GROCS 2008 Project Description

Project: NoteWorks

{ umbaugh, pturley, rlalexan, dapafi } @umich.edu

Abstract

We propose to design and implement a computer application that enables users to create sound experiences and musical compositions in a completely new way. In particular, our software will enable users to design dynamic temporal networks in which the nodes correspond to sound clips, and directed edges represent time and other relationships between nodes. Furthermore, we will embed functionality in the application so as to enable different instances of our software to interact with other musicians' networks so as to create a truly interactive, collaborative music experience. We will also release our software to any interested parties so they can extend it as they see fit (and set up their own musical networks at home).

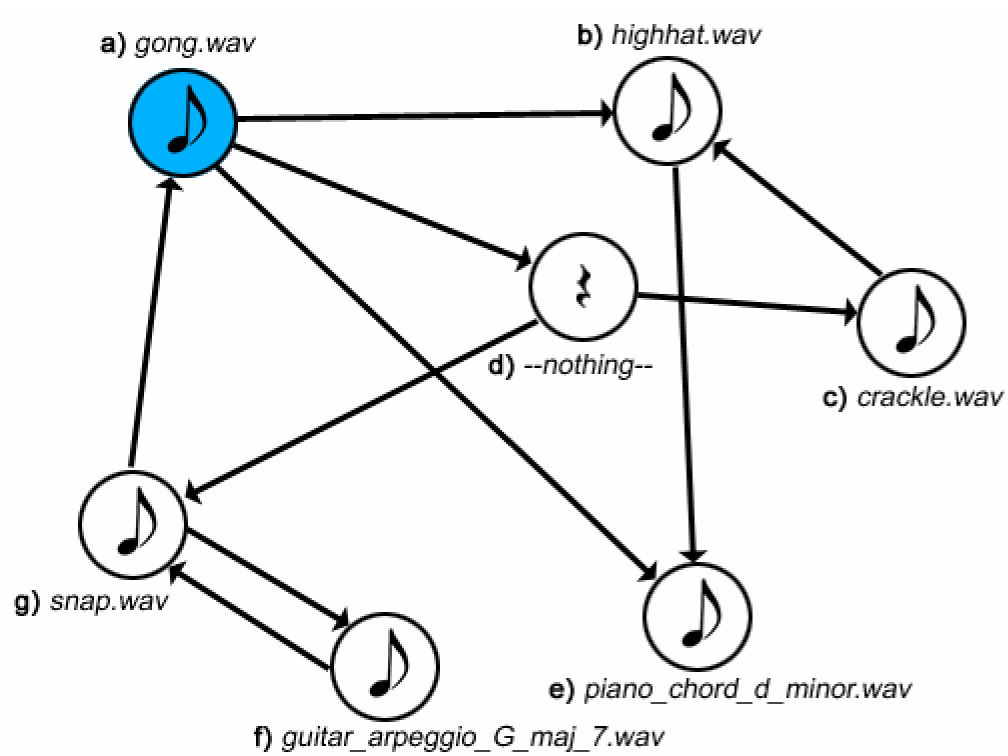
Introduction

In many regards, traditional musical composition has been a strictly linear affair. Composers often design their musical pieces note-by-note and chord-by-chord, around variations of several themes that might recur several times. Any interaction between notes must be envisioned outright by the composer and expressly embedded into the composition. While this musical paradigm has accounted for an incredible diversity and richness in our musical tradition – including nearly all Western musical traditions – there are other ways one might render musical expressions by designing more complex – and perhaps unexpected – interactions between the constituent *sounds* within the composition.

Our project, which we have codenamed *NoteWorks*, is a computer application that operates in many iterations of two simple stages. In the first stage, end users create a directed network with a simple drag-and-drop interface consisting of nodes and directed edges. Each node is associated with a certain sound clip (something the end user selects – including sound clips he or she has prerecorded). Directed edges connect nodes; if there is a directed edge from node *A* to node *B,* then this encodes that the *firing* (to be explained below) of node *A* will affect node *B.*

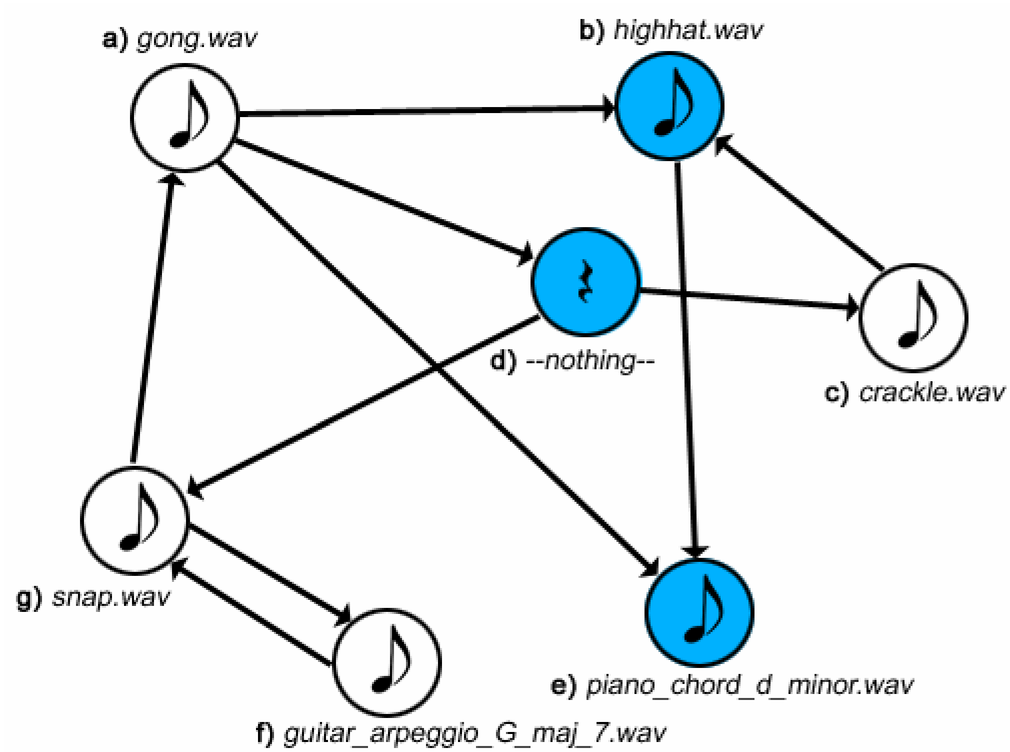
In the second stage, the user "starts" the network. That is, a given node or set of nodes S_0 (specified by the end user) is initially *fired* – and so the associated sound clip plays over the speakers – at time $t = 0$. In the simplest incarnation of our project, at $t = 1$, all nodes S_1 that are the targets of all edges whose sources are in S_0 are fired (and S_0 and S_1 are not necessarily disjoint sets). At $t = 2$, all nodes S_2 that are connected from the S_1 nodes fire, and so on. In this way, the network is reminiscent of a recurrent neural network.

Below is an example of four timesteps in a *NoteWorks* network (hereafter called a "notework"). The first node **a** (specified by the user) is fired at time $t = 0$, and so the sound *gong.wav* is played.



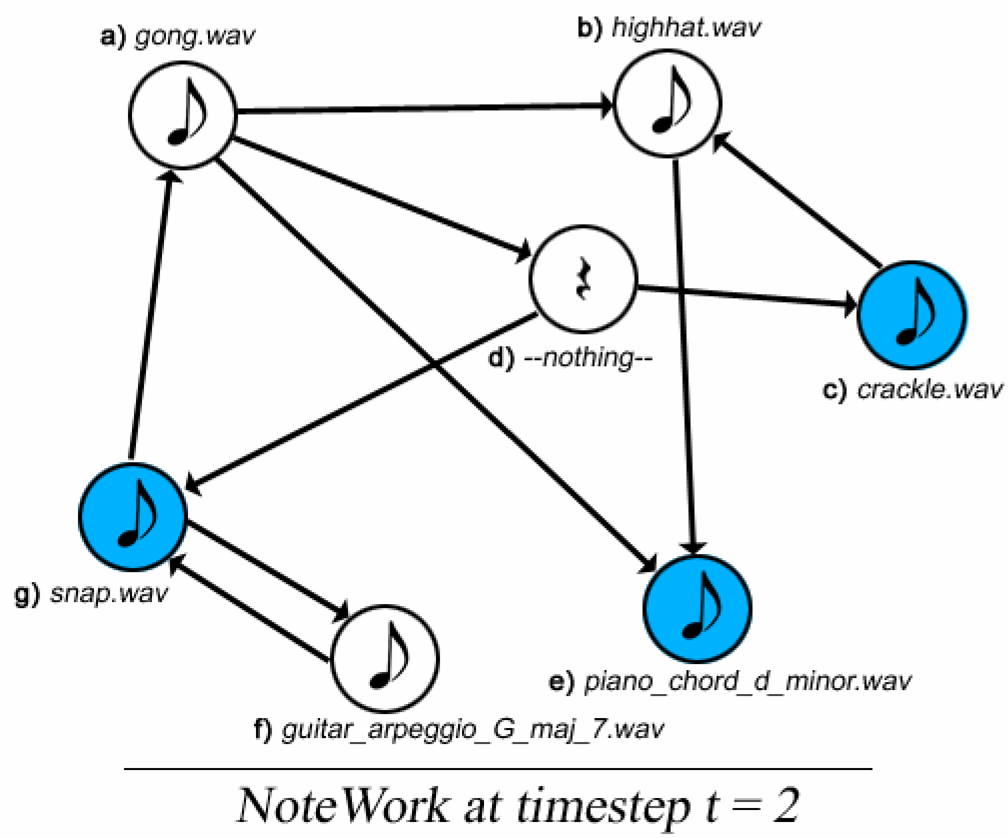
NoteWork at timestep t = 0

There are arrows going from **a** to the nodes **b, d,** and **e**, and so at time $t = 1$, nodes **b, d,** and **e** will be fired. In this case, a *highhat* sound and a *snap* sound will be emitted from the speaker. Since node **d** is a *rest* node, it will not emit any sounds; however, it *will* excite one of its own neighbors to fire on the succeeding timestep. This is one of the many possible extensions that we can embed into the application so as to expand the richness of created compositions.

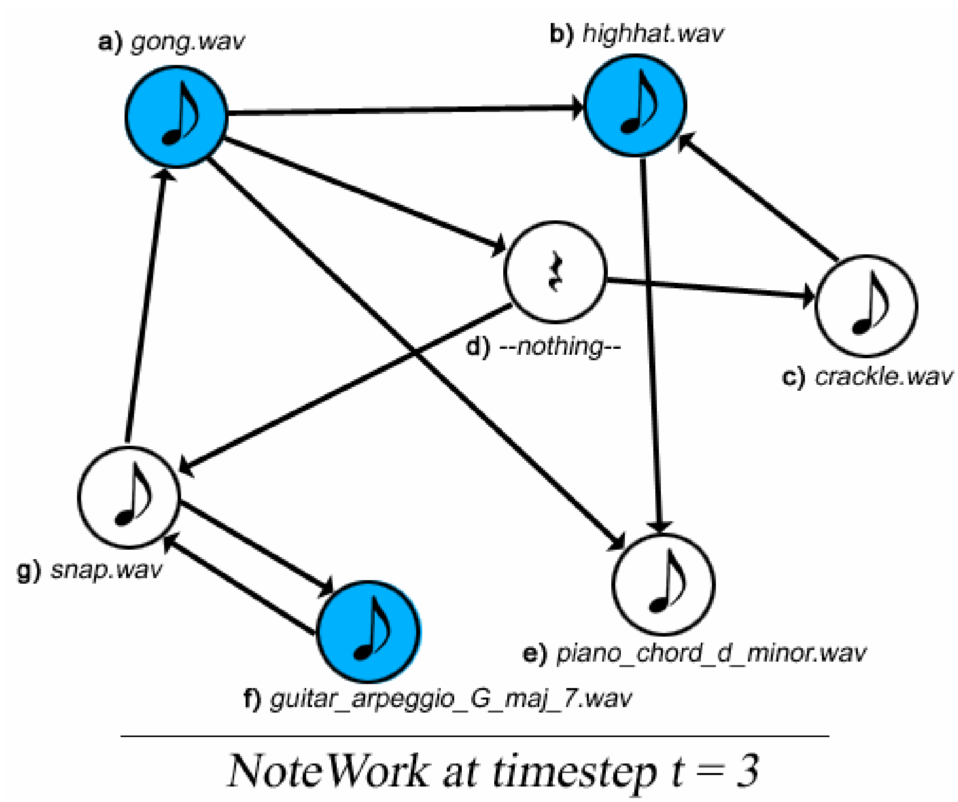


NoteWork at timestep *t = 1*

On the next timestep, both nodes **c** and **g** will be fired, but so will node **e**, since node **b** excites node **e** for the subsequent timestep. For timestep *t = 2*, therefore, we will expect to hear a *snap*, a *crackle*, and a *d-minor piano chord* at the same time.



For timestep *t=3*, nodes **a, b** and **f** will be fired. And so on.



In addition to sound clips and the directed edges between the nodes, nodes will also support additional properties that characterize their behavior, and the behavior of the network at large. For the sake of easy visualization, the nodes in the example above all send impulses to their neighbors exactly one timestep after being fired. Put another way, the example above describes a network in which all of the nodes fire in a kind of lock-step fashion. This needn't be the case, however, and we're planning to build in properties such as delay and sustain so as to maximize the flexibility and possibility of the network. We're also planning on building different node types, such as a node that must receive a certain threshold amount of impulses within a certain time frame before it can fire, or nodes that effect changes in the network that might not correspond directly to a sound.

In addition to a sound expression, the application will provide clear visual feedback of what's going on in the network to generate the sounds, so as to better understand the dynamics of the network.

We do not envision that our application will serve as a *replacement* for traditional music composition techniques. We *do*, however, hope that these kinds of techniques will *complement* existing music-production methodologies; for example, by helping musicians to identify and enact musical relationships quickly and easily – and also, we hope, in a way that is fun. Also, we plan to incorporate recording facilities so that composers can embed a run of their network into a linear sound file format, so as to listen to it later through a standard audio player application, or else perhaps embed it in any multimedia application they see fit.

Furthermore, we sincerely hope that users will be impressed with the profound ways in which even simple networks can represent some very sophisticated musical expressions, and through our project we want to encourage users to think about how they might apply network theory towards their own disciplines and interests.

Emphasis on Collaboration

One of the features of our application about which we are most excited is the facility we will include to extend *NoteWorks* across several computers over the Internet, and so create organic, multi-machine musical installation arrays that are possibly widely distributed in various locations on the globe – or possibly in the same room, so as to create a musical ensemble that is dynamic, and yet varies as an audience member walks about and comes closer to (and farther apart from) different *NoteWorks* installations. In particular, we plan on creating special nodes within the network that will "tunnel" to other installations through the Internet. Users will be excited because they will be able to see, first-hand, how their installations affect other installations, and vice versa.

Another focus of ours in the development of *Noteworks* is to position it so as to be a viable platform for public performances. Club DJs, for example, might have particular interest in our project. DJs traditionally weave their soundscapes by merging and changing between two or more songs to create unique, engaging, and coherent compositions. With our software, performers could create interesting networks ahead of time and perform in such a way that audience members would see a visual representation on how the dynamics of the network effect a musical expression.

It is a very high priority for us to make the interface as intuitive as possible. Existing software packages have the burden of high expense and high learning curves. Our mantra is to design the software in such a way that potential users will essentially already know how to use the application without a manual – at least for some basic compositions. The software will not only allow users to create compositions quickly and intuitively, but also it will foster interest in networks and graph theory. We plan to release the *NoteWorks* software to *anyone* who is interested, *for free* via the CSE and/or the SoMTD website(s), so they can continue their experimentations and set up their own installations at home – and with their friends, over the Internet. We will foster this community and encourage further development of the application through the community.

Equipment

We may require the use of some sophisticated recording equipment (and software) so as to provide a palette of good sounds with which users can work. For these purposes, we will utilize the Dudertstadt Center's audio studio, specialized workstations available through the Performing Arts Technology program in the School of Music, Theatre, and Dance, and experimental sound-design software available in the Interactive Systems laboratory in Computer Science and Engineering. For the demonstration, we can install our software on any Windows (and potentially Macintosh) machine with a reasonably good graphics card and a reasonably good sound card. We will need several machines to run the several instances of *Noteworks*, as well as, if available, projection monitors, a curved projection screen, and some high-end speakers. We believe that all of this can be provided by Michigan's media facilities upon reservation.

Participants

John Umbaugh is a second-year Masters student in Computer Science: Intelligent Systems at Michigan. He has worked as a software developer in the corporate Internet application space, and as a software architect and developer for the video game developer *Pyramid* in Tokyo, Japan, and also independently (http://gamesresearch.com). As an undergraduate at Case Western Reserve University, a major interest of his was the dynamics of biologically-inspired recurrent neural networks. He is an avid music fan and a sometime-musician. He has participated in three bands, including a Japanese *Taiko* drum troupe.

Patrick Turley is a second-year Masters student in Computer Science: Intelligent Systems at Michigan. He has worked as a research assistant at the United States Geological Survey on graph theory and traversal methods. In his undergraduate work at the University of Missouri-Rolla he was heavily involved with college radio eventually working his way to a station manager position. Patrick is proficient in many different instruments including drums, piano, harmonica, guitar, and bass. He has been a member of a wide variety of bands in the past and present, most recently notable the Phoenix Brass Band and MI5: Under Surveillance.

David Fienup, an Ann Arbor native, is a first year Master of Arts student in the Media Arts department. He holds a Bachelor of Arts in Music Performance from Albion College, with a concentration in Music Composition, and a minor in Economics & Management. David has spent the last two years in the Performance Art Technology department at the Bachelor level working on his portfolio before beginning the MA in Media Arts. His experience in music includes performance, composition (for music and film), theory, philosophy, history, music technology, band management, digital signal processing, computer music, sound recording, and sound production, as well as performing with multiple rock bands. His musical life has taken him through the genres of classical, jazz, blues, rock, funk, electronica, literary and dance collaborations, and free improvisation. He is proficient in guitar, and has played French horn, bass, lap-steel, piano, various hand drums, and turntables. Recently, he recorded live sound for the full-length film Zombie Apocalypse, by Ryan Thompson from Grand Valley State University. He has also composed and recorded the majority of the soundtrack. Currently, David holds a GSRA with The Internet Publication Task Force, which runs Block M Records, the UofM Record Label. David recently won the Block M competition: New Music on the Block, scoring him a recording contract, which will be published and distributed on iTunes.

Robert Alexander is a first-year Masters student in Media Arts. He received his Bachelor of Fine Arts in Performing Arts Technology from the University of Michigan School of Music where he worked for three years in the office of Computer Technology under Gregory Laman. His undergraduate thesis involved the creation of a new tool for working with electronic music that fused sight and sound in a digital space. He has written electroacoustic music for ten years using software such as Reason, Buzz, Max/MSP, Csound, Digital Performer, Pro tools, Kontakt, Reaktor, and Logic. Live performance venues have included the Threshold Electroacoustic Music festival, Sync 05, and New Music on the Block. He plays the piano, guitar, violin, and has performed vocally with the Michigan State Honors Choir.

Technology and Literature Review

We know of no projects in which dynamic temporal networks are used and manipulated to create any sort of artistic expression, though there are commercial tools such as Max/MSP in which composers can construct complex networks of sound objects so as to create virtually any sound imaginable. Nevertheless, we believe that it is difficult if not impossible to coax out the complexity (in other dimensions) that our dynamic temporal networks will be able to express. Furthermore, Max/MSP is a serious tool for serious composers and sound designers, and as such, its learning curve is reported to be extremely high. With *Noteworks,* we want to concentrate on simplicity and intuitiveness. Finally, we want to build *Noteworks* in such a way that it can be used as a viable platform for engaging performance; this is not something that falls within the core competencies of a complex package like Max/MSP.

There is a rich body of knowledge within the science of network theory. In particular, we are going to have to pay special attention to motifs, network centrality measures, cyclic networks, diffusion and contagion, and prestige. We are also going to have to take special measures to make sure that the user cannot easily create a network that emits a bunch of *noise*; after all, we want this to be an enjoyable experience for users. We also plan on consulting several neural network references to implement features like oscillating neurons (nodes) and neural threshold functions. We may also employ sophisticated network operations such as realtime preferential attachment, dynamic network growth (i.e. the growth of "neurons" in mid-execution), and syntactic network generation, with something akin to context-free grammars (for graphs).

References

Abbot, Curtis, and Gareth Loy. "Programming languages for computer music synthesis, performance, and composition." ACM Computer Survey 1985: 235-265. Google Scholar. University of Michigan Library. <http://doi.acm.org.proxy.lib.umich.edu/10.1145/4468.4485>

Réka, Albert, and Barabási, Albert-László. *Statistical Mechanics Of Complex Networks.* Reviews Of Modern Physics, Volume 74, January 2002

Barabási, Albert-László. (2003). *Linked: How Everything Is Connected to Everything Else and What It Means for Business, Science, and Everyday Life.* NYC: Plume

Chartrand, Gary. (1985). *Introductory Graph Theory*. Boston: Prindle, Weber, & Schmidt

de Nooy, Walter; Mrvar, Andrej; Batagelj, Vladimir. (2005). *Exploratory Social Network Analysis with Pajek.* New York: Cambridge University Press

Gleick, James Chaos: Making a New Science. New York: The Penguin Group, 1998.

Granovetter, Mark. The Strength of Weak Ties. (May, 1973). *The American Journal of Sociology*, Vol. 78, No. 6. 1360-1380

Jorda, Sergi. "New Musical Interfaces and new Music-Making Paradigms." NIME '01:

Proceedings of the 2001 conference on New interfaces for musical expression. Singapore: National University of Singapore, 2001.

Koch, Christof & Segev, Idan. (1998). *Methods in Neuronal Modeling: From Ions to Networks. Second Edition.* Cambridge, Massachusettes: The MIT Press

Newman, M. E. J. (2003). The Structure and Function of Complex Networks. *The Siam Review* 45, 167-256.

Pellman, Samuel. An Introduction to the Creation of Electroacoustic Music. Belmont, California: Wadsworth, 1994.

Pennycook, Bruce. "Computer-music interfaces: a survey." ACM Computer Survey 1985: 267-289. Google Scholar. University of Michigan Library. <http://doi.acm.org.proxy.lib.umich.edu/10.1145/4468.4470>

Roads, Curtis. The Computer Music Tutorial. Cambridge, Massachusetts: The MIT Press, 1996.

Tufte, Edward R. (1990). *Envisioning Information.* Cheshire, Connecticut: Graphics Press.

Tufte, Edward R. (1997). *Visual Explanations*. Cheshire, Connecticut: Graphics Press.

Watts, Duncan & Strogatz, Steven. (1998). Collective Dynamics of Small World Networks. *Nature* 393, 440-442.