A GAME THEORETICAL MODEL FOR MUSICAL INTERACTION

Grace Leslie   *Navid Hassanpour*
Center for Research in Computing and the Arts; Music Department
University of California, San Diego

ABSTRACT

In this paper we analyze musical interaction using a game theoretical model and propose methods for generating music based on predefined interactive frameworks. We emphasize particular games that exhibit behaviors that can be exploited by musical means. In the case that a computer is involved in this interaction, examples of strategies are provided that can be implemented with finite state machines in order to model human decision-making behavior. Example systems which pit a user-performer against a computer in a game of strategy, and algorithmic agents against each other, demonstrate the set of behaviors that can emerge from the use of repeated games with learning methods such as reinforcement in interactive computer music.

1. INTRODUCTION

A live musical performance is a delicate set of interactions held between the composer, performers, and audience. Traditional Western music assumes a deterministic control by the composer over all interaction levels, whereas in relatively recent times, composers have developed systems that take the performer's choices into consideration, thereby placing them in a position of control. However, within this field of open notation there is a wide range of composer's approaches to the interaction between players, as seen when we compare for example, Morton Feldman's graphic scores with more deterministic interaction, to John Zorn's game piece Cobra, in which players decide on their own mixture of cooperation and competition. In the case of free improvisation, players often respond to each other's output based on their stored knowledge of their partner's playing habits.

In this paper, the above examples of interaction and learning are modeled using game theoretical tools. These models are then used to determine interactions between players in a completely instrumental setting, in addition to computer music applications that employ finite state machines to imitate different learning behaviors. The proposed framework is intended to be a natural model for musical interaction that can simulate musical learning and improvisation. If we consider the idea that the origins of our musical languages are derived from elementary musical interactions between players [1] it becomes apparent that a computerized process modeling human interaction with learning may be able to produce more natural musical

structures than other approaches to algorithmic composition which generate musical patterns based on graphical or mathematical processes.

Iannis Xenakis adopted game theoretical concepts in composing his pieces *Duel* and *Stratégie* [2], both for 2 orchestras, and *Linaia-agon* for brass trio [3]. He used two-player zero-sum games to model a predetermined musical interaction. This method was chosen to create a framework that would somewhat model competitive behavior [2]. However, the games he used were of a size that made it impossible for the players to strategize. In the case of *Linaia-agon*, musicians are known to have planned their moves before the concert in order to convey more fluid interactions [4]. His games are all zero-sum (which excludes the use of classic games such as Prisoners' Dilemma) and do not consider learning through repeated games, or any formalized approach to strategy.

We propose a compositional framework that uses repeated musical games with small sizes so that players, whether human or computer, can effectively strategize. When simulated with computers, different methods of learning can be implemented efficiently that model human decision making behaviors. When applied to instrumental music composition, this framework can provide a nuanced approach to cooperative and competitive behavior.

Section (2) will present examples of games that exhibit different behaviors that can be mapped to musical interactions, and repeated games that can be used for music composition. Section (3) will illustrate how various strategies can be implemented by finite state machines in the case of human-computer interaction. The final section will detail the application of different learning strategies in the case of the interaction between two algorithmic musical agents.

2. MUSICAL INTERACTION THROUGH REPEATED GAMES

In the following examples, we model human-human musical interaction with games involving strategy. For simplicity the descriptions and examples are restricted to two player cases, but the ideas are applicable to the games with more than two players, and more than two actions for each player. For now, each of the two players is involved in a game with S strategies for each of the players.

Let us first consider the simplest musical example of a repeated game structure. To demonstrate the abstraction mechanism in the modeling of musical interaction, one

can map specific examples of musical interactions to the outcomes of a game. The simplest way to demonstrate this mapping is to use a binary representation where 1 and 0 represent a "win" and a "lose" respectively. It is clear that there are four possible outcomes, whose audible effect would be chosen by the composer in order to encourage or discourage particular situations from being chosen by the performers.

For example, let's assume that two players are involved in a game where the first player is playing either piano or trombone and the second player is playing either flute or trumpet. We can summarize the details of the game in this simplified 2×2 game matrix shown in Table (1). The first number in the pair a, b, is the first player's (the row player's) payoff while the second number, b is the second player's (the column player's) payoff. This game

	Flute	Trumpet
Piano	$1, 1$	$0, 1$
Trombone	$1, 0$	$0, 0$

Table 1. A musical game

is arranged so that the players are encouraged to play piano and flute, respectively. When the players choose their quiet instruments, they each receive a payoff of one. When they both choose their loud instruments, they receive no payoff. However, a given player can also receive a payoff of one if he chooses to play his loud instrument when his opponent decides to play her quiet instrument. After several repetitions of this game, if learning is assumed, we can predict that the players will approach an equilibrium in which they always cooperate in order to receive the highest payoff. Thus the composer is able to control the evolution of the players' interactions without explicitly notating them. Interestingly, in [5] a similar musical game was proposed for the analysis of fairness in moral philosophy.

In the following example the two players are each given a set of pitches from which they are allowed to choose their next note. A new payoff matrix (see Table (2) below) is arranged such that the first player receives the highest payoff when they play a note and the other player is not playing or vice versa. The second player receives the highest payoff when they play at the same time as the first player or they both don't play. A tempo is specified by the composer, as is the direction that all notes are to be played on the beat. The end of the game is specified by a conductor or referee. The winner is chosen after a tally of solo notes and duo notes is taken. This repeated series of

	Note	Rest
Note	$0, 1$	$1, 0$
Rest	$1, 0$	$0, 1$

Table 2. Game matrix for the simplest human-human musical interaction example.

games invokes an unpredictable chain of mutual speculations on both sides of the game. The role of the composer is to compose appropriate music for each of the actions so that her desired result is achieved. In this case, the two players will be playing the same melody with various phase offsets, and interrupted by silences.

A composer may decide to write a piece in which there are clear patterns of cooperation and defection between the two players, in which case a classic game called the Prisoners' Dilemma could be chosen, an example matrix for which is shown in Table (3) below. This implementation will have one Nash Equilibrium at the $1, 1$ point, meaning that any independent deviation from this point by a player will result in a loss in his payoff.

$3, 3$	$0, 5$
$5, 0$	$1, 1$

Table 3. Prisoners' Dilemma

Finally it is possible to think of a fully symmetrical case such as the game in Table (4). This game has only one mixed Nash equilibrium. In this specific example it will be a mixture of $1, 2$ and $2, 1$. We will use Prisoners'

$1, 2$	$0, 0$
$0, 0$	$2, 1$

Table 4. Battle of the Sexes

Dilemma in the next section to demonstrate the application of finite state machines (FSMs) in the modeling of musical interaction. The game described in Table (4) will be used in section (4) to demonstrate the convergence behavior of learning strategies.

3. MUSICAL INTERACTION THROUGH FINITE STATE MACHINES

The game theoretical framework described in the previous section can be applied to musical pieces and systems involving interaction between human musicians and computers. The composer can specify certain musical behaviors from the human performer by tweaking the interactive system, rather that explicitly sending orders to the performer.

In this section we describe how this can be achieved with the use of finite state machines whose behavior is based on well-known game strategies.

3.1. Strategizing in Games by Finite State Machines

The computer playing a musical game with a human can implement either a learning algorithm or a fixed strategy based on a finite state machine. For example, a human player is asked to play a simple musical game (such as a Prisoners' Dilemma shown in Table (3)) with a computer, which is programmed to run a fixed strategy, such

as the one based on the FSM shown in Figure (1). For example, the response of this strategy to a chain of actions (C,C,D,D,C,D) from the human player will be the chain (C,C,C,D,D,C). In Figure (1) the computer player starts
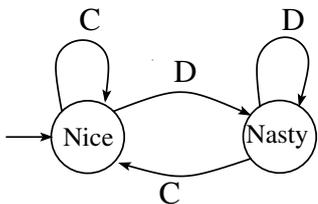
Figure 1. Tit for Tat

by being "Nice". When it is in the "Nice" state it plays action C (for cooperate), whereas if it is in the "Nasty" state it plays D (for defect). The labels of transition arrows are the acts of the opponent. For example in Figure (1) if the opponent plays D, the first player will switch from being "Nice" to being "Nasty."

Two finite state machines can dictate the strategy on both sides of the game. For example if the FSM in 1 plays Prisoners' Dilemma against the FSM in Figure (2),
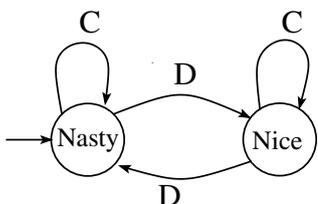
Figure 2. Tat for Tit

Assuming (X, Y) means the first player plays X and the second Y, the chain of plays will be

$$(C, D)(D, D)(D, C)(C, D)(D, D)(D, C)(C, D) \ldots \tag{1}$$

It will end up in a chain with a period of 3 and repetitions of $(D, D)(D, C)(C, D)$. There are 24 possible different two state machines. In this case, we have used only two examples of these FSMs. Further analysis of FSMs playing repeated games can be found in social and political science literature such as [6] and [7].

4. LEARNING IN REPEATED MUSICAL GAMES

As an alternative to fixing a strategy using a FSM, one can implement learning in repeated games. In such scenarios, players start by making each move with some probability and gradually learn from each other's plays. This setup can be considered as an abstract model for learning in the process of interactive musical play such as free improvisation. The composer can now choose a game matrix and strategy or learning method, i.e., reinforcement learning or best reply [8], in order to coax particular behaviors from the two performer-agents. We present a simplest example below employing the "Battle of the Sexes" game that demonstrates how different games and learning methods produce different musical outputs.

We use 4 notes to represent the 4 acts in a 2×2 game. $f\sharp$, $c\sharp$, c and g. In the game of Table (5), g and $c\sharp$ are

	g	$f\sharp$
c	1, 2	0, 0
$c\sharp$	0, 0	2, 1

Table 5. Example payoff matrix for the simplest computer-computer interaction.

played "loud" while $f\sharp$ and c are played "soft". Therefore, in the cases of desirable outcomes of (c, g), $(c\sharp, f\sharp)$, the payoffs are higher for louder pitches (2 compared to 1 for soft pitches). Later we will discuss the convergence behavior of reinforcement learning in this game. Several
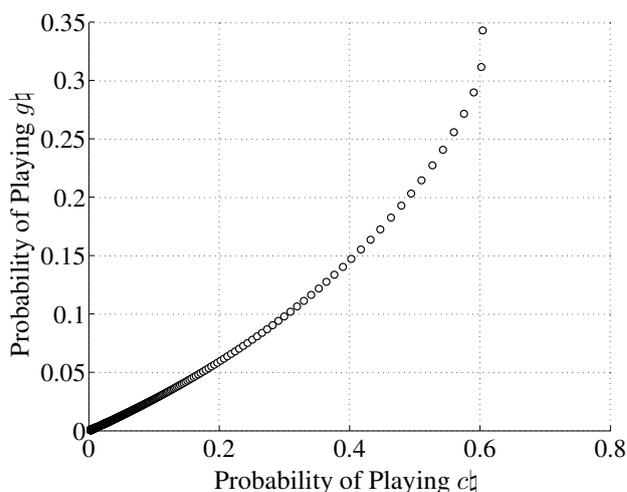
Figure 3. Convergence of Probabilities for Reinforcement Learning in the Game of Table (5)

learning mechanisms are possible. We will go through a few of these strategies, from the simplest to the most sophisticated. One of the simplest models of learning in a multi-player game is imitation, whereby strategies that are successful for one player are adopted by other players involved in the game. This learning strategy recalls one of the most fundamental properties of both improvised and through composed music, both of which rely on imitation to structure interaction between voices and musical ideas.

4.1. Reinforcement Learning

In this learning model players reinforce their acts based on the payoffs they get from each play. Each player starts with some initial probability of playing different acts and after each round of play, both players update their probabilities of playing the acts based on the payoffs they received in the previous round.

More concretely, assume that players $1, \ldots, K$ are involved in a game with positive payoffs and S strategies. Each player k starts with initial probabilities $p_{k,1}(0), \ldots, p_{k,S}(0)$ of playing strategies $1, \ldots, S$. After each play, the players participating in the game update their probabilities of playing each of the S strategies. Assume that

each player k has an urn with $A_{k,s}(n)$ balls for the act s at time n. We restrict our analysis to two player games, $K = 2$, for the ease of presentation, but results apply to general K player games. Taking the payoff earned when the acts s and j are played by player k, $k = 1, 2$, at time $n = 1, \ldots, N$ as $\pi_{s,j}^k$,

$$A_{k,s}(n+1) = A_{k,s}(n) + \delta\pi_{s,j}^k \qquad (2)$$

where δ is the reinforcement step size. Note that each urn, $k = 1, 2$, has an initial number of $A_{k,s}(0)$ s balls. At each time $n = 1, \ldots, N$ the probability of playing s at time n for player $k = 1, 2$ is

$$P_{k,s}(n) = \frac{A_{k,s}(n)}{\sum_{r=1}^{S} A_{k,r}(n)}. \qquad (3)$$

For example if the players end up playing $(c\natural, g\natural)$ in Table (5), the row player will add 1 reinforcement unit δ to his urn for the act $c\natural$ while the column player adds two reinforcement units 2δ to her urn for the act $g\natural$.

We have implemented this learning mechanism using the musical game in Table (5) and the convergence behavior for one learning instance is depicted in Figure (3). This plot shows that the probabilities of playing $c\natural$ and $g\natural$ go to zero asymptotically, i.e. players learn to play $(c\natural, f\natural)$ eventually. It turns out that for the game in Table (5) $(c\natural, f\natural)$ is the outcome of reinforcement learning half of the time, while in the remaining half $(c\natural, g\natural)$ is the result of learning. Reinforcement learning results in convergence of the probabilities to the unique mixed Nash equilibrium of the game. Other games have different behavior; for example, reinforcement learning in Prisoners' Dilemma makes probabilities converge to the Nash equilibrium, the $(1, 1)$ strategy, almost surely [9].

4.2. Best Reply

This is based on the history of the played games. Players can choose the acts that yield the highest payoff based on the games played in the m last plays.

In the above definition assume that each of K players, k, has a history of playing $(s_k(1), s_k(2), \ldots, s_k(n))$ up to time n. Each of the players decides about his act at time $n+1$ based on the empirical frequencies of his opponents' plays $s_{-k}(1), \ldots, s_{-k}(n)$. The expected utility of each act against this distribution is calculated. The largest expected payoff determines the act at $n + 1$.

For example, assume that in the prisoners' dilemma in Table (3), the second player has played (C,D,D,C,C) during the first five plays. Assuming this history, by playing C, the first player would get an expected payoff of $3 \times 3/5$ and with playing D, $(3 \times 5 + 2 \times 1)/5$. Therefore she will play D. In general, we set

$$s_k(n+1) = \arg\max_s \sum_{i=1}^{n} \pi_{s,j_{-k}}^k(i) \qquad (4)$$

The results of the implementation of this learning strategy converge faster than when reinforcement learning is used.

Other learning mechanisms such as Bayesian learning are applicable to musical games. In this case, the players assume that their opponents play based on a probability distribution and update that distribution after each play based on the acts played. Based on these updated expected distributions, they choose acts that maximize the expected payoffs.

Our list of learning methods is by no means exhaustive. However, our aim is to suggest that musically meaningful behaviors can be simulated using some of these methods.

5. CONCLUSION

This indeterminate approach opens up the musical process to allow for performer participation, while still lending the overall stylization of the interaction between these performers to the composer. Thus the music presents a delicate balance between the uncertainty of the human's behavior, and a game situation which is designed to coax particular behaviors out of the interaction between these players. The framework we have suggested demonstrates a variety of human emotions and projections in the format of a stylized, repeated game. A wide variety of options are available to the composer including exploration of different payoff matrices, finite state machines, and various learning algorithms.

6. REFERENCES

[1] Miranda, Eduardo. *Composing Music with Computers*, Focal Press, Oxford: 2002.

[2] Xenakis, Iannis. *Formalized Music.* Revised and Enlarged Edition, Pendragon Press, New York, 1992.

[3] Xenakis, Iannis. *Linaia-agon.* Editions Salabert, Paris: 1972.

[4] Bluchin, Benny. "Linaia-agon: Towards an Interpretation Based on the Theory." *Proc. of the International Symposium Iannis Xenakis*, Athens, May 2005.

[5] Braithwaite, R. B. *Theory of Games as a Tool for the Moral Philosopher.* Cambridge University Press, New York: 1955.

[6] Axelrod, Robert. *The Evolution of Cooperation.* New York: Basic Books, 1984.

[7] Binmore, Ken. *Game Theory and Social contract* MIT Press, Cambridge: 1994.

[8] Fudenberg, D. and Levine, D. K. *The Theory of Learning in Games.* The MIT Press, Cambridge: 1999.

[9] Beggs, A. W. "On the Convergence of Reinforcement Learning." *Journal of Economic Theory* 122(1): 2005.