Phillip Hermans

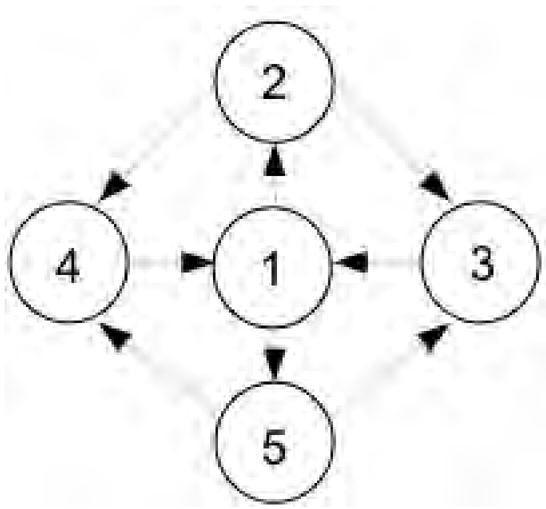
Paper : **Computer Modelling and Analysis of Goal-Oriented Acoustic Music Compositions**

Abstract:
This paper will describe the design, modelling and analysis of music compositions created by the author. Goal-oriented compositions refer to musical compositions in which the performers have some goal, that they attempt to reach by following the music and instructions found within the musical score. Three such compositions are presented in this paper: a percussion quartet, a trombone octet and a percussion trio. Models of these compositions were developed in MATLAB and simulations using these models allow for testing and experimentation during the composition process and analysis after a composition is complete.

The inspiration for the music compositions is derived in part from the engineering/computer science field of networked multi-agent systems. The distributed computing algorithms used in consensus problems[1] gives rise to emergent phenomenon such as synchronization of networked oscillators or flocking behaviour [2]. The use of these concepts in musical composition lends itself to computer modelling and statistical analysis, as well as a highly variable musical output. The computer models aid in experimentation during the composition process, testing of finished compositions and analysis after performance of a work.
Statistical methods of analysis are particular useful for algorithmic and generative works due to the multitude of realizations possible. The analytical tools provided in this paper are specifically for music compositions with discrete states. Future work includes the use of statistical models for the analysis of music compositions with continuous states as well as the application of these concept to other artistic media and forms.

Topic: Music

Authors:
Phillip Hermans
Dartmouth College,
Department of Digital Musics
www.digitalmusics.dartmouth.edu



Planar, Connected Digraph of "5 Choose 4" Percussion Quartet

References:
[1] Reza Olfati-Saber, "*Consensus and Cooperation in Networked Multi-Agent Systems*", Proceedings of the IEEE, Vol. 95, No. 1, 2007
[2] Reza Olfati-Saber, "*Flocking for Multi-Agent Dynamic Systems: Algorithms and Theory*", IEEE Transactions of Automatic Control, Vol. 51 No. 3, 2006

Contact:
phillip.hermans@dartmouth.edu

Keywords:
algorithmic composition, music analysis, consensus protocol, computer modelling, networked multi-agent systems

Computer Modelling of Goal-Oriented Acoustic Music Compositions

Phillip M. Hermans
Department of Digital Musics, Dartmouth College, Hanover, New Hampshire
website: http://digitalmusics.dartmouth.edu/~phermans/
e-mail: Phillip.M.Hermans.GR@dartmouth.edu

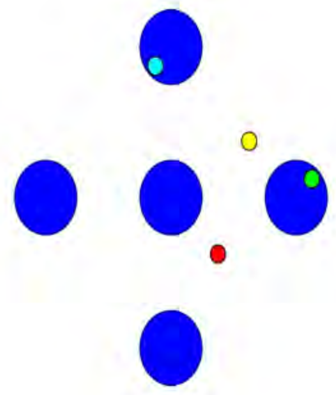


Fig. 1 Visualization of "5 Choose 4" Simulation

Introduction

This paper will describe the design and modelling of music compositions created by the author. Goal-oriented compositions refer to musical compositions in which the performers have some goal, which they attempt to reach by following the instructions within the musical score. Three such compositions, and their general models, are presented in this paper. Models of these compositions were developed in MATLAB and simulations using these models allow for testing and experimentation during the composition process and analysis after a composition is complete.

1. Background

The history of networked multi-agent systems provide relevant context for the framing of these musical compositions and their computer simulations. Examples of "networked systems" include sensor networks, swarms, gene networks, social networks, synchronous networks of oscillators, networks of autonomous vehicles, and mobile ad-hoc wireless networks. This field is built on subjects from control theory, complex networks, graph theory [1] and distributed computing. [2]

The "goal-oriented" pieces presented in this paper can be seen as analogous to consensus problems [3] from the field of computer science and engineering. A consensus protocol/algorithm is a method used by networked agents (or dynamical systems) to reach a consensus (or in this case, goal). All of the systems dealt with in this case are discrete, in the sense that all of the states can be represented by integers. For an example of discrete consensus algorithms see Kashyap [4]. A related subject, flocking, is introduced in Saber [5]. For another musical work using networked multi-agent systems see Furlanete [6].

2. Discrete Symbolic Consensus Protocol

This section will detail a composition titled “5 Choose 4”, a percussion quartet originally performed by *Sō Percussion* [7]. The composition will be presented in its original form as well as in a generalized form allowing it to be adapted to other musical ensembles or mediums.

2.1 Original Form of “5 Choose 4”

This piece is scored for 4 percussionists and 5 percussion instruments situated in a planar, connected digraph as shown below (Fig. 2).

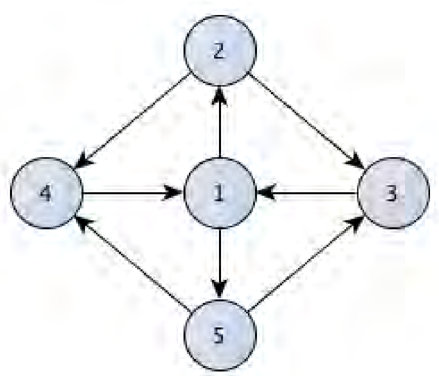


Fig. 2 Node Locations and Connections of “5 Choose 4”

Each instrument has a unique location at one of the 5 nodes. Additionally, each node has a corresponding page from the score with a list of four rhythms, each rhythm corresponding to one of the players. The rhythms (Fig. 3) are distributed throughout the network of nodes so that each player has each rhythm once at a unique location.

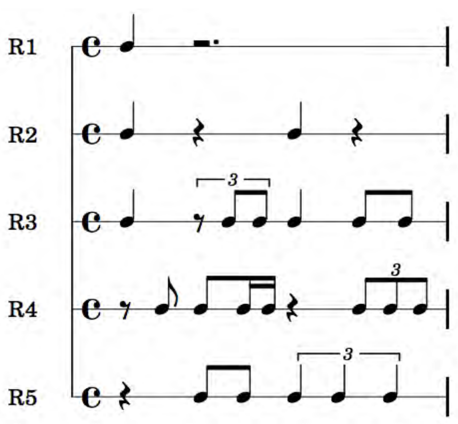


Fig. 3 Rhythms from “5 Choose 4”

The musicians begin the piece at node 1 and play their assigned rhythm in tempo with the other players. After playing the rhythm once, players may either repeat the rhythm or travel to a new node following the arrows on the map (Fig. 1). The travel time allotted to move between nodes is equal to the duration of one rhythmic phrase (one measure of music) and is hereafter referred to as one “time step”. The players may only repeat a rhythm a certain maximum number of times in a row, and must play it a certain minimum number of times. These maximum and minimum values are adjusted according to the skill level of the players and the desired difficulty of the performance. The piece ends once all of the players are playing the same rhythm in unison.

2.2 General form of “5 Choose 4”

This piece will now be presented as a symbolic consensus protocol for a multi-agent system. Each agent follows the methods outlined in the flow chart below (Fig. 4).

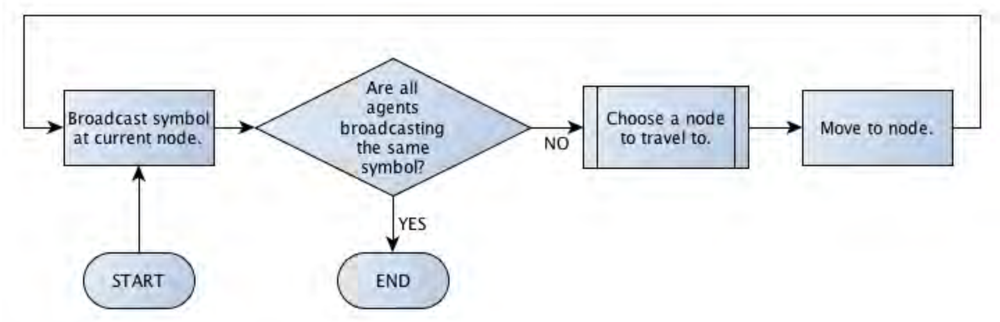


Fig. 4 Flow Chart of Symbolic Consensus Protocol

The agents carry out this methods on a planar, connected digraph with at least as many nodes as agents. Connectivity ensures that all nodes are reachable by each agent, making the digraph planar avoids collisions between mobile agents and having the minimum number of nodes be equal to the number of agents makes reaching consensus less trivial. However, it is feasible for this method to reach consensus on undirected or non-planar graphs, but not necessarily in every case.

At each node there is a symbol corresponding to each agent. In the most trivial case, all nodes have the same symbol for each agent and consensus is reached immediately. For each additional symbol added to the system there exists at least one more solution if and only if that symbol exists at at least one node per agent and there are less than or as many symbols as nodes.

Given n nodes and r agents, the amount of possible combinations is given by:

$$n^r \qquad (1)$$

In “5 Choose 4”, there are 5 nodes and 4 agents. That is 625 configurations, with only 5 of those resulting in consensus.

2.3 Computer Simulation of “5 Choose 4”

The simulation was conducted in MATLAB [8]. Four different methods for the sub-routine “Choose a node to travel to” from Fig. 3 were designed. While these methods strive to reach a consensus in the simulation, it is beyond the scope of this study to try and model all of the complex behaviour that occurs within the human performance of the piece. The simulated agents also have complete knowledge of the symbol locations, while human performers must discover these through exploration. The design and performance of these methods is detailed below.

2.3.1 Method 1, “Random”

This method simply chooses any neighboring node with equal probability.

2.3.2 Method 2, “Weighted Random”

This method weights the choice of any node by taking into consideration what symbols the other agents are broadcasting. Specifically the probability of selected a node is increased by 1/n % for each n agents broadcasting the same symbol as that node.

2.3.3 Method 3, “Altruistic”

The first priority of any agent using this method is to reach the node that has the most popular symbol. If there is no clear majority leader for symbols then the agent joins the symbol of their nearest neighbor.

2.3.4 Method 4, “Selfish/Lazy”

Any agent using this method will try to stay at the same node for as long as possible, regardless of which symbol is being broadcast. Once they reach the maximum number of repetitions they simply choose a neighboring node at random.

2.3.5 Performance of Methods

A simulation was run for each method independently as well as in combination. When ran independently all agents would use one method, when combined some agents would use one method and others agents another. The most useful results are summarized in the table below (Fig. 5).

Simulation of Symbolic Consensus Protocols for 1000 Trials	Min	Max	Med	Mean	% Consensus
M1 Random	5	1000	709.5000	637.8830	62.4000
M2 Weighted Random	3	1000	347	412.9000	93
M1 & M2	1000	1000	1000	1000	0
M3 Altruistic	1000	1000	1000	1000	0
M4 Selfish/Lazy	1000	1000	1000	1000	0
M3 & M4 50/50	7	545	56	75.7610	100
M3 & M4 25/75	8	1000	1000	917.7260	10
M3 & M4 75/25	5	981	69	100.3110	100

Fig. 5 Statistics of Simulations for Symbolic Consensus Protocol

The combination of the “altruistic” and “selfish/lazy” methods, with half of the agents using each method, reached consensus the fastest on average and also had the lowest maximum number of steps required to reach consensus. Combining M1 and M2 never resulted in a consensus, the agents just oscillated between two different combinations of nodes. Similar behavior was observed for M3 and M4 alone.

3. Discrete Position Consensus Protocol

This piece can be seen as a simplified version of the symbolic consensus protocol described above. As of writing this paper, this piece has not yet been premiered by human players. The simulation tools described below do allow for rudimentary sonification of the algorithmic work.

3.1 “Beau Noir Jamz” Original Form

This piece is designed to work with most combinations of brass instruments (trombone, trumpet, euphonium, horn and tuba). The total ensemble should be broken into groups of similar or mixed instruments, however horns should not mix with the other instruments. Each part gives the player a starting pitch and fingering/slide position (Fig. 6).

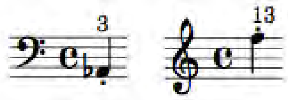


Fig. 6 Trombone and Trumpet Score Examples from “Beau Noir Jamz”

Within any sub-group of the ensemble, the goal of the players is to play the same pitch in unison. Players are allowed to depress/release one valve, or move one slide position per beat and may only repeat a given note once. Once all players are playing the same pitch, they move to the next line of the score, playing the given note. This occurs seven times before the piece is finished.

3.2 General Form of “Beau Noir Jamz”

This piece has the same general function as the flow chart for “5 Choose 4” (Fig. 4). Here the symbol being transmitted is the pitch of the note being played, and the nodes are the slide positions or valve combinations. Thus the trombonists can be thought of as agents, hereafter referred to as T-agents, traversing a line with seven discrete positions (Fig. 7).

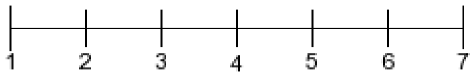


Fig. 7 Line With Seven Discrete Spaces

All of the valved brass instruments can be thought of as agents, hereafter referred to as V-agents, traversing a three-digit binary cube (Fig. 8).

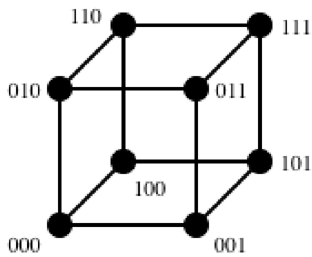


Fig. 8 Three-Digit Binary Cube

At its basic level, there is a simple mapping of the eight three-digit binary codes to the seven discrete points on the line.

Trombone Position to Valve Combination Mapping

Trombone Positions	7	6	5	4	3	2	1
Valve Combinations	111	101	011	110	100	010	000
Alt. Valve				001			

Fig. 9 Simple Mapping of T-agent Symbols to V-agent Symbols

If V-agents and T-agents are not mixed, then this becomes merely a consensus of position, rather than a consensus of symbols. That is, V-agents only need to match binary strings, and T-agents the single integer. For T-agents there are seven possible positions for each agent, so the total number of combinations between all agents is the same as equation 1, with n = 7 and r = the number of agents, with seven possible combinations resulting in consensus. The same is true for V-agents except that n = 8.

3.3 Computer Simulation of "Beau Noir Jamz"

In this simulation different trials were run for groups of four agents. T-agents and V-agents were both simulated alone. Since this interpretation of the score is framing the system as a position consensus, the T-agents and V-agents will not be combined in any trials. While this is possible in simulation, it is not anticipated to be practiced by human performers and therefore is not considered. The methods used will be described below followed by their performance.

3.3.1 T-agent Methods

M1 averages the current position of all of the other agents, and then rounds this number off to the nearest whole number. The agent then moves one step in the direction towards that value, or stays in the same position if it is the same value. The "random" method simply chooses to increase or decrease the current position with equal probability.

3.3.2 V-agents Methods

M1 calculates the average three-digit binary string of all the other agents. This string is compared to the current valve combination of the agent and if there are any differences, one bit is flipped accordingly. The "random" method chooses one of the three digits at random and then flips the bit.

Simulation of Position Consensus Protocols

	Min	Max	Med	Mean	% Consensus
M1 T-agents	1	10000	4	383.7360	96.2000
M2 T-agents	1	1763	196	286.1330	96.5000
M1 & M2 50/50 T-agents	1	133	15	20.7440	100
M1 & M2 75/25 T-agents	1	36	6	7.2810	100
M1 & M2 25/75 T-agents	1	580	59.5000	66.3900	100
Valve M1 & M2 V-agents	2	148	15	20.9270	100

Fig. 10 Statistics of Simulation for Position Consensus

For the T-agents the best performance came from three agents using M1 and one agent using the random decision making. The random decision making helps to break the agents out of any infinite oscillations between positions that may arise, lowering the maximum, medium and mean values for the trials. The V-agents had the best performance with the same combination of M1 and random choice for the same reasons as the T-agents.

4. Consensus of Discrete Time Delays

This section is about a piece “Cinque” originally orchestrated for a percussion trio with the initial performance done by TIGUE [9]. A description similar to the original score as well as the computer simulation will be described below.

4.1 Original Form of “Cinque”

In this piece players are allowed to choose which phrases they play from a given selection of music staves aligned vertically. The musical phrases are of different lengths, and once a player reaches the end of a chosen phrase they move on to the right reading the next phrase or selecting from another group of phrases. There are also “token phrases”, found within a box, which are played by all players. The goal of the players is to play these token phrases in unison.

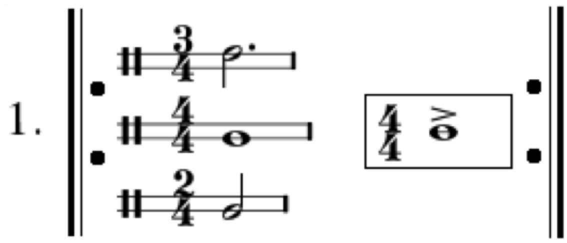


Fig. 11 First Movement of “Cinque”

The difficulty of the piece progresses as it moves through the five movements, adding token phrases, increasing the tempo and the number of different instruments employed. This culminates in the final movement which requires playing five different token phrases in unison consecutively.

Fig. 12 Fifth Movement of “Cinque”

4.2 General Form of “Cinque”

This basic behavior of one agent in this system is outlined in the flow chart below.

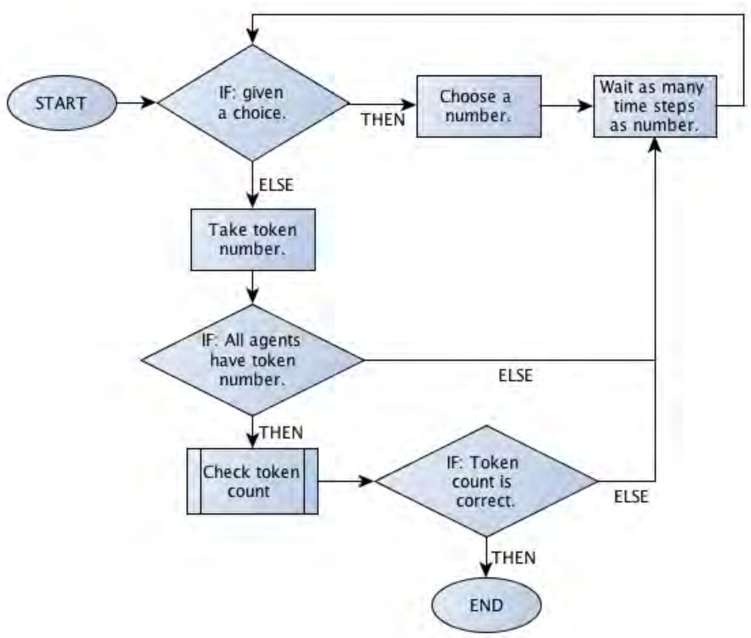


Fig. 13 Flow Chart of Discrete Time Delay Agent

This general model is adaptable to a variety of scenarios by explanation of the sub-routine “Check token count”. If the agents need to simply reach one token number then the checking the token count is trivial. However, if there are multiple token numbers within the system than the check token count subroutine must keep a running total of previous token counts, the routine is outlined below.

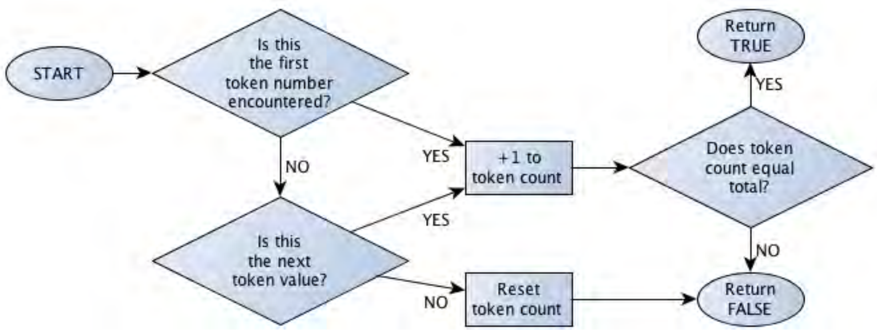


Fig. 14 Sub-routine “Check Token Count”

4.3 Simulation of "Cinque"

The simulation of "Cinque" proved difficult as the author did not to come up with any simple strategy for the agents to employ. Therefore each agent randomly choses any length phrase with equal probability. The results of this method for 10,000 trials for each movement are shown below.

Simulation of Cinque

	Min	Max	Med	Mean	% Consensus
Mov 1	1	4138	313	509.2800	83.7000
Mov 2	4	5153	645	925.4600	65.1000
Mov 3	8	10000	1.4785e+03	2.2347e+03	38.1000
Mov 4	5	2035	220	307.8450	96.7000
Mov 5	6	1266	108	157.8780	99.9000

Fig. 15 Statistics for Simulation of "Cinque"

5. Discussion

The simulation of these pieces is useful during the compositional process as it allows the composer to test ideas and experiment with different parameters of the piece. It may also give confidence that a consensus will be reached without having human performers test the piece. Additionally sonification and visualization (Fig. 1) of pieces is possible, again without the use of human performers.

Once these goal-oriented pieces are performed by humans there are many subtle differences and idiosyncrasies that are difficult to model. One of these differences observed during the performance of both "5 Choose 4" and "Cinque" is cheating. That is, doing things explicitly against the rules outlined in the piece. This is not considered to be detrimental to the performance of the piece, often times it is essential in ensuring that a piece will end in a timely manner.

Specifically, during the performance of "5 Choose 4", two behaviors were observed that aided in reaching a consensus, but these behaviors were not present in the simulated agents. The first of these was going the wrong direction from one node to another as seen on the map (Fig. 2). Most likely this was a mistake rather than conscious defiance of the rules described, however it was not an intended action by the composer. The second behavior that was observed was verbal communication between the performers. While this was not specifically outlawed, it does go against a mechanism of the piece being a team-based listening exercise. These human errors and adaptations enhanced the entertainment value and interest of the performance by giving it a much more realistic feel than the robotic computer simulations.

When “Cinque” was rehearsed the players adapted strategies to reach a consensus at any time of their choosing. By simply cueing on another they would all jump to a specific part of the score and play a pre-determined group of phrases so that they would align on the token phrase and play it in unison. Again, this is not explicitly against the rules and it does indeed help the performance as they can adapt the piece to fit into any time scale.

Cheating, in this sense, helped the players to reach a consensus more quickly. Indeed, the human players for both “5 Choose 4” and “Cinque” outperformed, on average, the best computer simulated methods. The subtle communication between ensemble members and trained musicians is very difficult to model accurately. This compounded with breaking rules allows humans to reach a consensus more quickly than the rudimentary agent protocols designed by the composer.

6. Conclusion and Future Work

The goal, from the composer's perspective, of these goal-oriented pieces, is to give the musicians a simple system or method to follow from which music emerges. Simulations of these pieces allow a composer to observe the multitude of realizations possible giving one an ability to analyze the collection of pieces rather than each piece individually. These models also serve as a proof of concept for a piece and an experimental playground for system design and composition.

It is the author's belief that the tools and general concepts presented here can be used in a variety of other mediums not limited to the arts. The strong connection of these works to engineering and the sciences allows for potential insight into real-world phenomenon that is perhaps unavailable through standard practices. As these ideas are further refined and developed it is hoped that they will better represent processes found in nature and technology.

7. References

1. D. Reinhard, “Graph Theory”, Heidelberg: Springer, 2002. Graduate Texts in Mathematics

2. R. O. Saber, ENGS 114: Networked Multi-Agent Systems (course description), Reza Olfati-Saber, Spring 2012

3. R. O. Saber, “Consensus and Cooperation in Networked Multi-Agent Systems”, Jan. 2007, Proceedings of the IEEE, Vol. 95, No. 1

4. A. Kashyap, “Quantized Consensus”, Nov. 2006, Preprint submitted to Automatica

5. R. O. Saber, “Flocking for Multi-Agent Dynamic Systems: Algorithms and Theory”, March 2006. IEEE Transactions on Automatic Control, Vol. 51, No. 3

6. Furlanete, F., Manzolli, J. Mase, K. “Musical Interaction with Multi-Agent Based Network Game”. IPSJ SIG Technical Report, pg 205-210, Japan, 2008

7. Sō Percussion is Eric Beach, Josh Quillen, Adam Sliwinski, and Jason Treuting. www.sopercussion.com

8. MATLAB version 7.13.0. Natick, Massachusetts: The MathWorks Inc., 2011.

9. TIGUE is Amy Garapic, Matt Evans and Carson Moody (USA percussion trio).