

# earGram Actors: An Interactive Audiovisual System Based on Social Behavior

## **Peter Beyls**

CITAR, UCP,  
Porto,  
Portugal

-----  
pbeyls@porto.ucp.pt  
-----

## **Gilberto Bernardes**

SMC group, INESC TEC,  
Porto,  
Portugal

-----  
gba@inesctec.pt  
-----

## **Marcelo Caetano**

SMC group, INESC TEC,  
Porto,  
Portugal

-----  
mcaetano@inesctec.pt  
-----

## **ABSTRACT**

In multi-agent systems, local interactions among system components following relatively simple rules often result in complex overall systemic behavior. Complex behavioral and morphological patterns have been used to generate and organize audiovisual systems with artistic purposes. In this work, we propose to use the Actor model of social interactions to drive a concatenative synthesis engine called earGram in real time. The Actor model was originally developed to explore the emergence of complex visual patterns. In turn, earGram was originally developed to facilitate the creative exploration of concatenative sound synthesis. The integrated audiovisual system allows a human performer to interact with the system dynamics while receiving visual and auditory feedback. The interaction happens indirectly by disturbing the rules governing the social relationships amongst the actors, which results in a wide range of dynamic spatiotemporal patterns. A user-performer thus improvises within the behavioral scope of the system while evaluating the apparent connections between parameter values and actual complexity of the system output.

## **KEYWORDS**

Audiovisual system; Distributed agent system; Emergence; Complexity; Concatenative sound synthesis; Interactive musical improvisation.

## **1 | INTRODUCTION**

Natural systems such as insect swarms, the immune system, neural networks, and even chemical reactions (Bak, 1995; Kauffman, 1995; Camazine, 2003) are widely considered to exhibit complex behavior arising from multiple local interactions among agents following simple rules. The self-organizing behavior of social animals (Reynolds, 1987) has been used to explain certain social interactions, including those in human society (Ulanowicz, 1979). Interestingly, the emergence of complex behavior in computer simulations of natural systems has been explored aesthetically in artistic settings such as dance (Tidemann, 2007), audiovisual installations (Beyls, 2012), sound and music (Miranda, 1994; Blackwell, 2002; Caetano, 2007), and sculpture (Todd, 1992), among others.

In contrast to top-down design in most cultural artifacts, natural systems exhibit patterns arising from multiple local interactions among individuals or entities that do not exhibit the patterns themselves. From the stripes of zebras to snowflakes and termite mounds, pattern at the global level emerges solely from interactions among lower-level components (Camazine, 2003). Much research in the discipline of artificial life investigates the emergence of life-like forms of synthetic biology (Langton, 1997). Recent work in artificial chemistry (Dittrich et al., 2001) offers

a wealth of models for constructing emergent behavior. For example, the idea of molecular interaction may successfully underpin complex musical human-machine interaction (Beys, 2005). Various music systems were built exploiting swarming behavior (Blackwell and Bentley, 2002) – a model first formalized in the original flocking algorithm (Reynolds, 1987). Miranda (1994), in turn, proposes to use the patterns that emerge from cellular automata in music composition. Caetano (2007) exploits the self-organizing dynamics of different algorithms inspired by biological systems to obtain trajectories that drive sound transformations.

In this work, we propose to use the complex behavior that emerges from a multi-agent system called the Actor model to drive earGram, a concatenative sound synthesis engine, in real time. The Actor model of social interactions uses the concepts of affinity and sensitivity to iteratively displace the agents, called actors, to different settings of social stress. The self-organizing nature of the Actor model results in intricate visual trajectories followed by the actors. These trajectories, in turn, are used as input to earGram. EarGram organizes a collection of sounds in the plane according to their intrinsic perceptual qualities, such that neighboring sounds are more similar than sounds that are far apart. Therefore, spatial trajectories result in sonic trajectories that become gradual transformations along the perceptual dimensions used to organize the sounds. The user can choose the sound features corresponding to the dimensions of the space, which results in different configurations of the sounds in the plane. Consequently, the same trajectory can have several different sonic results.

Our goal is to build a system supporting non-trivial rewarding human-machine interaction. In contrast to conventional linear mapping, the user interacts with the Actor model indirectly by changing the affinity and sensitivity values, which results in different dynamic configurations. The system dynamics becomes the organizational paradigm followed when exploring the conceptual space of sonic results. The actors behave autonomously from the specification of simple local instructions, yet the system is open to disturbance by an external human performer (HP), offering fascinating aesthetic potential for human-machine interaction. Then, a perception of life-like qualities becomes

apparent, one interacts with a quasi-unpredictable system while the structural integrity of that system remains. Such a work suggests critical consideration of the notions of interactivity, intricacy, participation and unpredictability.

This paper is further structured as follows, firstly we explain the Actor model and its behavioral scope, then we address concatenative sound synthesis in earGram. The implementation of a functional bridge between both components is presented. We proceed to discuss the technical aspects of the system implementation, followed by aesthetic considerations and user interaction. Finally, we discuss the conclusions and future work.

## 2 | MODELS OF SOCIAL INTERACTION

Linear top-down planning and design suffer from a knowledge acquisition bottleneck. In contrast, collective behavior commonly presents self-organizing properties whereby pattern at the global level emerges solely from interactions among lower-level components. Remarkably, even very complex structures result from the iteration of surprisingly simple behavior performed by individuals relying only on local information. Social interactions are a typical example of self-organizing behavior that leads to the emergence of complexity.

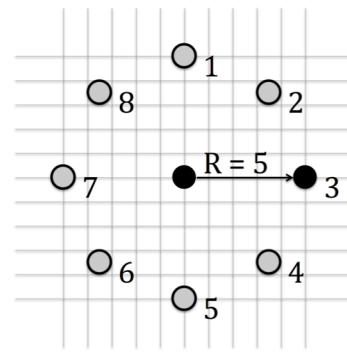
### 2.1 THE PARTY PLANNER MODEL

Our implementation is inspired by the Party Planner Model (PPM), developed by Rich Gold and documented in his seminal book *The Plenitude* (Gold, 2007). Imagine a party where each individual aims to be physically close to people one likes and as far away as possible from people one dislikes. An individual's level of unhappiness is the perceived social stress impinging at a particular location in physical space. Formally, given  $N$  individuals, the level of unhappiness of individual  $i$  is expressed in Equation 1 as the sum  $S(i)$  of absolute values of the differences in ideal distance  $d(i, n)$  minus the actual distance  $\delta(i, n)$  between individuals  $i$  and  $n$ . An individual does not express any social opinion towards oneself, thus  $N - 1$  evaluations take place.

$$S(i) = \sum_{n=1}^{N-1} |d(i, n) - \delta(i, n)| \quad (1)$$

|       |             |             |             |             |
|-------|-------------|-------------|-------------|-------------|
|       | $A_1$       | $A_2$       | $A_n$       | $A_N$       |
| $A_1$ | 0           | $a_{(1,2)}$ | $a_{(1,n)}$ | $a_{(1,N)}$ |
| $A_2$ | $a_{(2,1)}$ | 0           | $a_{(2,n)}$ | $a_{(2,N)}$ |
| $A_n$ | $a_{(n,1)}$ | $a_{(n,2)}$ | 0           | $a_{(n,N)}$ |
| $A_N$ | $a_{(N,1)}$ | $a_{(N,2)}$ | $a_{(N,n)}$ | 0           |

a) Representation of the affinity matrix.



b) Representation of moving actor.

Figure 1 | Affinity matrix and moving actor.

Every person aims to minimize his/her level of unhappiness by moving in space to a neighboring spatial location, a few steps away from the current location, potentially offering less social stress. As a result, a person will relocate to his/her ideal distance from every other person thus minimizing the total perceived level of unhappiness.

## 2.2 THE ACTOR MODEL

An extended version of the PPM called Actors has been used to simulate collective musical improvisation (Beys, 2010). One may think of the Actor model as a complex dynamical system that, according to the specification of particular social preferences, will produce spatiotemporal patterns of considerable intricacy. Similarly to the PPM, the Actor model aims to minimize social stress of a society of actors with certain predefined degrees of affinity towards one another. The dynamic scope of the system is conditioned by two parameters, affinity and sensitivity. The affinity-matrix shown in Figure 1-a specifies the affinity values  $a_{(i,n)}$  between actors  $A_i$  and  $A_n$  and the sensitivity parameter specifies the distance threshold to apply the affinities. The sensitivities are unique to every actor and unilateral, conditioning the interaction to any neighbor within range independently from the neighbor’s own sensitivity value.

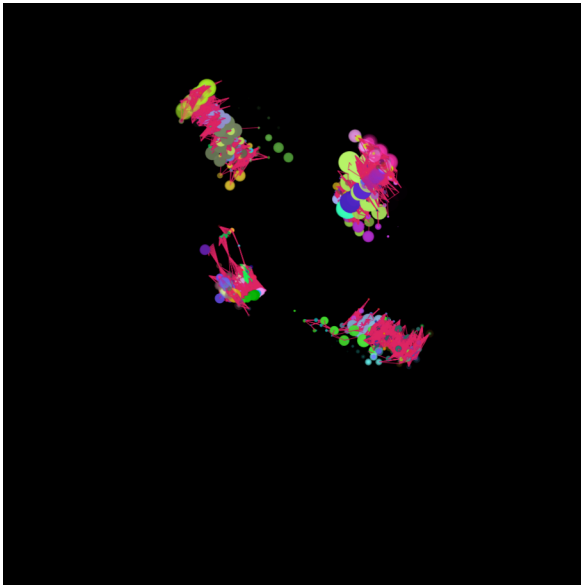
At each iteration, the affinity-matrix is consulted to compute a list of individual social tensions from the observation of the grand sums of impinging stress calculated using Equation 1. Each actor will be relocated to a region of lower social stress from eight potential locations at a 5-pixel radial distance from the position of the actor considered, as depicted in Figure 1-b. Only actors whose distance is within the

sensitivity range are considered neighbors of the perceiving actor. All actors proceed according to the same logic. However, actions by individual actors only observe local social concerns i.e. the evaluation of stress towards the closest neighbors. As a result, the process evolves as an animated sequence of globally complex spatial configurations. In addition, conflicting requirements may contribute to highly non-linear behavior. For example, actor  $A_1$  may prefer to be close to actor  $A_2$  while actor  $A_2$  aims to be far away from actor  $A_1$ . Merging this local concern with impact from neighboring actors, complex following or push-pull oscillatory behavior might emerge.

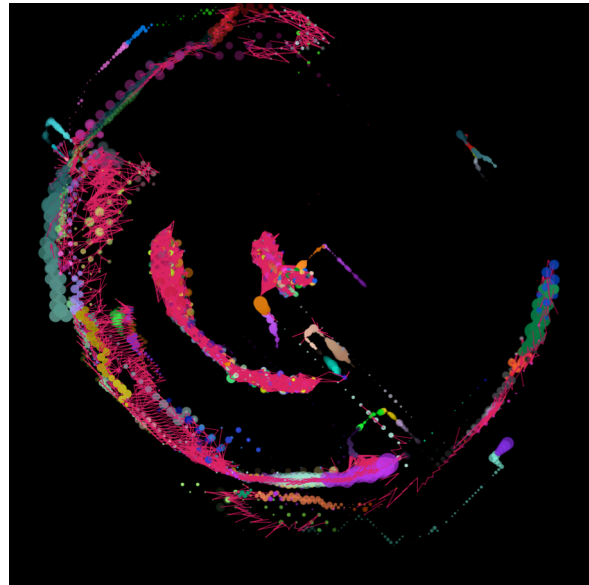
Intuitively, we can explain the connection between the complexity of the resulting spatiotemporal behavior and the range and diversity of values in the matrices. For example, given equal values in the affinities-matrix, all actors will relocate to be at equal distances. A wide range of spatiotemporal phenomena is generated from the specification of individual matrix values. Such a control structure is aesthetically attractive because the HP has the impression of interacting with an intricate system whose behavior is only partially understood. The causal link between matrix and behavior is non-trivial, however it is perfectly coherent and offers structural integrity. Although individual actor behavior is unpredictable, the system nevertheless offers a strong overall impression of coherent performance.

## 2.3 INTERFERENCE OF HUMAN PERFORMER IN SOCIAL INTERACTION

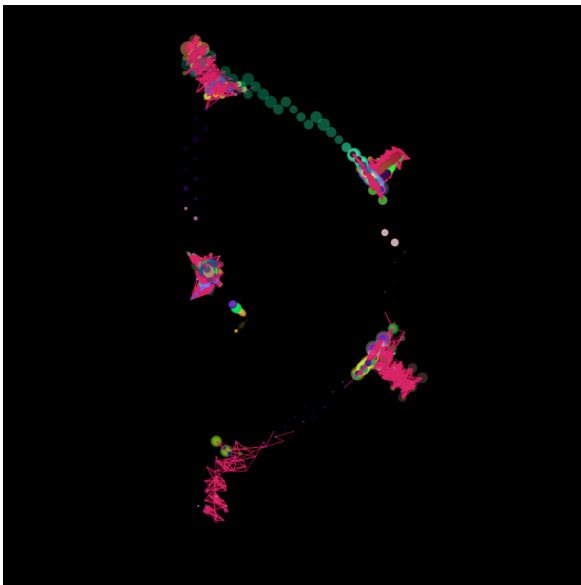
In the Actor model, the user influences the outcome of an otherwise self-organizing social system instead of directly controlling the system. Mapping commonly



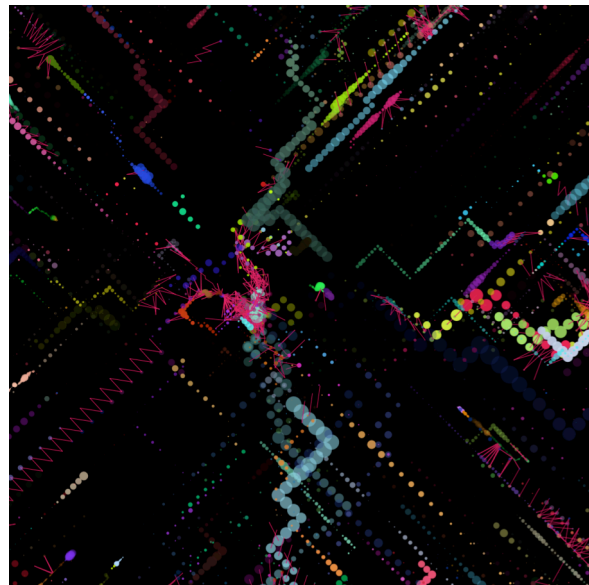
a) Local oscillations



b) Circular Movement



c) Basins of attraction



d) Radial configuration

Figure 2 | Spatial configurations resulting from different settings for the Actors model.

aims to create specific functional relationships between user input and system responses. Conventional approaches to mapping are deterministic, usually yielding predictable results. Conventional mapping typically generates responses selected from a user-designed palette of options. For example, user swipe gestures used to turn the pages of an e-book. Given an aesthetic orientation favoring unpredictability and surprise, the concept of deterministic mapping is problematic. The Actor model suggests an alternative; the HP interferes with the parameters affecting system dynamics rather than directly determining the output - not unlike Sal Mariano commenting on himself playing the SALMAR

Construction: “it was like driving a bus” (Chadabe, 1997).

In earGram Actors, the HP can interfere with the system’s innate behavior in two possible ways. First, a HP virtually present interacts with the actors who, in turn, acknowledge the HP’s social preferences. The HP interacts with the system via a control interface (e.g., Microsoft’s Kinect or Nintendo’s Wii) that causes actions in the space. The system is influenced only locally but might entail the emergence of complex patterns. The other possibility is to have the HP conceptually outside the actor society but able to adjust global parameter settings. So the interaction happens via the parameter settings of the system.

The current implementation, reported here, documents the second method.

Figure 2 displays a collection of 4 snapshots, momentary spatial configurations captured in a continuous animated process. The window size is 1000 by 1000 pixels. Social affinities are set at random in the range of 50 to 500 pixels, whereas actor sensitivity is fixed (at a radius  $R = 5$  pixels) in this experiment. Each image displays the configuration after 100 iterations (typically more), illustrating how different complex spatial configurations might emerge from different affinities. Only the affinities-matrix is occasionally slightly modified while the process is running. In Figure 2-a, all actors coalesce into four specific locally oscillating configurations. The effect of the forces of attraction and repulsion merges into a stable spatial pattern. Circular movement is seen in Figure 2-b with parallel trajectories showing evidence of attraction and repulsion cancelling out. Five major islands of activity emerge in Figure 2-c while a spatial explosion occurs in Figure 2-d.

We explore the behavioral scope of the system through interactive modification of the affinities-matrix. As the actors interact their trajectories oscillate between quasi-periodic and irregular. We end up having a control structure of high plasticity that morphologically blends the spatiotemporal complexity in the sound application. The sound synthesis module receives control data from the Actors via Open Sound Control (Schmeder et al., 2010).

### 3 | MUSICAL COMPONENT OF THE SYSTEM

In this section, we briefly introduce concatenative sound synthesis and earGram, the application used in this work. The discussion covers how the sound features capture perceptual qualities of the sounds, how to create sound spaces using the features as dimensions, and how different spatial configurations of sounds result from different dimensions.

#### 3.1 CONCATENATIVE SOUND SYNTHESIS

Concatenative sound synthesis (CSS) is a sample-based technique that creates “musical streams by selecting and concatenating source segments from a large audio database using methods from music information retrieval” (Casey, 2009). Technically, CSS is a derivation of concatenative text-to-speech

synthesis, which breaks speech recordings down into elementary units and then rearranges these units to match words from text. In turn, CSS allows the manipulation of heterogeneous corpora of sounds other than speech. Historically, CSS can be grouped with other sample-based techniques such as micromontage and granular synthesis which originated from the early *musique concrète* experiments and, to a certain extent, it can be understood as being an extension of micromontage and granular synthesis towards a higher degree of automation. While its conceptual and first software implementations were presented in 2000 (Schwarz, 2000), CSS only began to find its way into musical composition and performance in 2004, in particular through the work of Sturm (2004, 2006) and Schwarz (Schwarz et al., 2007).

Briefly, what's unique about CSS in relation to other sample-based techniques is the annotation layer of the segments database, which not only provides the user with a good description of the audio source content, but also allows him/her to adjust, organize and re-synthesize the temporal dimension of the source in refined ways. Segment annotations include features automatically extracted and grouped into a single vector with the help of low-level audio descriptors, in a similar fashion as the audio-annotation layer of the MPEG-7 standard (Kim et al., 2005).

CSS shows great potential for high-level instrument synthesis, resynthesis of audio, interactive exploration of large databases of audio samples, and procedural audio, especially in the context of interactive applications such as video games. Despite its mature development at the engineering and technological levels, CSS is rather undeveloped in aesthetic and utilitarian terms. In addition, even though most research in CSS is oriented toward music, the technique still lacks substantial contributions in terms of creative output.

#### 3.2 EARGRAM

EarGram (Bernardes, 2013, 2014) is an open-source and freely available application created in Pure Data for the real-time creative exploration of CSS [1]. EarGram extends CSS with new possibilities for generative audio by adopting strategies from both

algorithmic-assisted composition and music information retrieval (MIR). The latter is responsible for segmenting an audio stream into elementary units, describing the most relevant features of the segments, and extracting patterns from the resulting collection of segments. Additionally, the system unpacks MIR terminology and concepts to a more adapted usability for musicians by relying on musicological and psychoacoustic theories, and presents most processing stages of the system in an intuitive manner, mainly through visualizations. The set of MIR tools adopted in earGram constitutes a valuable aid for decision-making during performance by revealing musical patterns and temporal organizations of the database, which are then used to represent audio in common algorithmic-assisted composition techniques.

EarGram includes four generative modes: spaceMap, soundscapeMap, infiniteMode, and shuffMeter, which cover a wide range of musical applications, such as the automatic generation of soundscapes, remixes, and mashups, to cite a few. Of interest here is the spaceMap mode, which is used to interact with the Actor model adding a sonic layer that offers musical functionality.

SpaceMap synthesizes sound textures with variable density driven by spatial trajectories defined by navigating in a 2D-plot representation of the corpus. It can be seen as a granular synthesizer with an extra layer of control over the acoustic results, driven by and depending upon the audio descriptors used to represent the audio segments. The resulting descriptor space can organize the audio segments into spatial configurations, defining neighborhood relations and relative distances. For example, two segments might present similar loudness values at very different pitches, which would place them close together along the loudness dimension but far apart along the pitch dimension.

The interface of spaceMap is shown in Figure 3 as a plane whose axes can be assigned to single audio descriptors or linear combinations of them. For example, the vertical axis might be loudness and the horizontal axis might be pitch. Each sound segment is represented by a (square) point in space, and their spatial organization is defined by their sound qualities (as measured by the descriptors). The visual

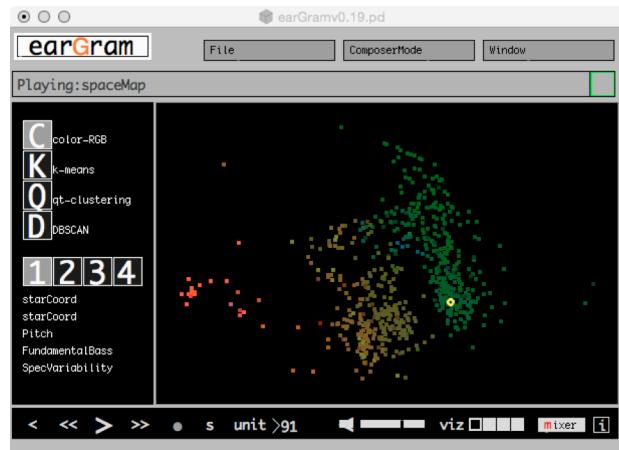


Figure 3 | 2D plot of the speech sound database used in the musical component of the system.

representation of the database is used to play sound segments in the descriptor space as spatial trajectories. Hovering the mouse pointer (round point) plays the sound that is closest in the space. So, in the example, sliding the pointer vertically upward would play sounds that are louder and horizontally to the right would play sounds higher in pitch. Diagonal upward right-hand movement would play sounds with increasing pitch and loudness. While small movements synthesize similar sounding segments, larger movements pick sounds with greater sonic differences. SpaceMap allows the creation of sonic textures by the user with highly controllable nuances.

## 4 | TOWARDS A UNIFIED AUDIOVISUAL SYSTEM

### 4.1 SPEECH SOUNDS AS A SOCIAL METAPHOR

We selected a database of multi-linguistic speech sounds as a metaphor of social interaction for the integration of earGram with the Actor model. Our aim was to represent social interaction, and particularly the affinity among individuals, by the perceptual proximity of speech sounds. Whenever the “society” of Actors reaches a stable configuration, the sonic response of the system should reduce the amount of variation to a minimum while highly unstable configurations should result in a high level of sonic variation. Between these two poles, there is a continuous degree of variation.

The speech sounds were retrieved from the UCLA Phonetics Lab Archive [2], which includes both native female and male speakers of different languages, such as Bulgarian, Dutch, Estonian, Javanese, Nepali, Portuguese, Zulu, among others. After some basic sound editing to improve the sound file quality,

including filters, equalization and noise removal, earGram automatically segmented the collection of speech sounds into short snippets of 200 ms each. Then, we solved the most crucial pre-processing stage of our database creation: the selection of a set of audio descriptors to represent our segments in the system.

The analysis of the database segments comprised two main tasks. First, we manually restricted the set of available audio descriptors to a sub-set of audio features that included: noisiness, pitch, brightness, spectral width, and sensory dissonance. Then, we weighted the set of selected audio descriptors to adjust their contribution in the feature space. Weights were automatically assigned according to the computed variance of each of the selected descriptors, assuming that features with higher variance enhance the similarity computation, because they provide a more distinctive characterization of sound objects. By reducing the number of audio descriptors and weighting their contribution, we not only discard redundant information from the analysis of speech segments, but also enhance the computation of their perceptual similarity, which consequently improves their visual representation on the interface.

The number of features used dictates the dimensionality of the sound space where the segments are organized. We created a 2-D feature space to overlay the representation of the sounds onto the 2-D visual representation of the Actor model. The original feature vector was reduced to two dimensions using the algorithm star coordinates, first proposed by Kandogan (2000) and used in the scope of CSS by Bernardes et al. (2014). The resulting feature space can be seen in Figure 3, which shows a 2-D plot visualization of the database in a 2-D space whose axes are a linear combination of the aforementioned audio descriptors.

#### 4.2 INTEGRATING THE SYSTEMS

After the database creation, we tackled the mapping between the Actor model and earGram's spaceMap, i.e. the visual and musical components of our system. In spaceMap, synthesis is typically controlled by defining trajectories in earGram's interface with the mouse pointer. EarGram then retrieves the closest

unit to the mouse position and plays the selected segment with a Gaussian amplitude envelope. In our work, we replaced the mouse control by the position of each Actor in the space, given by its X and Y coordinates. This rather simple mapping strategy is effective in the sense that the segments plotted in the spaceMap interface are organized according to their perceptual distance. Therefore, actors with high affinity values are close together in the descriptor space, so they will trigger similar sounding units, resulting in affinity being related to perceptual similarity. A video with an example of the final system is available at [http://mat.inescporto.pt/?page\\_id=312](http://mat.inescporto.pt/?page_id=312). In this example, there are 50 actors, the affinity matrix was initialized with low values, and the sensitivities are initialized all at 5 pixels.

Initially, the large amount of data sent to earGram over the network via the OSC protocol resulted in both technical and aesthetic problems. The video frame rate (the same rate at which the location of the actors is computed) was too high to be sent over the network and to be synthesized by earGram in real-time. Technically, the network failed to transmit all the data which affected the sonic result aesthetically mostly due to saturation. To address this issue, we set the video frame rate to 25 frames per second and implemented a clock that controls the rate of data sent over the network. The location of all actors is stored in a memory buffer and sequentially read every 800 ms. Therefore, we hear a new segment every  $800/N$  ms, where  $N$  is the total number of Actors.

Finally, we added an extra processing layer to the musical counterpart of the system to reinforce the relationship between the overall activity of the Actors in the space and its auditory feedback. The overall activity is estimated as the sum of piecewise displacements across frames. In turn, the displacement is computed as the Euclidean distance between the current location and the previous one. The total displacement of the Actors controls the wet-dry parameter of a spectral freeze audio effect [3] in earGram, where small displacements result in low spectral change via spectral smoothing.

#### 5 | TECHNICAL DISCUSSION

A user engages with the proposed system mainly via modification of the parameters (global affinity matrix

and local sensitivity values) in the Actor model, exerting influence on the dynamic behavior of the system. This indirect method for influencing the system behavior has implications on both the visual and sonic components of the system along two conceptual dimensions, level and extent of activity. The level of activity is related to the displacement of the actors, ranging from stationary to dynamic. The extent of activity refers to the distribution of the actors in the plane, which can vary between concentrated and spread out.

However, other decisions also influence the sonic outcome, such as selection and pre-processing of the sound source material, selection of the features used to define the dimensions of the space in earGram, and the affinities and sensitivities for the Actor model. In general terms, the source material determines the range of sonic possibilities. Speech sounds will produce a different outcome than instrumental, environmental, or synthetic sounds. The features have a direct impact on the distribution of the sound segments in the plane in earGram. Changing the features will reorganize the same sounds according to different perceptual similarities, such that the same trajectory will generate a different sonic outcome. In this section, we will discuss the impact that each decision has in the aesthetic result.

In general terms, the spatiotemporal behavior of the system is determined by the level of social stress, which, in turn, depends on the magnitude and homogeneity of the affinities and sensitivities. High affinities result in strong attraction between actors, while low affinities generate repulsive forces. Homogeneity in the affinity matrix also impacts the global dynamic behavior. The level of activity decreases whenever the user sets all the affinities to the same value, resulting in point attractor behavior. Heterogeneous affinity values entail complex dynamic behavior. The sensitivity also plays an important role in the dynamic behavior of the actors because it determines the radius  $R$  of influence of the affinity values  $a_{(i,n)}$  between actors  $A_i$  and  $A_n$  (see Figure 1-a). High sensitivities force the actors to consider distant neighbors, while low sensitivities cause the actors to only interact with nearby neighbors.

For example, high magnitude homogeneous affinity values with high sensitivity will likely result in all actors

clustered in a point because they are all highly attracted to one another. Low magnitude homogeneous affinity with low sensitivity will likely result in a uniformly spread out configuration across the plane because all actors are equally repulsed by their nearby neighbors. Notice that both scenarios result in low levels of activity because the examples suppose a homogeneous affinity matrix and nearly equal sensitivity values. Highly dynamic complex behavior is commonly achieved through heterogeneous affinities and sensitivities.

The sonic response depends on the level and extent of activity as well. On the one hand, the level of activity is responsible for the dynamic response of the system. Each spatial trajectory results in a sonic trajectory that translates as temporal variation of the corresponding sound texture. On the other hand, the extent of activity influences the diversity of the sonic response by exploring different regions of the sound space.

The level of social stress drives the visual and sonic components of the system in symbiosis. The more complex and chaotic the oscillatory behavior of the actors, the more heterogeneous the sonic response. Stable configurations result in sound textures with little variation. In other words, the actors' dispersion is related to the variability or "spreadness" of the sound segments selection, which equate with the level of coherence of the resulting texture due to the organization of the segments on earGram's feature space. In between the two poles a wide and virtually endless range of possibilities exists.

Another interesting feature of the matrix-based control structure is the synthesis of smooth trajectories when one of more values fluctuates in the sensitivities-matrix. Since actors move through the consideration of a step-by-step evaluation process, changes gradually accumulate towards a spatial niche of lower social stress—the pull towards the basin of minimum stress decreases as a function of the distance of the actor from that location. In addition, considering one actor, since all its neighboring actors are all engaged in the same process, global behavior crystallizes into trajectories of considerable plasticity—the system produces smooth waves of spatiotemporal patterns. These smooth trajectories of actors in the visual domain are then mapped to the pointer position



responsible for selecting audio segments in earGram's organized database visualization. The resulting sonic feedback matches the dispersion/cohesiveness and continuity of both the overall visual representation and the trajectories of individual actors. Furthermore, stable spatial configurations of the Actor society is further distilled into a blurred sonic texture obtained through spectral "smoothing" and filtering. The longer the actors are inactive, the blurrier the texture becomes and the fewer spectral peaks are synthesized, thus reinforcing the spatial configuration of the visual component of the system in the sonic domain.

## 6 | AESTHETIC CONSIDERATIONS

Gold refers to the Party Planner as a prime example of Algorithmic Symbolism – "a form of art where the underlying procedures of generation contain meaning that interplays with the surface meaning" (Gold, 2007, p. 30). The algorithms are not thought of as background procedures to generate some observable materialization. In contrast, algorithmic activity should be experienced as the core mechanism implied in the generation of a work of art.

Such an orientation suggests the display of a behavioral scope rather than single instantiation. In other words, the algorithm embodies multiplicity - the perception of massive (apparently limitless) yet finite behavioral opportunities. The Party Planner is a foremost example of a complex dynamical system; it is designed explicitly, though we cannot predict its behavior from the observation and analysis of its embedded logic.

A large and diverse body of research evolved based on chaos theory aiming to frame fundamental features of complex systems in a diversity of fields such as sociology, economy and the cognitive sciences (Lewin, 1992; Waldrop, 1994). In the cultural domain, David Borgo developed a captivating analysis of free musical improvisation constructed on principles of complexity theory (Borgo, 2005). Irrespective of any particular field of research, complex systems exhibit self-organizing behavior; particular coherent patterns of wave-like behavior emerges in an ocean of minute interacting components. Unpredictable global performance issues from designed (and therefore, predictable) local interactions. In addition, complex

systems are adaptive to large changes in context, thus we may modify their structure and surroundings in real-time still assuring reliably consistent performance.

From a perspective of aesthetics, a particular fascination developed for complex systems acting at the *edge of chaos* – systems that are neither too static (too much order) nor totally random (no order at all). Intuitively, we assume rich and intriguing behavior to exist somewhere in between these two extremes. In particular, we are interested in changes in complexity – the aesthetic potential of transitory patterns. Spatiotemporal patterns might suggest seemingly goal-oriented behavior and we might infer a particular meaning from an unsuspected transition in complexity.

The following question arises: how do we tune our systems to output this quasi-coherent behavior when the relationship between the rules and the apparent systems behavior cannot be understood and the effect of parametric influence cannot be predicted? Complex systems suggest an exploratory attitude; the HP develops an understanding of the system from tight embodied interaction and negotiation rather than abstract contemplation. In other words, the HP shares decision-making in close partnership with the system. Note that the cyclic activity of developing a system (writing and debugging the software) and exploring its potential (using the system) are thought of as a tightly linked, contiguous and iterative process. Both activities inform each other and propel the overall system functionality in an undetermined direction. The direction is unknown because the guidelines for further exploration develop (are discovered) along the way – that is, within the process of interaction itself.

One might characterize the intimate yet unstable cognitive relationship between programmer and software in the artistic domain as *speculative computing*. Conjecture and speculation imply dynamic, exploratory design; they basically challenge the imagination of the programmer. Fresh ideas may appear spontaneously, yet they are definitely conditioned by earlier private work and global (networked) culture. However, the ultimate confrontation is to acknowledge the instructive undercurrents of expectation and surprise (1) within

the process of software creation and (2) within the behavior of the particular micro-universe in question.

As a socially oriented micro-universe, the Actor model further relates to the essence of musical improvisation since we know that intended goal-oriented behavior surface as temporary attractors of variable complexity; the system's behavior is appreciated in a climate of relative uncertainty. In a wider artistic context, this in itself provides grounds for rewarding human-machine interaction since complex systems seamlessly merge meaning and mystery into an evocative experience.

## 7 | CONCLUSIONS AND FUTURE WORK

Multi-agent systems commonly exhibit complex behavior after multiple local interactions following simple rules. The dynamics of self-organizing systems has been extensively explored aesthetically in artistic settings. Here, we use the Actor model of social interactions to control a concatenative synthesis engine called earGram in real time. The self-organizing behavior of the Actor model was designed to be visually interesting from an aesthetic point of view, exploring the space as complexity emerges from the interactions. This visual complexity is used to aesthetically explore the feature space in earGram, whereby spatial trajectories become gradually evolving sonic textures. Trajectories are a powerful way to control earGram creatively because the spatial configuration reflects perceptual relationships among the sounds. The Actor model provides multiple trajectories, each controlling a sound texture in parallel, which result in an intricate and ever-evolving sonic tapestry.

A fundamental contribution of this work is the use of concatenative sound synthesis, an innovative sample-based synthesis technique, at the core of the software earGram. The integration of earGram with the Actor's model not only offered us more plastic and expressive sonic results in relation to similar approaches—which tend to focus on additive, subtractive or physical synthesis models—but also allowed us to better match the conceptual basis of the system through the synthesis of speech sounds. By adopting a fixed database configuration in earGram, we favored one robust solution over a myriad of possibilities offered by the system. However, the current integration of both systems allows a user to easily experiment with

different audio sources or even different feature spaces (i.e. database organization in the interface), while maintaining the same structural mapping, interactive behavior, and to a certain extent, the aesthetic basis. While adopting a different audio source has a greater impact on the sonic result, changing the feature space that organizes the audio segments database will offer a lower degree of variability, similarly in musical terms to the creation of variations of the same musical material. Ultimately, the positive outcome of this work spurs experimentation on sample-based techniques driven by artificial-life behavior.

User interaction is essential to explore the sonic result. Currently, the user interacts with the system by changing the parameters affinity and sensitivity that control the dynamic behavior of the Actor model. We plan to enhance the interactive feedback loop with a gestural device, such as Microsoft's Kinect. The gestures can be used to change parameter values in real-time. The sonic feedback would be used as system response to the interferences. The performer affects the visual and sonic output indirectly since the gestures do not control the system configuration, only the system parameters. More interestingly, the human performer can use a virtual presence device to interact directly with the actors. In this case, the human performer becomes the external perturbation that continuously upsets the states of equilibrium of the system driven by aesthetic judgments.

## ENDNOTES

[1] The software along with its documentation and many sound examples are available at: <https://sites.google.com/site/eargram/>.

[2] <http://archive.phonetics.ucla.edu>, last access on 7 January 2015.

[3] The spectral freeze effect implemented on the earGram system is largely based on Paul Nasca's Extreme Sound Stretch algorithm accessible online at: <http://hypermammut.sourceforge.net/paulstretch/>, last access on 7 January 2015.

## ACKNOWLEDGMENTS

This work is financed by the FCT - Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within project

“UID/EEA/50014/2013” and by the Media Arts and Technologies project (MAT), NORTE-07-0124-FEDER-000061 is financed by the North Portugal Regional Operational Programme (ON.2 – O Novo Norte), under the National Strategic Reference Framework (NSRF), through the European Regional Development Fund (ERDF), and by national funds, through the Portuguese funding agency, Fundação para a Ciência e a Tecnologia (FCT).

## REFERENCES

Bak, P. (1996) *How Nature Works: The Science of Self-Organized Criticality*. New York: Springer. <http://dx.doi.org/10.1007/978-1-4757-5426-1>

Bernardes, G., Guedes, C. and Pennycook, B. (2013). Eargram: An application for interactive exploration of concatenative sound synthesis in Pure Data. In M. Aramaki, M. Barthelet, R. Kronland-Martinet and S. Ystad, (eds), *From Sounds to Music and Emotions*, LNCS, 7900, 110-129. Berlin-Heidelberg: Springer. [http://dx.doi.org/10.1007/978-3-642-41248-6\\_7](http://dx.doi.org/10.1007/978-3-642-41248-6_7)

Bernardes, G. (2014) *Composing Music by Selection: Content-Based Algorithmic-Assisted Audio Composition*. PhD dissertation, University of Porto.

Beyls, P. (2005). A Molecular Collision Model of Musical Interaction. In C. Soddu (ed.), *Proceedings of the Generative Arts Conference*, Milan, Italy.

Beyls, P. (2010). Structural Coupling in a Musical Agency. In E. Miranda (ed.), *Artificial Life and Music*, Evanston, WI: A-R Editions.

Beyls, P. (2012). Autonomy, Influence and Emergence in an Audiovisual Ecosystem. In C. Soddu (ed.), *Proceedings of the Generative Arts Conference*, Rome, Italy, 2012.

Blackwell, T., Bentley, P. (2002). Improvised music with swarms. In *Proceedings of the 2002 Congress on Evolutionary Computation*, 2, 1462-1467. <http://dx.doi.org/10.1109/cec.2002.1004458>

Borgo, D. (2005). *Sync or Swarm: Improvising Music in a Complex Age*. Continuum. New York and London.

Caetano, M., Manzolli, J. Von Zuben, F. (2007). Self-Organizing Bio-Inspired Sound Transformation. *Applications of Evolutionary Computing*. LNCS, 4448,

477-487.

Camazine, S., Deneubourg, J-L, Franks, N., Sneyd, J., Theraulaz, G., Bonabeau, E. (2003). *Self-Organization in Biological Systems*. Princeton Studies in Complexity.

Casey, M. (2009). Soundspotting: A new kind of process? In R. Dean (ed.), *The Oxford Handbook of Computer Music*. New York, NY: Oxford University Press.

Chadabe, J. (1996). *Electric Sound, The Past and promise of Electronic Music*. Upper Saddle River, NJ: Prentice-Hall.

Dittrich, Ziegler & Banzhaf. (2001). Artificial chemistries, A review. *Artificial Life*, 7(3), 225-275. <http://dx.doi.org/10.1162/106454601753238636>

Gold, R. (2007). *The Plenitude*. MIT Press, Cambridge, MA.

Kandogan, E. (2000). Star coordinates: A multi-dimensional visualization technique with uniform treatment of dimensions. In *Proceedings of the IEEE Information Visualization Symposium*.

Kaufmann, S. (1995). *At Home in the Universe: The Search for the Laws of Self-Organization and Complexity*. Oxford University Press.

Kim, H.-G., Moreau, N., & Sikora, T. (2005). *MPEG-7 Audio and Beyond: Audio Content Indexing and Retrieval*. Chichester, UK: John Wiley & Sons. <http://dx.doi.org/10.1002/0470093366>

Langton, C. (1997). *Artificial Life: An Overview*. Cambridge, MA: The MIT Press.

Lewin, R. (1992). *Life at the Edge of Chaos*. The University of Chicago Press, Chicago, IL.

Miranda, E. (1994). Music composition using cellular automata. *Languages of Design*, 2.

Reynolds, G. (1987). Flocks, Herds and Schools: A Distributed Behavioral Model. In *SIGGRAPH 87 Conference Proceedings*, Anaheim, CA. <http://dx.doi.org/10.1145/37401.37406>

Schmeder, A., Freed, A., and Wessel, D. (2010). Best Practices for Open Sound Control. In *Linux Audio Conference*, Utrecht, Holland.

Schwarz, D. (2000). A system for data-driven concatenative sound synthesis. In Proceedings of the COST G-6 Conference on Digital Audio Effects (DAFx-00), 97-102.

Schwarz, D., Britton, S., Cahen, R., & Goepfer, T. (2007). Musical applications of real-time corpus-based concatenative synthesis. In Proceedings of the International Computer Music Conference, 47-50.

Sturm, B. L. (2004). MATConcat: An application for Exploring Concatenative Sound Synthesis Using MATLAB. In Proceedings of the 7<sup>th</sup> International Conference on Digital Audio Effects (DAFx'04).

Sturm, B. (2006). Adaptive Concatenative Sound Synthesis and its Application to Micromontage Composition. *Computer Music Journal*, 30(4), 46-66. <http://dx.doi.org/10.1162/comj.2006.30.4.46>

Tidemann, A., Ozturk, P. (2007). Self-organizing Multiple Models for Imitation: Teaching a Robot to Dance the YMCA. *New Trends in Applied Artificial Intelligence, LNCS*, 291-302. [http://dx.doi.org/10.1007/978-3-540-73325-6\\_29](http://dx.doi.org/10.1007/978-3-540-73325-6_29)

Todd, S., Latham, W. (1992). *Evolutionary Art and Computers*. Academic Press.

Ulanowicz, R. (1979). Complexity, Stability, and Self-Organization in Natural Communities. *Oecologia (Berl.)*, (43), 295-298. <http://dx.doi.org/10.1007/BF00344956>

Waldrop, M. (1994). *Complexity, The Emerging Science at the Edge of Order and Chaos*. Penguin Science Books, London, UK.

## **BIOGRAPHICAL INFORMATION**

Peter Beyls holds a PhD in computer science from the University of Plymouth, UK. His current research focuses on emergence in gestural control structures, participative audiovisual installations and the representation of musical intelligence in human-machine interaction. Since 2013, Beyls has been a research professor at CITAR, Center for Science and Technology for the Arts, Universidade Católica Portuguesa, in Porto.

Gilberto Bernardes received a Master's in Music Performance from the Amsterdamse Hogeschool voor de Kunsten and the PhD in Digital Media from the University of Porto under the auspices of University of Texas at Austin. He was awarded the Fraunhofer Portugal Challenge 2014 prize for his PhD dissertation. Currently, he is a postdoctoral researcher at INESC TEC where he pursues work on automatic music generation and an Associate Professor at the Polytechnic Institute of Castelo Branco. Dr. Bernardes is active as saxophonist, new media artist, and researcher in sound and music computing.

Marcelo Caetano received the Ph.D. degree in signal processing from Université Pierre et Marie Curie - Paris 6 in 2011 under the supervision of Xavier Rodet, head of the Analysis/Synthesis group at IRCAM. He was a Marie Curie postdoctoral fellow with the Signal Processing Laboratory at FORTH in 2012-2013. Currently, he is a postdoctoral fellow with the SMC group at INESC TEC and invited Assistant Professor at FEUP, University of Porto. Dr. Caetano's research interests range from musical instrument sounds to music modeling, including analysis/synthesis models for sound transformation and music information retrieval.