Networked Virtual Environments as Collaborative Music Spaces

Cem Çakmak
Center for Advanced Studies in Music
Istanbul Technical University
Istanbul, Turkey
cemcakmak3@gmail.com

Anıl Çamcı
Electronic Visualization Lab
Univ. of Illinois at Chicago
Chicago, US
anilcamci@gmail.com

Angus Graeme Forbes
Electronic Visualization Lab
Univ. of Illinois at Chicago
Chicago, US
aforbes@uic.com

ABSTRACT

In this paper, we describe a novel multimedia system for networked musical collaboration. Our system, called *Monad*, offers a 3D virtual environment that can be shared by multiple participants to collaborate remotely on a musical performance. With Monad, we explore how various features of this environment in relation to game mechanics, network architecture, and audiovisual aesthetics can be used to mitigate problems inherent to networked musical performance, such as time delays, data loss, and reduced agency of users. Finally, we describe the results of a series of qualitative user studies that illustrate the effectiveness of some of our design decisions with two separate versions of *Monad*.

Author Keywords

Networked music performance; virtual environments; game mechanics; audiovisual performance systems.

ACM Classification

H.5.5 [Information Interfaces and Presentation] Sound and Music Computing—Systems, H.5.2 [Information Interfaces and Presentation] User Interfaces—Auditory (non-speech) feedback, H.5.2 [Information Interfaces and Presentation] Multimedia Information Systems

1. INTRODUCTION

Musical collaboration over a network differs from performing with other musicians in the same physical space in many ways. For instance, a lack of embodied interaction can impede the communication between the participants of a networked performance. Furthermore, such performances are intertwined with technology to the extent that the expressive medium becomes a technology itself. Making networked music thus requires a different approach not only to performing, but also to designing instruments for such performances. In this paper, we outline some of the issues inherent to networked music and describe some of the design choices we made to address these.

Digital music instruments display a great variety in terms of their characteristics and functions. As a result, studies in this field focus on a great variety of topics ranging from the epistemic nature of digital instruments [9] to the interaction methods they utilize [19]. Our relationship with new instruments continuously evolve alongside the media through which we are exposed to them. For instance, virtual environments afford a medium for live electronic music performances, and offer unique ways of interaction [5]. Although the use of virtual spaces for networked performance dates back to early days of computer networking [2], its potential reaches new extents with new technologies. Modern high-speed networks can transmit many types of media, allowing networked instruments to make use of graphics, video, text as well as sound. For instance, the artist Atau Tanaka's work *MP3q* [18] makes use of a graphical text interface for a participative music system on the web browser. The artists Angus Forbes and Kyomitsu Odai's interactive multimedia composition *Annular Genealogy* [7] is comprised of aural and visual engines that exchange information between each other over a wireless network. The network band Glitch Lich [13] builds network performance instruments, where 3D visualizations are used to improve the communication with the audience.

Over the last decades, multiplayer games have proven to be successful venues for stimulating competitive and collaborative behavior in networked settings. As a result, we observe various technical and aesthetic characteristics of computer gaming being inherited by networked instruments [16, 11]. For instance, Chad and Curtis McKinney have developed a network music engine inspired from video game synchronization systems [12]. Similarly, Rob Hamilton's *q3osc* and *UDKOSC* are modified versions of existing game engines dedicated to musical performance within virtual environments [8].

Over the past decade, researchers have offered various categorizations of networked music systems, such as Blaine and Fels' collaborative interface contexts [3], and Föllmer's three-dimensional net music space [6]. These evaluations indicate how diverse performances can be in terms of their level of complexity, interaction types, and network characteristics.

In networked performances, timing, which is a crucial element of music in general, can become unpredictable due to technical constraints [15]. This can hinder the sense of copresence in a multiplayer performance. The artist Álvaro Barbosa demonstrates how latencies in networked music can never be completely overcome based on his calculation of the bidirectional transmission time between two opposite points on the globe that communicate under perfect conditions (i.e. data transfer in light speed, unrestricted bandwidth) [1]. Accordingly, we treated network characteristics that may obstruct traditional performances as a natural constituent, and we designed our system around these characteristics. For instance, time delays can be considered as a reverbera-

NIME'16, July 11-15, 2016, Griffith University, Brisbane, Australia.

tion of the network infrastructure [18]. Instead of attempting to deal with the technical bottlenecks of the telecommunication system involved in a network performance, we approached the matter by focusing on the performative and aesthetic qualities of our audiovisual environment. This can be described as a virtual site-specific approach, similar to how an artist can compose for a specific concert space.

2. *MONAD*

Monad is a networked multimedia instrument for electronic music performance. Users interact with virtual objects in a 3D graphical environment to control sound synthesis. The visual objects in *Monad* expand upon the idea of optical discs with the addition of interactivity, real-time synthesis parameters, and three dimensional motions. Every virtual object in a *Monad* performance is accessible by all participants rather than being assigned to individual performers. The objects therefore act less like personal music instruments and more like shared components of a musical collaboration.

Monad clients are able to control the timbre, dynamics, filtering, and temporal characteristics of each graphical sound object. Each player can not only manipulate various parameters of the existing objects but add and remove them. A global Perlin noise function provides gradual randomized movements for each disc in the z-axis. The random seeds for each disc's noise generator are determined by the server so that all clients observe the same randomized movements of the objects.

Non-musical communication between performers is an important element of a musical collaboration, where performers can use bodily gestures rather than musical cues to signal their intents. This type of communication can be lost in networked music due to the disembodied nature of the performance. In *Monad*, we implemented a chat system to facilitate the communication between players, following other network ensembles such as Glitch Lich [10]. We utilized this system to study both the effects of non-musical cues in networked performances and the role of such cues in the audience's appreciation of a networked performance.

2.1 Software

The *Monad* software is designed using the C++ multimedia toolkit *openFrameworks*[1]. For sound synthesis, we use the openFrameworks addon ofxTonic[2]. Furthermore, we implemented the graphical user interface of our system using the ofxUI addon[3]. Users are provided with a binary file based on their platforms. The source code of the system for the client and server nodes can be downloaded at `http://github.com/occak`.

2.2 Interface

The *Monad* interface, as seen in Fig. 1, offers several modules. On the top left corner is a UI window, which allows users to add new grooves, as well as to modify the texture type, rotation speed, texture density, radius, and z-axis motion of existing grooves individually. A console beneath this window reports each user's actions and the rewards granted by the server. Next to the UI window is a chat stream overlay. On the right-hand side of the screen are bars that display each user's remaining resources. On the bottom of the screen, another UI window for global changes allows the users to toggle or reset the z-axis motion of all objects globally. A separate button next to these controls allows the users to toggle the visibility of the chat stream. Besides these visual UI elements, the users can utilize the W, A, S, D keys on their keyboards to scroll through discs and texture types. Furthermore, the users can change the camera perspective with simple click and drag interactions.

Figure 1: A screen-shot from a Monad v0.2 performance with 3 collaborators.

2.3 Agency

During the initial evaluation of our system with remote players, a strong preference for a clear representation of each user's presence within the environment was expressed by the participants. Furthermore, the users noted feeling dissatisfied when their actions were not evident in the audiovisual output. This is in line with Tanaka's findings on how users expect an instrument to be responsive and to give them a clear *sense of musical agency* [18] within an ensemble.

We found that a fundamental method of improving agency is to articulate each player's identity within the virtual environment. Furthermore, a user's presence should be clearly highlighted to not only themselves but also the other players, effectively eliminating anonymity in the performance. In an early prototype, our chat system relied on IP addresses to label users. While this was sufficient to differentiate between the participants of a conversation, it was described as not being clear enough to keep track of the identities. A self-assigned nick name was reported to offer a better sense of agency. We also observed that even when each player is able to pick a unique nickname, they also announce their real names to the other participants to reveal their identities. Furthermore, each player's sense of presence is reinforced with individual color labels that are persistent throughout the UI as seen in Fig.1.

2.4 Topology

The topology of a network dictates what is communicated during transmission, the order in which the communication happens, and the direction of information flow. For example, a centralized network takes information from the players' input and sends it to a center of activity, where the data will be analyzed [20]. Decentralized systems, on the other hand, enable direct interaction between the participants but are limited by the computational capacity of each node. However, Rohrhuber argues that the network topology alone does not provide a complete representation of a network music system [14]. The *causal topology* of a networked performance becomes an integral aspect of the audience experience. In other words, solely focusing on the logical organization of the network cannot fully reflect the

[1]http://www.openframeworks.cc
[2]http://github.com/TonicAudio/ofxTonic
[3]https://github.com/rezaali/ofxUI

end product. To investigate the effects of our network topology we conducted user evaluations with performers and audience members, which we will discuss in Section 4.

With *Monad*, we designed a network structure based on a server-client relationship as illustrated in Fig. 2. All users are clients with equal privileges, while the server, which is hosted at one of the performers' computer, maintains the shared environment. Upon starting the program, the users are expected to enter the server's IP address and their unique nicknames. In the latest version of our system, up to 4 clients are able to connect to the server. The server initially assigns equal amounts of resources to each node and broadcasts each participant's details to the others. During the performance, the server reports the momentary states of the environment to each participant. Here, the server functions as a *shared object* that all clients are able to impose changes on. Furthermore, the server maintains an automated rewarding system, which we will further describe in the next section.

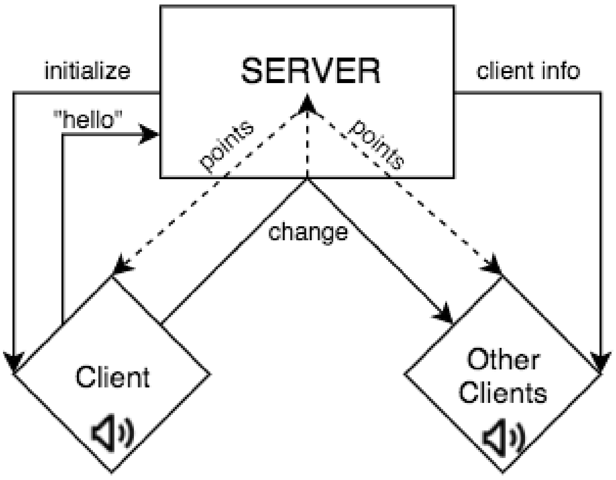

Figure 2: Monad v0.2 network structure for individual clients.

3. MITIGATING NETWORK LATENCY

Given the physical limitations elaborated in the first section, an absolute temporal uniformity cannot be achieved in networked music systems. As a result, while transmission latency can be utilized as a creative component of networked performances [4], it is nevertheless an inherent component of such systems. Given the importance of timing in musical performances, it is essential for a networked instrument to address the effects of latency so that these do not impede creative flow. We have utilized several techniques, as descrubed below, to mitigate the effect of latency during a performance.

3.1 Timbral vs. Temporal Characteristics of Sound

In *Monad*'s audiovisual output, the sound synthesis routines that are mapped to each graphical object are identical in each client's node. However, due to network latency, the phase relationships between individual synthesis modules can vary from one node to the other. To alleviate the effects of such temporal non-uniformities across participants, musical interactions in *Monad* are designed to focus on textural qualities of sound. Instead of triggering discrete musical events, the players initiate looping patterns and then manipulate the timbral, spectral and dynamic characteristics of the patterns during the course of a performance. This maintains that the players are exposed to a musical structure that is consistent across individual nodes.

3.2 Game Mechanics

Network delays between players pose the risk of disengaging them from a common musical purpose; in order to increase cooperation between players and emphasize their togetherness in a performance, we introduced simple game mechanics to our environment. With this, we aim to elicit certain behaviours from all participants to improve conscious interactions during a performance. Different from regular games, the foremost function of the implemented mechanics in *Monad* is to facilitate musical interaction without extra-musical goals. Accordingly, the mechanics are intended to moderate a musical performance rather than rendering sound as a byproduct of non-musical interactions. Furthermore, we wanted the mechanics to nurture emergent musical behavior across participants instead of confining performance to a predetermined structure.

An internal economy mediates the performances in *Monad*. All clients begin with an equal amount of resources (i.e. points). Each action carried out during a performance (e.g. changing synthesis parameters, adding new grooves) costs the players points. When a user runs out of resources, he or she is unable to perform any further actions. To manage this resource system, we tested two different rewarding mechanisms based on either a client-driven or a server-driven approach.

In *Monad* v0.1, we implemented a *client-driven* reward system that relies on players to give each other points if they "like" the changes performed by other players. Each player's most recent actions are displayed as a stream of buttons on the user's screen as seen on the left-hand side of Fig. 3; clicking on these event reports reward points to the corresponding player. In this system, the clients are in charge of the rewarding mechanism, thus their decision-making is critical to the internal economy.

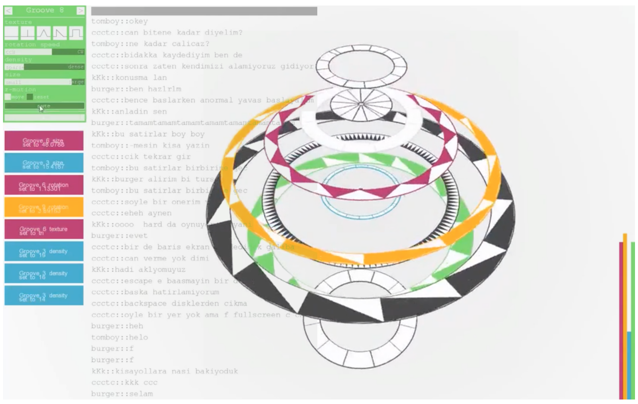

Figure 3: Screenshot from a 4-player performance with Monad v0.1. The remote players' actions are streamed on the left-hand side in the form of color-labeled buttons which a player can click on to reward these actions.

In *Monad* v0.2, we developed a *server-driven* reward system that does not depend on players' approval, but is automated by the server. As illustrated in Fig. 2, the server keeps track of all players' actions and assigns points to players who make changes that are analogous to those by other players. This model is inspired from the call-and-response style often used in improvisatory performances. In the current implementation, similarity of actions between participants are rewarded when these actions are performed within a limited time span (i.e. 10 seconds). As a result, users are encouraged to pay attention to other performers, and to promptly respond to their actions in order to retain resources.

3.3 Visual Aesthetics and Crossmodality

In accordance with the focus on timbral qualities of sound in *Monad*, we have designed the corresponding virtual objects in a way that reflects the same patterned and looping characteristics in the visual domain rather than consisting of sparse visual events. Disc-like shapes seen in Fig. 3 are used by the players to control the sound output. This design was inspired from Evgeny Sholpo's Variophone, which was a graphical sound system based on optical discs, from the early 20th century [17].

Furthermore, we particularly focused on the mappings between auditory and visual features to facilitate perceived coherence across modalities and to improve the sense of causality between visual changes performed by the players and their sonic outputs. In order to secure an audiovisual unity, all drawing and synthesis operations are handled by client nodes. Although the server maintains the momentary global state of the environment, the transmitted information consists only of numeric values pertaining to visual coordinates and synthesis parameters, which puts a minimal strain on the network.

4. EVALUATION

4.1 Performer Evaluations

In a series of preliminary evaluations, different versions of *Monad* were tested with a total of 10 users aged between 24 to 32 years. Participants were first given a basic explanation on how to use the software. The tutorial was followed by an exercise performance which was recorded. The users were then asked to respond to survey questions with Likert-type ratings (i.e. from 1 to 5), and free-form verbal responses. Participants rated a set of statements that determined their background ("I am a musician", "I often play computer/console games"), *Monad*'s learning curve ("Getting to learn the program was easy", "I felt getting better as I played"), their relationship with other players ("Communication with others was easy", "I felt competitive against other players", "I rewarded players due to their specific actions"), and their general experience ("UI controls were practical", "The musical output was satisfying", "I would like to play it again").

Results indicated that the participants of this initial survey were all music performers, and the group consisted primarily of music professionals. But, the users were evenly divided in terms of their gaming experience: one half rated their computer gaming experience high, while the other half asserted that they rarely played video games. Nevertheless, all participants indicated that learning to use the software and communicating with others were fairly easy. The results showed that all participants enjoyed the musical output, felt that they improved as they played, and would like to play again. Interestingly, the amount of competitive behavior observed across the board was lower than we anticipated. This reflected as a collaborative attitude throughout each performance, where participants called for help when low on resources and awarded each other points to keep each other in the game.

Following these initial studies, several *Monad* performances were carried out with 3 to 4 players. Involving more players in the same environment naturally increased the overall activity at a given time, which in return impacted the musical output. During performances with v0.1, which relied on the client-driven reward mechanism, the sound output tended to get cacophonous during increased activity. Nevertheless, interesting musical dynamics appeared in such situations, where users began to rapidly alter the textures on a groove, causing noticeable changes in timbre. Musically, such actions resulted in solo-like gestures where other users displayed a musical inactivity and provided resources to the 'soloist' to keep them in the game.

The client-driven rewarding system was described by some of the players as detaching them from the performance, since they felt they needed to stop interacting with the graphical objects and press update buttons to reward other players. Overall, most participants expressed that they used the rewarding mechanism just to help each other remain in the game rather than paying attention to individual actions. Other players did report giving rewards based on performance; one user stated that instead of reading the action labels on the buttons, she pressed them whenever a button synced with a change in music that she enjoyed.

A major motivation to try other game mechanics was to address the cacophonic quality of the sound output. We wanted to motivate players to follow others, which we expected to result in more unified structures. Participants who performed with both versions of the system reported that the second version which relied on the automated rewarding system based on similarities felt more intuitive to perform with. The audio recordings of the performances with the second version of the system evidenced a more coordinated and balanced collaboration, since the mechanics led the players to take actions that are similar. This implies that changing the mechanics of the system has inherently affected the style of music created with the instrument.

4.2 Audience Evaluations

Two public *Monad* performances were carried out with the first and second versions of the system. While the first performance had 15 viewers, the second concert had 35 audience members present. Brief surveys were carried out with the members of the audience after each performance. Similar to the performer studies, audience survey first sought to determine the music and gaming backgrounds of the participants, followed by their experience with electronic music concerts and more specifically networked music performances. The survey also included questions regarding the game mechanics, and lastly the participant's personal experience of the performance.

Figure 4: A public performance with Monad v0.2 with local players in Istanbul and remote players from Vancouver and Berlin.

On a Likert-type scale, the participants mainly reported having experience with electronic music and networked performances, but indicated that they lacked experience with gaming. A majority of the viewers expressed having enjoyed the performance and were interested in experiencing it again.

In both performances, very little information about the system were disclosed to the audience in advance. The audience members were able to view a projection of one of the local performers' screen as seen in Fig. 4. While nearly half of the survey-takers reported not having noticed the underlying game mechanics throughout the performance, the rest stated that it became noticeable as the performance developed. Interestingly, the latter group rated the coherence between sounds and visuals higher than the former group.

One of the participants stated that the disembodied (i.e. laptop-based) and process-oriented interactions were boring and non-musical. Other participants have expressed the local players could have utilized the physical space more expressively. Local performers, however, expressed that physical gestures would go against the democratic nature of the performance by way of overpowering the role of the remote players in the audiovisual output.

5. CONCLUSIONS AND FUTURE WORK

Building upon the study described in this paper, we intend to explore other game-like rules and their effects on musical dynamics between the players of a networked performance. We also plan to extend our user interface to allow for more complex sonic interactions. Furthermore, to improve the ease of participation, we are currently porting our system to a browser-based application, using WebGL and WebAudio.

Besides the issues discussed in this paper, networked music can also pose problems in terms of the presentation of a performance to the audiences in a concert space. We intend to explore how *Monad*'s game-based interactions can be communicated to the audiences effectively when most or all of the performers are in remote locations. For instance, we plan to evaluate how disclosing information about the system with varying levels of detail might alter the audience's appreciation of a networked performance.

In this paper, we discussed various aspects of networked music performances that make use of multimedia and virtual environments. Furthermore, we described the methods we used to mitigate latency issues that are inherent to such environments. These methods included the uses of temporal uncertainty as a musical tool, a custom network topology, and game mechanics. In our user evaluations with *Monad*, we concluded that collaborative music-making in virtual spaces can inherit elements from c omputer gaming in terms of interfacing, agency, topology and mechanics to improve the performers' sense of active collaboration, and the audience's appreciation of the performance.

6. ACKNOWLEDGMENTS

We would like to thank the MIAM Sonic Arts Department for providing a workspace and performance venue for this project. Also many thanks to Barış Demirdelen for helping with debugging the code, Gizem Oruç and Kıvanç Tatar for regularly participating in performances and surveys, Görkem Özdemir for providing feedback and assistance with the UI, Çağrı Erdem and the *A.I.D.* concert organizers for inviting us to take part in their event, and everyone who participated in the surveys.

7. REFERENCES

[1] A. Barbosa. Displaced Soundscapes: A Survey of Network Systems for Music and Sonic Art Creation. *Leonardo Music Journal,*, 13:53–59, 2003.

[2] J. Bischoff, R. Gold, and J. Horton. Music for an interactive network of microcomputers. *Computer Music Journal*, 2(3):24–29, 1978.

[3] T. Blaine and S. S. Fels. Contexts of collaborative musical experiences. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 129–134, Montreal, Canada, 2003.

[4] J. P. Cáceres and A. B. Renaud. Playing the network: the use of time delays as musical devices. In *Proceedings of International Computer Music Conference (ICMC)*, pages 244–250, Belfast, Northern Ireland, 2008.

[5] L. Carroli. Virtual encounters: Community or collaboration on the internet? *Leonardo*, 30(5):359–363, 1997.

[6] G. Föllmer. Electronic, aesthetic and social factors in net music. *Organised Sound*, 10:185–192, 2005.

[7] A. G. Forbes and K. Odai. Iterative synaesthetic composing with multimedia signals. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 573–578, Ljubjiana, Slovenia, 2012.

[8] R. Hamilton. *Perceptually Coherent Mapping Schemata for Virtual Space and Musical Method*. PhD thesis, Stanford University, CA, 2014.

[9] T. Magnusson. An epistemic dimension space for musical devices. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 43–46, Sydney, Australia, 2010.

[10] C. McKinney and N. Collins. Liveness in network music performance. In *Proceeding of Live Interfaces: Performance, Art, Music*, Leeds, UK, 2012.

[11] C. McKinney and N. Collins. An interactive 3D network music space. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 400–405, Daejeon, Republic of Korea, May 2013. Graduate School of Culture Technology, KAIST.

[12] C. McKinney and C. McKinney. Oscthulhu: Applying video game state-based synchronization to network computer music. In *Proceedings of the International Computer Music Conference (ICMC)*, 2012.

[13] C. McKinney and C. McKinney. Visualization of network based multi-user instruments. In *Proceeding of Live Interfaces: Performance, Art, Music*, Leeds, UK, 2012.

[14] J. Rohrhuber. *The Cambridge Companion to Electronic Music*, chapter Network Music, pages 140–155. Cambridge University Press„ Cambridge, 2007.

[15] J. Rohrhuber and A. de Campo. Waiting and uncertainty in computer music networks. In *Proceedings of the International Computer Music Conference (ICMC)*, 2004.

[16] J. Rudi. Computer music video: A composer's perspective. *Computer Music Journal*, 29(4):36–44, 2005.

[17] A. Smirnov. *Sound in Z: Experiments in Sound and Electronic Music in Early 20th Century Russia*, chapter Graphical Sound, pages 175–236. Koenig Books, London, 2013.

[18] A. Tanaka. *Consuming Music Together: Social and Collaborative Aspects of Music Consumption Technologies*, chapter Interaction, Experience and the Future of Music, pages 271–292. Springer, Netherlands, 35 edition, 2006.

[19] M. M. Wanderley and N. Orio. Evaluation of input devices for musical expression: Borrowing tools from HCI. *Computer Music Journal*, 26(3):62–76, 2002.

[20] G. Weinberg. Interconnected musical networks: Toward a theoretical framework. *Computer Music Journal*, 29(2):23–39, 2005.