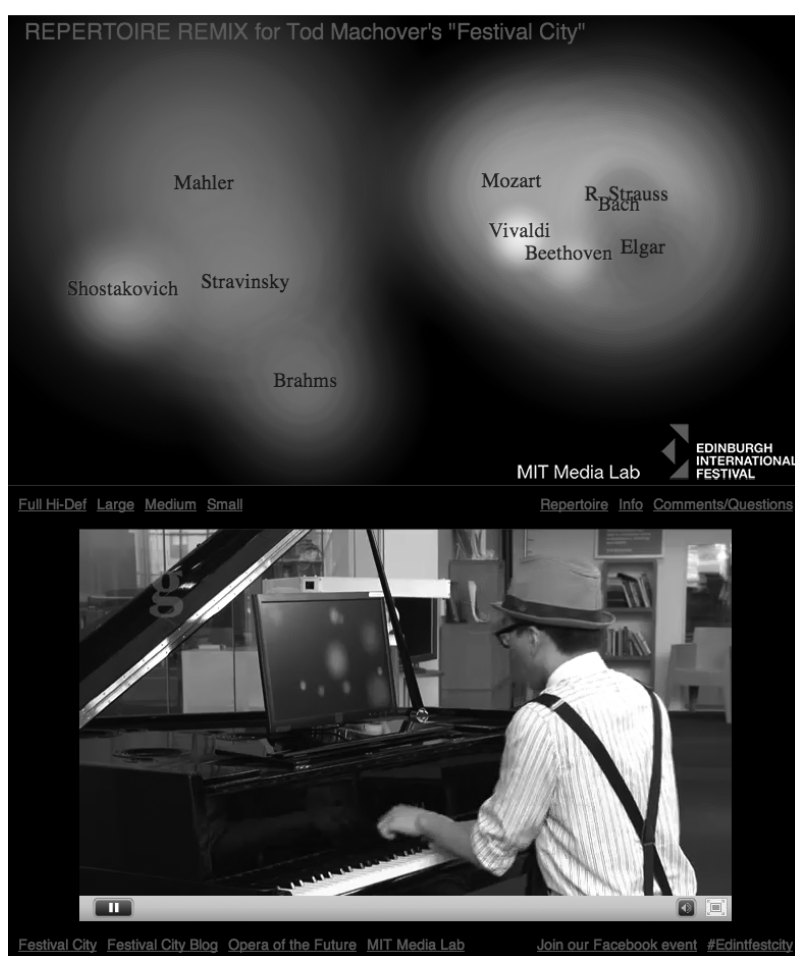


Figure 1. A screen shot of the Repertoire Remix web interface during the pilot run of the improvisation session

2. Audience Integration in Network Music Environments

Many network music projects incorporate audience members as integral parts of their performances. Such performances typically encourage them to be expressive and collaboratively influence the performance in real time [Miletto et al. 2011]. The earliest examples of such performances are Brain Opera and Cathedral, where online collaborative interactions remotely influenced live concert performances [Machover 1996][Duckworth 1999]. These projects provided opportunities to rethink the nature of collective interaction in public spaces when remote participants were also involved in this process. Similarly, more

recent projects such as Quintet.net [Hajdu 2003] and Graph Theory [Freeman 2007] provide feedback and music-making systems for remote participants to contribute to performances in real time. These projects demonstrate the capability of network music performance systems to provide unique and enriching interactive musical experiences to remote audience members.

Recent developments in computer network systems have led to new approaches for composition and improvisation for musical novices. These network music systems, termed “shared sonic environments” [Barbosa 2003], enable remote participants to collaboratively create shared soundscapes. Such environments can often blur the boundaries between audience and performers when, for example, remote participants become the central agents in producing sounds that are then mixed and processed in real time over a broadcast/network system to create a musical performance [Freeman et al. 2005]. In other cases, remote participants’ collaborative activities on a simple visual-oriented web interface result in creating unique music improvisations and compositions [Bryan-Kinns et al. 2007][Burk 2000]. These projects partly rely on visual information to show other participants’ activities in the shared environment. In addition, this visual information often works as a feedback system and helps participants by suggesting other people’s presence in the environment. Such use of visual information also facilitates competition and collaboration among the participants, making musical creation fun and social.

In designing Repertoire Remix, we also explored how existing real-time music notation and audience participation-based performance systems facilitate interaction between audience and performers to the point that they can both contribute to creating an improvised performance. For this interaction to happen, we imposed a condition that such performance systems do not require participants to learn about or be skilled at operating the interface, as most of them are likely to be first-time users and have no time for a learning curve [Blaine and Fels 2003]. Similar approaches can be seen in many of the prior real-time music notation systems. Two examples are real-time music notation systems in which the audience was able to shape the ongoing music at the performance site [Baird 2005][Freeman 2008] and online music composition systems that enable website visitors to compose music through an open-form music scoring method that was then used for the actual performance [Freeman 2010]. The challenge of these projects was to enhance the participatory experience of musical novices through simple music composition interfaces. Furthermore, these projects gave new perspectives to the prospect of collaboration among amateur and professional musicians.

3. Design and Implementation

The core interaction flow of Repertoire Remix is illustrated in Figure 2. The figure can be divided into three parts: remote locations, servers, and headquarters. Remote locations represent the website visitors. They use a web browser-based interface that is designed to encourage them to stir their mouse to influence the bubbles within the shared music-style arranging environment (SMSAE) (See the top portion of Figure 1). At the same time, they also see and hear the musicians’ performance and director’s musical instructions through web streaming video at the bottom part of the interface (See the bottom portion of Figure 1). These activities for website visitors are made possible by the servers that relay interaction data and web streaming video to the remote client. Servers also make the remote

collaboration happen between remote locations and headquarters. Headquarters is where the actual performance is taking place, and the Repertoire Remix system assumes that performers, director, and comment reviewers exist in the same physical space. Performers improvise music during the session using a simplified version of the SMSAE interface while a director, who has his/her own unique web interface, oversees and moderates the interaction between performer and remote audience. Although optional, comment reviewers exist to filter out unwanted comments from remote audience and deliver appropriate comments to the performers and the director for live responses via the web stream. Comment reviewers also have their own unique interface to regulate comments. Repertoire Remix consists of four different interfaces but the system is able to realize bidirectional communication among them through the SMSAE, video streaming, and web comments.

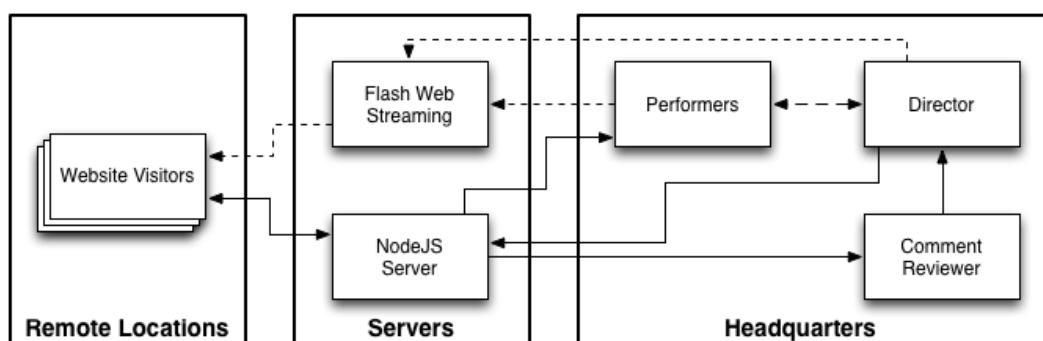


Figure 2. The core interaction flow in Repertoire Remix. Bold lines represent interaction data flow while dotted lines shows the performance audio/video capturing and streaming. The dashed line represent a verbal dialog that can take place between director and performers during the session.

3.1. Shared Music-Style Arranging Environment

One of our goals in designing the Repertoire Remix system was to provide a shared visual environment between participants, who may have little or no music training, and professional musicians, so that they can communicate their musical ideas and intentions through the same abstract visual representations. We chose to illustrate such abstract visual representation using graphical bubbles that contain composer names. These bubbles can morph their texture, size, and positions during the course of the improvisation session. We believe that this approach can empower novices to easily express their musical ideas only using their mouse gestures. The system works so that whenever the participants stir their mouse, the closest composer bubbles grow in size. This approach of using abstract visual representation with simple mouse gestures also helped promote our ideas to keep the musicians' interpretation of music style suggestions open. They do so by viewing a simplified non-interactive version of the SMSAE interface responding to aggregated user data shown in Figure 1 with a clock to keep track of time.

To technically realize SMSAE, we used NodeJS as the underlining server architecture. NodeJS is a server-side platform for building fast and scalable web applications [NodeJS 2011]. It is also simple and very well-suited for creating servers that can hold shared data across all clients. The type of data shared across all browsers were related to the properties of the composer bubbles and the logical propositions of who

influenced which of those bubbles. The shared data are then communicated among the clients using NowJS which makes building real-time web applications using WebSockets effortless within the NodeJS environment [NowJS 2011][WebSocket 2011]. The data can also be stored on the server using Mongoose, a NodeJS module that interfaces with MongoDB [Mongoose 2011] [MongoDB 2011], for an optional post analysis of a performance. Client browsers that support HTML5 standards are then used to render the graphical bubbles using the canvas element [HTML5 Canvas 2009].

3.2. Slider and Button Interface for Director

In designing the Repertoire Remix performance system, we felt the need for a person who can moderate and run the improvisation session since our first pilot run was not only going to involve several tryout performances within an hour, but also a discussion between remote participants and the performers. For this reason, we created a role of director in the system who could facilitate the dialog and also oversee the performance sessions. The director has his/her own unique interface with sliders and buttons for influencing some parameters of the SMSAE (See Figure 3). These parameters are related to the properties of the composer bubbles such as their position, blurring effect, size growth rate, and speed. The sliders for growth rate and the buttons at the bottom of the interface are ways for the director to take control over composer bubbles that are getting “out of control” so that the improvisation session can proceed with appropriate variations and some cohesiveness. Other sliders, such as bubble position and blurring effect, are used to give performers variations in how to combine different styles of music. As the nature of such an abstract/non-traditional interface can be confusing for the remote participants in understanding the interaction, for our first pilot run, we explained the interface functionality to remote participants prior to and at the beginning of the session so that they will not be confused by the unexpected behavior of the composer bubbles during the performance.

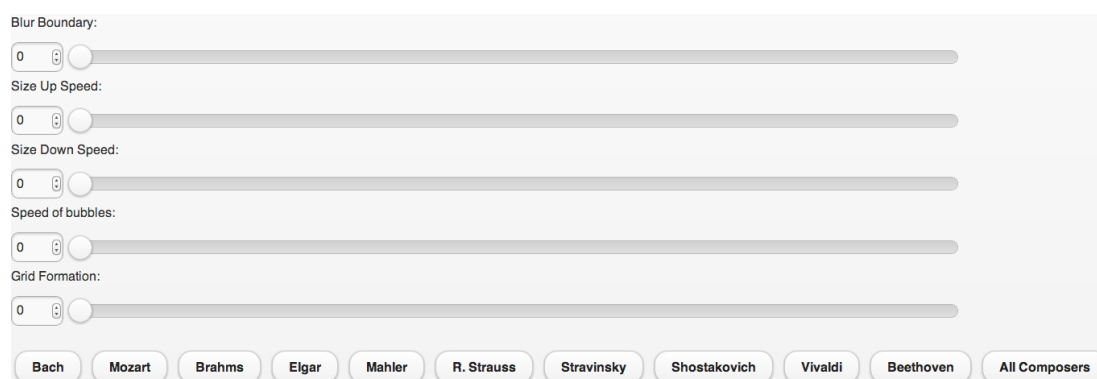


Figure 3. A slider and button interface for a director

3.3. Web Streaming

The set up for the Adobe Flash-based web streaming system is entirely independent from the rest of the interactive Repertoire Remix system. This enables anyone using the Repertoire Remix system to flexibly arrange how to capture the audio/visual contents and how to send the streaming view to the remote participants. In our first pilot run for instance, we operated two cameras to capture the improvisation session from different view angles, which then were mixed live using a video mixer. In addition, we also used an audio mixer

to capture several different sources of sounds from microphones (mostly for conversation) and MIDI piano sounds captured from Yamaha AvantGrand N3 piano. The MIDI events from the piano were ideal in our situation to synthesize electronic piano sounds and keep the quality of the audio at the maximal level for the web streaming. The web streaming display at the bottom of the main interface served as the primary way for the remote participants to be able to view the performance in real time and see the effects of their participation in the performance.

3.4. Comment/Feedback System

embedded in the main interface, the Repertoire Remix system also features a way for the remote participants to send live comments. When a participant sends a comment, the message is displayed in the interface illustrated in Figure 4. Although this interface is not necessary for all performance scenarios, we intend this live commenting system to be interpreted by someone other than the director or the performers for filtering unwanted comments and delivering the right ones that promote conversation about the performance and how to improve it. The reason why we implemented such system was to enable remote participants to exchange their musical ideas that would otherwise be hard to communicate with the director and the performers through visual information alone. The comment system can enable high-level and sophisticated ways of sharing musical idea across remote locations.



Name:	jdub
Location:	boston
Comment:	I like the stravinsky to bach transition
Email:	

Name:	Nicola
Location:	Edinburgh
Comment:	How do Tae interpret the circles on screen and translate it into what he plays?
Email:	

Name:	Jonathan
Location:	Edinburgh
Comment:	Who will perform the final piece at the Edinburgh International Festival?
Email:	

Name:	Gabriela
Location:	Brasil
Comment:	Amazing! I am just missing Scarlatti... any chance? :-)
Email:	

Figure 4. An example interface for comment interpreters

4. Interpretation of Shared Visual Score by Performers

Our basic protocol for how the performers interpret the shared visual representation is based on the accumulated weight of the bubble sizes; the performers establish their improvisation style in favor of the bubbles that are bigger in size than others and build up the

aggregated musical styles all together. In our first pilot run for example, the pianist often took three to four bubbles that are larger than the rest and mixed the style sequentially in improvisational fashion. We also imposed that interpreting the position and speed of the composer bubbles will affect how chaotic or random the mix of styles becomes, while the blurring effect of each bubble will be used to influence how performers blend different music styles into one coherent performance style.

5. Pilot Run

The first pilot run of Repertoire Remix took place on July 9, 2013 between 7pm and 8pm GMT hosted on the Guardian website. In our first pilot run, over one thousand people visited our website and 71 unique visitors interacted with the shared interface during the improvisation session. Geographically speaking, participants literally were from all over the world including Brazil, Thailand, Ireland, U.K., and U.S.A. Participants were informed that the improvisation session was a one-time-only special event and musical data collected in this event will be used by Machover as he crafts the “repertoire fragments” section of *Festival City*. Composer names for the graphical bubbles were chosen based on the most frequently performed piece at EIF since its inception in 1947.

During this hour-long session, Machover and Kim created four different versions of Repertoire Remix improvisation, of fifteen, eight, five, and three minutes, refining with each iteration. Between each performance, they discussed the previous version and how to make it tighter and more effective. While this was happening, participants also sent their comments and questions, some of which got highlighted in the conversation at the headquarters. The entire improvisation was captured in video, audio, MIDI events, and activities of the participants including mouse gestures, comments, and their locations as well as names.

6. Activity Assessment of the Pilot Run

This section explores the visualization and analysis of the “mouse gesture” data logged by the system during the improvisation session. The data being analyzed in this section were collected every time remote participants influenced one of the composer bubbles in the shared visual environment. Through the collected data, we looked for any identifiable and meaningful patterns that could provide quantitative information about the improvisation session and the behaviors of remote participants.

Figure 5 shows the aggregated mouse gestural influences of all remote participants on the graphical bubbles throughout the entire improvisation session. We counted all participants’ mouse gestural influence on the bubbles every 12 seconds and visualized them to see when participants actually interacted with the bubble interface. The graph also indicates with gray regions when the test session and each improvisation session happened. The trend in the graph tells us that the increase in mouse activities happens consistently with times each improvisation sessions was taking place. In contrast to our worries about network delay and providing web-streaming content on time, the graph indicates that remote participants were able to participate in influencing each improvisation session in real time with little lag between video and interaction. On the other hand, we also observe relatively intense activity in the intermissions between sessions 2 and 3 as well as sessions 3 and 4. This trend may suggest that the participants were becoming comfortable

using the shared interface and were gaining intuition on the effect of the system on the performance.

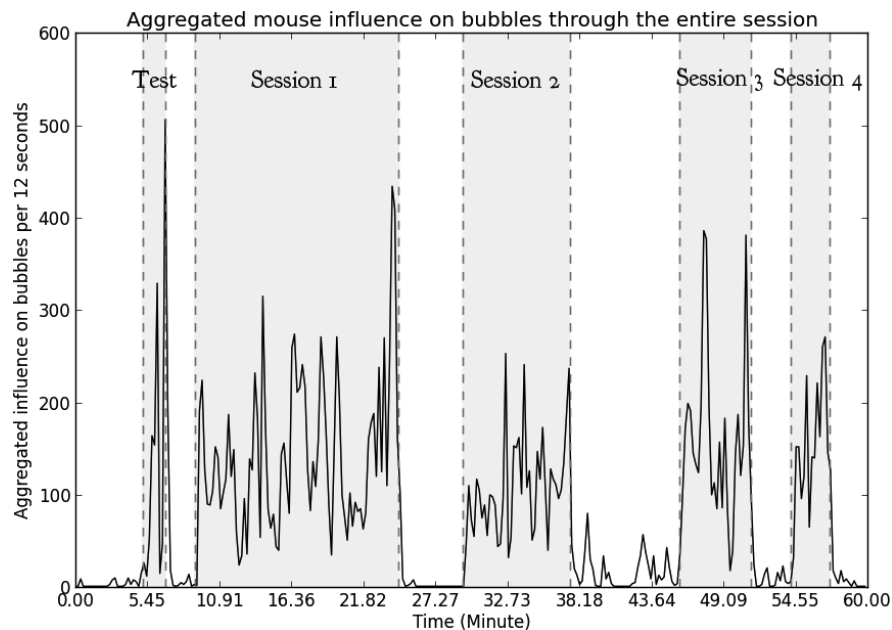


Figure 5. Aggregated mouse influence on graphical bubbles throughout the entire improvisation session

Table 1 summarizes the activity of the remote participants with respect to the duration, number of participants, number of influences on bubbles, average number of influences on bubbles, and number of influences on bubbles per minute in each improvisation session. We mentioned in the previous section that the total number of unique remote participants who interacted with the interface was 71. According to the table, however, each improvisation session had only approximately 20 to 30 participants and we also see that the number of participants generally decreased as the web-streaming session progressed. We assume from this data that a fair amount of individual remote participants were either in and out of each improvisation session or decided to just observe the performance passively depending on the session.

Table 1. Summary and comparison of each improvisation session

	Session 1	Session 2	Session 3	Session 4
Performance time (second)	928	485	322	178
Number of participants	29	26	26	19
Number of influence	11063	4420	4316	2301
Average number of influence	381.48	170	166	121.1
Average number of influence per minute	20.04	15.53	19.71	9.14

Regardless of who actually participated in each session, this table suggests that sessions 1 and 3 were the most fruitful and engaging sessions. The box plot in Figure 6, which describes the average bubble influence per minute in each improvisation session, reinforces this observation. The graph shows that activities during session 1 and 3 are slightly higher than the other two. We think the first session had high participation by the

visitors because it was the beginning of the performance and got many people excited to participate in the performance. In contrast to the first session, the reason why we think the third session also had high levels of interactivity is related to our previous observation that the remote participants have learned the interface by the third session and knowing how to effectively manipulate the shared interface collaboratively.

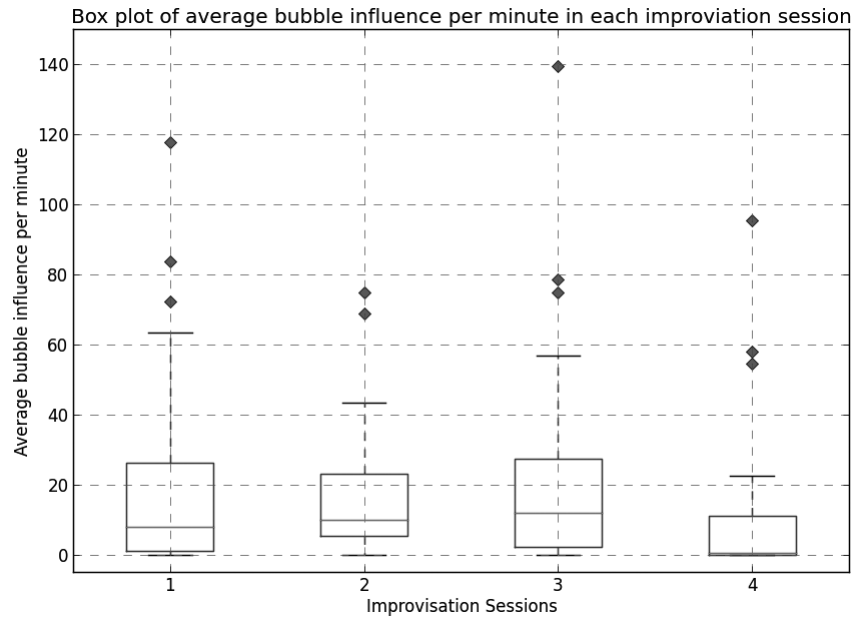


Figure 6. A box plot of average bubble influence per minute in each improvisation session

7. Comments from Participants During the Pilot Run

We collected approximately 30 comments during the live improvisation session. The collected comments ranged from appraisal of the performance, questions on technical implementation of the system, and also how they like to shape the music for the pianist. One of most common comments collected from participants was regarding the interpretation of the visual representation by the pianist. One participant asked “How does Tae interpret the circles on screen and translate it into what he plays?” We had explained how the improvisation session works through the live video stream and also in the instructions on the webpage that the pianist reads the accumulated weight of the composer bubble size to guide his improvisation, but the amount of comments we received shows that our effort in having a mutual understanding in the collaborative music creation between the pianist and remote participants did not always perfectly work out. We also received fair amount of comments that we sought from remote participants including comments about how to improve the next improvisation session. These comments include: “I like the slower start. It’s a nice way to ease into it,” “I like the Stravinsky to Bach transition,” “More Stravinsky than just the Rite chord!,” and “What sort of juxtapositions in particular really evoke Edinburgh for you?” These comments suggest to us that some remote participants were truly interacting with the interface to influence the course of improvisation with the pianist and the director.

8. Conclusion

In this paper, we have presented Repertoire Remix, a system that enables remote audience members to dynamically suggest their musical preferences for live web-streaming musical improvisation sessions. The web browser-based interface encourages remote participants to collaboratively use “stirring” mouse gestures to influence the size of graphical bubbles that contain composers’ names. The accumulated weight of these bubble sizes and other properties of the bubbles, such as position and texture, are then interpreted by talented musicians to improvise in a certain way in real time. The paper also addresses a number of challenges in designing a real-time shared music-style arranging environment (SMSAE) for a networked live improvisational session and interprets the result of a performance by assessing the participants’ mouse gestures collected during the pilot run.

In summary, Repertoire Remix was both an experiment and an interactive remote musical performance. Furthermore, the first pilot run of the Repertoire Remix system succeeded in creating a novel music piece through collaborative interaction among remote participants, the pianist, and the director. This study generated new insights into Network Music Performance by receiving musical preferences suggested by remote audience members and using those to create an improvisational session responding to the system status in realtime.

9. Acknowledgment

The author would like to thank Muriel R. Cooper Professor Tod Machover, Simone Ovsey, and Ben Bloomberg for their help in the creation of the Repertoire Remix web application. The author would also like to thank Tae Kim for his collaboration in making the pilot improvisation session a successful one. Thanks to the members of the Opera of the Future group at MIT Media Lab for your moral support.

References

- Baird, K. C. (2005). Real-time generation of music notation via audience interaction using python and gnu LilyPond. In *Proceedings of the 2005 conference on New interfaces for musical expression*, pages 240–241. National University of Singapore.
- Barbosa, Á. (2003). Displaced soundscapes: A survey of network systems for music and sonic art creation. *Leonardo Music Journal*, 13:53–59.
- Blaine, T. and Fels, S. (2003). Contexts of collaborative musical experiences. In *Proceedings of the 2003 conference on New interfaces for musical expression*, NIME ’03, pages 129–134, Singapore, Singapore. National University of Singapore.
- Bryan-Kinns, N., Healey, P. G., and Leach, J. (2007). Exploring mutual engagement in creative collaborations. In *Proceedings of the 6th ACM SIGCHI conference on Creativity & cognition*, pages 223–232. ACM.
- Burk, P. (2000). Jammin’ on the web—a new client/server architecture for multi-user musical performance. In *ICMC 2000*. Citeseer.
- Duckworth, W. (1999). Making music on the web. *Leonardo Music Journal*, pages 13–17.
- Edinburgh International Festival (2013). *Festival City*. <http://www.eif.co.uk/festival-city>.

- Freeman, J. (2007). Graph theory: interfacing audiences into the compositional process. In *Proceedings of the 7th international conference on New interfaces for musical expression*, pages 260–263. ACM.
- Freeman, J. (2008). Extreme sight-reading, mediated expression, and audience participation: Real-time music notation in live performance. *Comput. Music J.*, 32(3):25–41.
- Freeman, J. (2010). Web-based collaboration, live musical performance and open-form scores. *International Journal of Performance Arts and Digital Media*, 6(2):149–170.
- Freeman, J., Varnik, K., Ramakrishnan, C., Neuhaus, M., Burk, P., and Birchfield, D. (2005). Auracle: a voice-controlled, networked sound instrument. *Organised Sound*, 10(3):221.
- Guardian (2013). Repertoire Remix. <http://www.guardian.co.uk/music/interactive/2013/jul/05/tod-machover-festival-city-interactive-live-stream?INTCMP=SRCH>.
- Hajdu, G. (2003). Quintet. net—a quintet on the internet. In *Proceedings of the International Computer Music Conference*, volume 29, pages 315–318.
- HTML5 Canvas (2009). <http://www.w3.org/TR/2009/WD-html5-20090825/the-canvas-element.html>.
- Machover, T. (1996). Brain Opera. <http://brainop.media.mit.edu>.
- Miletto, E. M., Pimenta, M. S., Bouchet, F., Sansonnet, J.-P., and Keller, D. (2011). Principles for music creation by novices in networked music environments. *Journal of New Music Research*, 40(3):205–216.
- MIT Media Lab (2013). Repertoire Remix: The Sounds of Edinburgh. <http://blog.media.mit.edu/2013/08/repertoire-remix-sounds-of-edinburgh.html>.
- MongoDB (2011). <http://www.mongodb.org/>.
- Mongoose (2011). <http://mongoosejs.com/>.
- NodeJS (2011). <http://nodejs.org/>.
- NowJS (2011). <https://github.com/Flotype/now>.
- van Troyer, A. (2013). *Festival City* online score making environments. <http://edinburgh.media.mit.edu/scores/>.
- WebSocket (2011). <https://tools.ietf.org/html/rfc6455>.