'SOUND IS THE INTERFACE'
Sketches of a Constructivistic Ecosystemic View of Interactive Signal Processing

Agostino Di Scipio

Scuola di Musica Elettronica, Conservatorio di Napoli, Italy
discipio@tin.it

ABSTRACT

This paper takes a systemic perspective on interactive signal processing, and introduces the author's *Audible Eco-Systemic Interface* project (AESI). Based on a constructivistic view, it discusses cybernetics principles of a bio-ecological kind (energy exchange, structural closure, organizational openess, system/environment coupling) in the design of signal processing interfaces, with implications for actual musical work (not only live performance situations or studio work, but also sound installations).

Crucial to the subject is an understanding of 'interaction' in terms of a network of interdependencies among system components, and in terms of the system dynamics and its 'structural coupling' to the external environment. The paper illustrates the meaning of these notions as actually implementated in the current release of the AESI (created with Kyma5.2).

1. INTRODUCTION

Talk of 'interactivity' is today common, ubiquitous. And often meaningless, too. The history of the 'interactive arts' and their paradigms [1] is documented in an overwhelming body of literature.

It is not the goal of this paper to overview existing work in the area of 'interactive music systems'. It rather focuses on the paradigm of 'interaction' inherent in most efforts in this research area, and particularly on interactive signal processing interfaces. What kind of 'systems' are musical systems called 'interactive'? I try to answer adopting a particular system-theory view – a *radical constructivistic* view [2], of a kind that emerged from efforts in the cybernetics of living systems [3], including social systems and 'ecosystems' [4]. This requires a reformulation of the notion and function of what is meant by 'interaction'. To illustrate the argument, I describe the design philosophy developed in some recent personal efforts, resulting in the *Audible Eco-Systemic Interface* project, with the aim to implement a kind of 'sonorous ecosystem'.

2. WHAT KIND OF SYSTEMS ARE 'INTERACTIVE MUSIC SYSTEMS'?

Typical interactive music systems can be viewed as dedicated computational tools capable of reacting in some way upon changes they detect in their 'external conditions', namely in the initial input and the run-time control data. Such data are usually set and adjusted by some agency – a performer, or group of performers (could be a composer, too, either working in the studio or experimenting on stage, improvising) – using some control device (mechanical or visual interface).

Therefore, *the agent's operations, as reflected in the control data, implement the system external conditions and all changes therein.*

The agent's operations upon the control data are turned by the computer (according to a variety of digital signal processing techniques and/or program routines operating at a higher, more symbolic level) into sound events, or cause some transformation of input sound material. Upon hearing, the agent adjusts the data by means of the available controls, eventually 'playing' the system almost as if it were a new kind of music instrument (however, the 'instrument' metaphor cannot really be generalized). A variety of known interactive performance situations have been described in available publications on these matters, ranging from the 'solo instrument' set-up, to the 'duo' and larger 'ensembles', where many performers and/or computer systems are interconnected and play together (see [5] for a summary).

When it comes to live digital signal processing interfaces, it becomes clear that the system design itself, and particularly the interactions mediated by the user interface (interdependencies among control variables), become the very matter of composition [6] and can no longer be separated by the internal development of sound. Interestingly, in recent work we can observe that not only the visual interfaces, but also the design of mechanical devices can be crucial to the extent that it directly affects the details in the output sound (a peculiar example is [7]).

Notwithstanding the sheer variety of devices and protocols currently available, all 'interactive music systems' – including developments specific to the internet – share a basic design principle, namely a linear communication flow (figure 1).

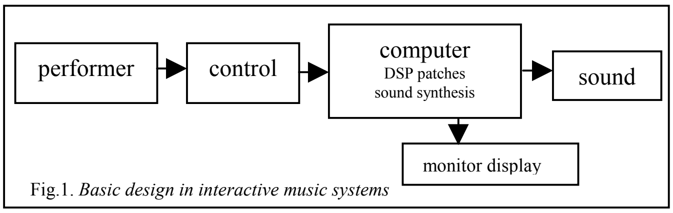

Fig.1. *Basic design in interactive music systems*

Usually, this design assumes an implicit recursive element, namely a loop between the 'sound' and 'performer': the computer output somehow affects the performer's next action (reaction), which in turn will eventually affect the computer system in some way (figure 2).

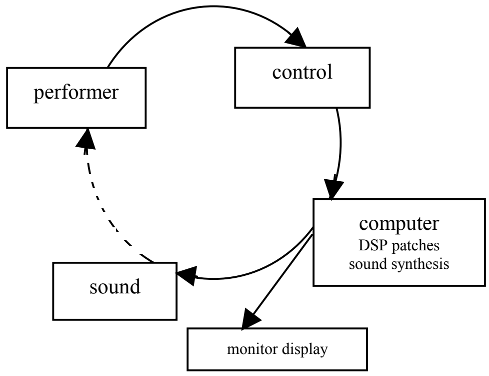

Fig.2. *Implicit feedback loop in interactive system design*

That recursion is a potential source of dyamical behaviour, of noise and unpredictable developments. Yet the basic design remains a linear one, i.e. from the performer's action to the computer reaction. The performer is first the initiator agent of the computer's reaction, and only secondly, and optionally, might become the very *locus* of feedback, injecting some features of nonlinearity into the overall system ('system', then, includes the performer). Most 'live electronics' music can be referred to that design, which also returns in most computer music endeavours, even in very recent discussions [8].

2.1 *Where* and *when* are systems 'interactive?

In the above mentioned approach, 'interaction' means that the computer change its internal state depending on the performer's action, and that the latter is potentially itself oriented in his/her action by the computer output (dashed line in figure 2). There – in the perfomer's mind and ears – lies the only source of unforeseen developments, of dynamical behaviour. Of course one could introduce all sorts of complex transfer functions in the mapping of data from one domain to another (similar to many musical instruments with their complex mapping of gesture into sound) but that is an option which is not in essential in the ontology reflecting 'interaction': *agent acts, system re-acts*.

Put it in another way, in existing 'interactive music systems', and particularly in their DSP components, *the system is not itself able to directly cause any change or adjustment in the 'external conditions' set to its own process*, it does not generate or transform the control data it needs to change its own internal state. The model prompted includes no recursion.

Also, 'interaction' is not usually referred to the mechanisms implemented within the computer: the array of available generative and/or transformative methods usually consists of separate functions, i.e. they are conceived independent of one another, and function accordingly. The agent selects the particular function(s) and process(es) active at any given time, and the latters' output data are simply summed together. As an example, the sudden occurrence of, say, too large a mass of notes or sound grains or other musical atom units would not automatically induce a decrease in amplitude (a perceptually correlate dimension of density): such an adjustment remains a chance left to the performer. No interdependency among processes is usually implemented within the 'interactive' system.[1] 'Interaction' (either between system and external conditions, or between any two processes within computer) is rarely understood and implemented for what it is in real living systems (either human or not): a by-product of implemented lower-level interdependencies among system components. Interfaces usually do not allow to create a communication between DSP processes, but only to independently handle their parameters, in the form of runtime variables.

3. FROM 'INTERACTIVE COMPOSING' TO 'COMPOSING THE INTERACTION'

In a different approach, the aim would be first of all to create a *dynamical system*, possibly exhibiting an adaptive behaviour to the surroundings ('external conditions'). It should be able not only to 'hear' what happens 'out there' (an 'observing' system, capable of tracking down relevant features of the external world, not demanding this from an external agency), but also to become in the end a *self-observing* system [10], ultimately using information on the external conditions to orient its own internal sequence of system states (*self-organization*).

This idea is motivated with a notion of 'interaction' as a means, not an end in itself, i.e. a prerequisite for something like a 'system' to emerge. Therefore, 'interaction' should be the object of design (hence, composition), and more precisely the by-product of carefully planned-out interdependencies among system components.[2] The overall system behaviour (dynamics) should be born of those interactions, in turn born of lower-level interconnections.

This is a move from 'interactive performance', or 'interactive composing' (as in the pioneering work of Joel Chadabe and others) to 'performing the interaction', or 'composing the interaction'. In the latter approach, one designs, implements and maintains a network system whose emergent behaviour in sound one calls music. When a such a system enters a non-destructive relationship to the surrounding environment – the system's *house*, literally: its οικος – it becomes an *eco-system*.

4. THE *AUDIBLE ECO-SYSTEMIC INTERFACE*

The *Audible Eco-Systemic Interface* (AESI) reflects a self-feeding loop design (figure 3). A chain of causes and effects is established without any intervention on a human performer's part (but that doesn't mean that a perfomer cannot enter the loop[3]): the

[1] The observation also applies to 'interactive music systems' capable of 'listening' to, say, an instrumentalist playing thereby making decisions based on features tracked down ('machine listening' [9]). In 'score following', runtime control variables are updated following the successful or unsuccessful matching of an instrumental performance against a stored event list (score): therefore here, too, decisions are 'dynamically' made by means of a predetermined knowledge-base (score representation).

[2] A similar approch is in George Lewis' interactive improvisation programs and musical pieces [11].

[3] Note that 'interaction' here should not be implicitly understood as 'man / machine interaction'. The AESI project leans on 'ambience / machine interaction'. Clearly, the ambience may host one or many performers. A similar annotation should be made for 'interface', as used later in the paper.

computer emits some initial sound, that is heard through the loudspeakers and also fed back into the computer, by microphones scattered around the performance place; the computer analyzes the microphone signals, and the features accordingly extracted are used to generate low rate control signals and drive the system's internal process (transformations or synthesis of sound material); also, the computer matches the microphone signals against the original synthetic or sampled signals, and the difference-signals (the difference values between original and fed-back sound), reflecting the contribution of the room resonances to the sound as actually heard, are used to adapt the computer performance to the room response: the AESI variables are in a constant flux of changes depending on the resonances in the environment, as elicited by the sound event initally emitted.

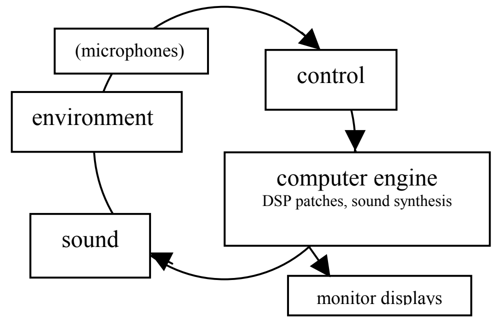

Fig.3 *Overall conceptual design of the AESI project*

The sound (room resonances) affects the DSP control variables, and the latter affects the sound. By means of this recursive coupling, the transformative or generative functions implemented in the computer become *iterated functions*. If they include some nonlinear mapping, they become *nonlinear iterated functions*, a source of dynamical behaviour inherent to the system, can subject to be modelled in mathematical terms [12]. Different from more usual interactive systems, this recursion is structural to the overall design, not optional: it is independent of human performers, although clearly human performers can enter the process cycle and become a part of the overall system (yet he/she would not be the only source of dynamical behaviour).

Observe that, because here the fed back sounds become source signals for the generation control signals, the feedback loop in this design is not in the audio frequency domain, but confined in the low frequency domain (control rate). The precise rate depends on the time scales in the feature-extraction processes, the integration time of observing functions, and other time-related variables in the mapping of control data (time dilation of control signals, time maps of input data, etc). The role of time variables is really crucial in the AESI project, as their side-effects contribute to install an element peculiar to dynamical systems such as ecosystems: namely the fact that dramatic long-term consequences could follow from what currently appears as marginal detail.

4.1 Ecosystemic dynamics

Variables. In the AESI project, system variables are represented by the control signals generated with feature-extraction processes. Sonic features so far taken into account include: amplitude, density of events (rate of onset transients), transient delay between microphones (resulting in perceptual precedence effects), and psychoacoustical variables such as 'brilliance' and other statistical spectral properties. Clearly, pitch and other frequency-related measures could be also considered. Beside, it is good idea to create many control signals out of a single feature-extraction process.[4]

Functions. Considered in the AESI are the following functions (described in their systemic meaning, because the particular math expression can change depending on context): *compensation* (e.g. decreasing amplitude, either that of the input or the output to some process, when the overall density increases); *following* (running after and finally matching some variable, with some delay; e.g. the *hysteresis* effect, implemented as *latency* filter circuits); *redundancy* (supporting a given predominant feature, rather than contrasting it; e.g. automatically adjusting the integration time of variable followers in order to make them identical with, or close to, the rate of repeated events – as used, for example, in dynamical tempo tracking and other adaptive filters); *concurrency* (giving way to an alternative feature competing with the predominant one; e.g. introducing high-frequency spectral energy when low-frequency energy is predominant in the room).

The overall goal of these functions is to implement and regulate the interdependencies among variables, depending on external conditions. By way of that, they make the internal routines and DSP modules to interact in a meaningful way. Finally, by means of those emergent interactions, they indirectly implement the system dynamics.

Depending on the long-term effects, a function contributes to install the system overall orientation. One can consider at least the follwing two

Competing orientation criteria. Include *omeostasis* (the system's tendency to keep and mantain a constant or recurrent behaviour, which determines its 'identitiy' to an external eye or ear); and *omeoresis* (the opposite tendency, to introduce and follow a more varied behaviour, which determine the system's ambiguity to an external eye or ear).

These contrasting criteria are implemented in terms of the above functions, and should always counterbalance one another. A rich system dynamics is typically achieved only as a result of their competition.

4.2 Some observations

In the AESI project, the overall system/environment coupling function is not explicitly formulated: it emerges as a property of the system dynamical self-organization (its ability to change state and to change the interdependencies among system components). The particular self-organizing behaviour in turn depends not only on the mapping of the control signals onto the parameter space (for the given sound synthesis or processing algorithms adopted), and the time variables (time scales at which the input sound stream is analyzed in the feature-extraction process, and time

[4] The real-time generation of control signals is an art in itself, and opens questions that are worth of a specific discussion.

maps of control signals). It also heavily depends on the sound material being used to excite the environment resonances, and clearly on the specific room acoustics (material and geometrical characteristics of the system environment).

The AESI runtime process unfolds as it 'learns' about the environment. Its coupling to the latter is indirect: it happens in the medium of sound. In a way that is *not* at all metaphorical, *by way of doing something to the environment* (sending sounds to it) it actually learns about itself as a system and develops its own internal organization.

Finally, note that all exchanges between the AESI and its οικος take place in the medium of sound: *sound is the interface* (all processes and equipement involved, e.g. microphones, are vehicles or transformers of sound signals).[5] As finally perceived by the listeners, sound bears traces of the structural coupling it is born of. One can speak of *audible interfaces*, i.e. interfaces whose process (the mediation between system and environment) *is* or *can* be something actually heard.

5. PRELIMINARY CONCLUSIONS

A thorough technical description of the AESI implementation is out of the scope of the present paper (it would extend to the discussion of the particular DSP methods adopted, the feature-extraction algorithms themselves, etc. – not to mention the type of microphones, their placement, relative distance and orientation, the size and geometry of the room, placement of loudspeakers, etc.). Furthermore, many details have been so far only planned out but not fully implemented yet. The purpose of the present paper was to introduce the perspective and the basic motivations.

In this approach the notion of 'interaction' is reformulated (and implemented) in terms of dynamical interdependencies among system components. The idea that a computer 'reacts' to a performer's gesture is replaced with a 'structural coupling' of system and environment. The system *acts upon* the environment, observes the latter's reactions, and then reacts based on the environment's response. Also, by tracking down its own previous internal states, and previous interactions with the ambience, it developes based on its own history, i.e. 'cognizant' of the past (system's *memory*, long term effects, etc).

Reflecting a radical constructivistic epistemology [15], the *Audible Eco-Systemic Interface* represents a 'structurally closed' but 'organizationally open' system. The 'closure' is meant to preserve the system identity (that's the case with typical 'interactive systems', too). The 'openess' reflects the system's ability to exchange energy (sound) with the environment [13] and to determine its own internal causal sequence of states, thanks to a rich dynamics emerging in the interactions among system components (this is not the case with existing 'interactive music systems').

Addendum

The current AESI implementation was created with SymbolicSound's KYMA5.2 (using the CAPYBARA320 as DSP engine). In a preliminary musical work composed with and for the AESI, titled *Ecosistemico udibile n.1* (2002), short sound impulses were used as the only raw sound material introduced into the feedback loop of the AESI. That was precisely in order to experimentally test the system's *impulse response*. The pulse material had been created with the PULSARGENERATOR program [14] during the author's composer residency at CCMIX (Centre de Creation Musicale Iannis Xenakis), in Paris, April 2002.[6]

References

[1] Dinkla, S., "The History of Interface in Interactive Art". *ISEA Proc.* 1994.
[2] von Glasersfeld, E., "The Roots of Constructivism". Paper presented at the Scientific Reasoning Research Institute, 1999.
[3] Maturana, H. and Varela, F., *Autopoiesis. The realization of the living*. Dordrecht, 1980.
[4] Morin, E., *La méthode. La nature de la nature*, Paris, 1977.
[5] Rowe, R., *Interactive Music Systems*, MIT Press, 1993
[6] Hamman, M., "From Symbol to Semiotic: Representation, Signification and the Composition of Music Interaction," *Journal of New Music Research*, 28(2), 1999.
[7] Smith, T. and Smith, J.O., "Creating Sustained Tones with the Cicada's Rapid Sequential Buckling Mechanism", *NIME Proc.*, 2002
[8] Schnell, N., and Battier, M., "Introducing Composed Instruments. Technical and musicological implications", *NIME Proc.*, 2002.
[9] Rowe, R., *Machine musicianship*, MIT Press, 2001
[10] von Foerster, H., "On self-organizing systems and their environment." In (C.Yovits, ed.) *Self-organizing systems*, New York, 1960. Also: von Foerster, H., *Sistemi che osservano* (edited by M.Ceruti and U.Telfener), 1987.
[11] Lewis, G., "Interacting with the Latter-Day Musical Automaton", *Contemporary Music Review*, 18(3), 1999.
[12] Collet, P., and Eckmann, J.-P., *Iterated Maps on the Interval as Dynamical Systems*, Boston, 1980.
[13] von Bertanlanffy, L., *General System Theory*, NY, 1968.
[14] Roads, C., "Sound Composition with Pulsars", *Journal of the Acoustical Engineering Society*, 49(3), 2001.
[15] see Riegler, A. www.univie.ac.at/constructivism/, 2000.

[5] In a similar vein, [11] writes: "There is no built-in hierarchy of human leader/computer follower: no 'veto' buttons, pedals or cues. All communication between the system [he means the computer, which is just a system component, though] and the improvisor takes place sonically. [Such] a performance… is in a very real sense the result of a negotiation…" (p.104). While in Lewis' approach the 'improvisor' is the only 'ambience' set to the computer, the only source of external conditions (noise), in the AESI project the human component is another possible source of information inhabiting the shared ambience.

[6] This composition was premiered in a concert at Keele University, Stoke-on-Trent, UK (October 2002). Kurt Hebel took care of the set-up and supervised the system performance. The italian prèmiere will take place in the 14th CIM concerts, in Florence, with the supervision of Alvise Vidolin.