

Agostino Di Scipio:

**„Audible Ecosystemics
n.3a Background Noise Study“
n.3b Background Noise Study with Mouth Performer“
„Modes of Interference“**

Mariano Mora

Sinopsis

Análisis de dos obras de Agostino Di Scipio, incluyendo una discusión de los principios teóricos que subyacen a ambas: sistemas autónomos y autoreguladores (autopoiéticos), sistemas cibernéticos e interacción entre un intérprete y un sistema de tratamiento de señal digital en tiempo real (*real-time DSP*). Asimismo la ausencia de material sonoro, siendo la única fuente de material musical el sonido de ambiente (*background noise*) de la sala donde se desarrolla la obra. El análisis técnico trata brevemente el método de síntesis granular, así como el análisis de los *patches*, realizados en Kyma y en PureData.

Palabras clave

Di Scipio, *autopoiesis*, música electrónica en tiempo real, sistemas autoreguladores, sonido de ambiente.

1. Introducción.

El trabajo del compositor italiano Agostino Di Scipio (Nápoles, 1962) viene atrayendo cada vez más atención entre el público de música digital europeo. Profesor de música electrónica en el conservatorio de Nápoles, conferenciante en el CCMIX de París, además de compositor invitado de la DAAD en Berlín (2004-5), sus propuestas compositivas y su labor como teórico están contribuyendo a establecerle como uno de los compositores más coherentes e interesantes del panorama musical europeo. Su producción incluye obras electrónicas, trabajos en cinta, instalaciones sonoras y tratamiento en tiempo real de material procedente de instrumentos acústicos.

Los dos trabajos que nos ocupan forman parte de su producción más reciente (2004-5 y 2005-6, respectivamente) y recogen casi todos los elementos más notables de su desarrollo como compositor. Ambos representan una parte de la aportación de Di Scipio al problema de la interactividad entre el intérprete y un sistema cibernético. La solución ensayada por Di Scipio muestra, a mi juicio, muy bien su fértil imaginación y su capacidad para transformar un problema en una fuente muy prometedora de ideas musicales. Si es cierto que la importancia de una idea no reside tanto en las preguntas que responde, sino en las nuevas preguntas que obliga a formular, entonces el pensamiento de Di Scipio debe ser considerado como muy importante, como una estimulante incógnita para aquellos de nosotros que queramos intentar descifrarla.

Ceñirnos a las dos obras que analizamos supone tener que dejar a un lado algunos de los aspectos más interesantes del pensamiento teórico de Di Scipio[1]. Nos referimos a desarrollos tanto técnicos: síntesis iterativa, sistemas de síntesis no-lineal (caótica), etc; como teóricos: la labor del compositor, la estética del fracaso (del error), y muy especialmente la crítica (en sentido de Feenberg) de la tecnología. Estas cuestiones deben ser, en gran parte, relegadas a la bibliografía[2].

Otros aspectos teóricos son, no obstante, ineludibles. Resultará muy difícil comprender la realización musical sin habernos enfrentado antes a algunas ideas que, aunque de extenso arraigo en ciencias naturales, no resultan tan frecuentes en pensamiento musical. Nos referimos al concepto de *autopoiesis*, originario de la biología, y del concepto de sistema autoregulator, procedente de la cibernética.

El análisis en sí consiste en un seguimiento paso a paso del flujo de información dentro del *patch*. En música „algorítmica“ el algoritmo es la pieza: el resultado sonoro es „una huella que la composición deja a su paso“[3]. El análisis también incluye las instrucciones al intérprete, que serán puestas en el contexto del flujo de información.

2. *Autopoiesis* y sistemas autoreguladores.

„una máquina autopoietica continuamente especifica y produce su propia organización a través de la producción de sus propios componentes, bajo condiciones de continua perturbación y compensación de esas perturbaciones“[4]

„Las máquinas autopoieticas no tienen entrada ni salida. Pueden ser perturbadas por hechos externos, y experimentar cambios que compensan esas perturbaciones. (...) Sin embargo, cualquier serie de cambios internos que se produzca está siempre subordinada a la conservación de la organización de la máquina.“[5]

El concepto de *autopoiesis* fue sugerido por vez primera por los biólogos chilenos Humberto Maturana y Francisco Varela como concepto definitorio de la vida. Para estos autores, aquello que define a los seres vivos es la capacidad que éstos muestran de regular las relaciones entre sus componentes de tal manera que se asegure la propia conservación. De hecho, los seres vivos *son* este conjunto de relaciones, y en el momento en que la interrelación entre sus componentes se rompe, entonces la vida se desintegra. Lo característico de dichas interrelaciones es su circularidad: los seres vivos producen sus propios componentes, componentes que a su vez sirven para regular la actividad del propio ser vivo.

El lector observará que las citas refieren específicamente a *máquinas*. Esto es una muestra de lo extraordinariamente fructífero que este concepto ha demostrado ser. Como narra el mismo Maturana, la idea de la circularidad de la

organización de un ser vivo surgió a partir del estudio de las células[6]. No obstante, los mismos autores pronto extendieron la aplicación a la psicología cognitiva, y más tarde a la cibernética. Niklas Luhmann adaptó el concepto a las organizaciones sociales... en fin, la *autopoiesis* se convirtió en un paradigma.

La visión de Maturana y Varela es estructural: lo que caracteriza a los sistemas autopoieticos es una determinada forma de organización. Mejor dicho, una forma de determinar esa organización: la autopoiesis es un proceso, o un conjunto de procesos. Los sistemas autopoieticos están organizados de una forma tal que permite (y necesita de) estos procesos. Y son estos procesos los que mantienen y definen esa organización.

Hemos utilizado el término sistema de manera intencionada. El interés del concepto reside para Di Scipio en su aplicación a la cibernética, o el estudio de sistemas. Recordemos brevemente qué es un sistema[7]:

- Un sistema es un conjunto de elementos.
- Los elementos de un sistema interactúan.
- La interacción entre los elementos de un sistema toma la forma de comunicación (transmisión de información).
- La comunicación entre elementos sirve a un objetivo: la regulación.

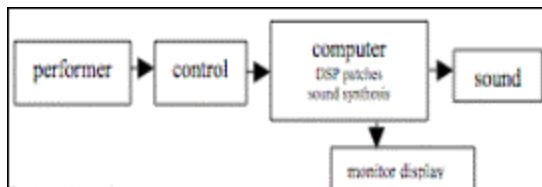
El principio fundamental de la cibernética es el cambio. Un sistema cambia en el tiempo, y a través de esos cambios el sistema alcanza diversos estados. La regulación de un sistema tiene como objetivo alcanzar determinados estados. En un sistema complejo cada elemento aporta algo al estado general del sistema, y con ello determina en parte el siguiente estado. Una secuencia de estados se llama „trayectoria“. Un sistema se encuentra en estado estable cuando las aportaciones de los diferentes elementos que lo componen no llevan al sistema a un nuevo estado. Una perturbación, sin embargo, cambia el estado del sistema (si es suficientemente fuerte), llevando al sistema a un nuevo estado. Veremos más tarde cómo el sistema responde de diferentes formas a las perturbaciones para acomodarse a un nuevo estado. Estas maneras de responder serán fundamentales, puesto que en ellos consiste la paleta de que dispone el compositor para determinar la interacción entre los diferentes elementos del sistema.

Uniendo ambos conceptos, vemos que los sistemas autopoieticos se caracterizan por ser conjunto de elementos, entre los cuales se establecen unas relaciones de comunicación que tienen como objetivo regular la organización del sistema. Todas las aportaciones de los elementos del sistema sirven para que el sistema mantenga su propia organización: el sistema se autoregula.

2.1. Sistemas musicales

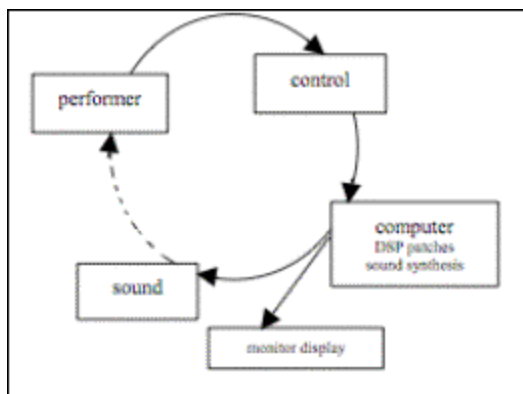
Tanto en la serie "Audible Ecosystemics" como en "Modes of Interference" intenta Di Scipio repensar el paradigma de sistemas musicales desde la raíz. El prototipo de sistema musical consiste en el sistema interactivo. En un sistema

interactivo, un procesador de DSP (un ordenador con módulo de tratamiento de sonido) está programado para reconocer cambios en las condiciones externas y reaccionar a ellos de una forma determinada. Los cambios a los que hacemos referencia consisten normalmente en *input* nuevo, o bien en cambios en los parámetros de control. Un ejemplo aclarará esto. Un intérprete toca en un escenario. El ordenador está programado para reaccionar de determinada manera a algún gesto del intérprete, por ejemplo, mantener una nota durante 5 segundos, y reaccionar de otra manera a otro gesto, por ejemplo tocar muy fuerte. Un nuevo input consiste en cambiar el gesto. El intérprete toca primero durante cinco segundos, y después toca muy fuerte. Un cambio en los parámetros de control consiste en variar la manera en que el ordenador debe reaccionar, o cambiar el estímulo que causa la reacción del ordenador. En ambos casos, hay un agente que es el responsable de estos cambios. El proceso comunicativo de este tipo de sistemas toma esta forma[8]:



Este esquema es válido tanto para tratamiento de sonido en tiempo real como para cinta.

En este tipo de sistemas se utiliza también muy a menudo la recursión: el *output* del ordenador determina la siguiente acción del intérprete, la cual a su vez vuelve a determinar la reacción del ordenador. El esquema es:

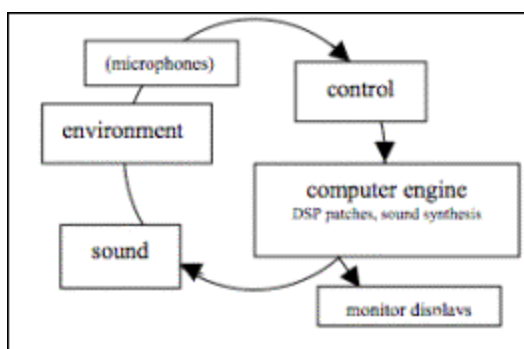


El sistema representado en este esquema es "abierto" a la introducción de nuevas fuentes de energía; "sin embargo, no tiene la capacidad de provocar directamente por sí mismo cambios o adaptaciones en las condiciones externas"[9]. Expresado en términos ya conocidos: el sistema no es autopoietico. En este sentido, el sistema no es verdaderamente interactivo, dado que el flujo de comunicación se interrumpe. Los sistemas autopoieticos

establecen relaciones con su medio, son capaces de provocar cambios en las condiciones externas, estableciendo de esa manera un ecosistema, se convierten en ecosistemas. Un ecosistema desarrolla medidas de regulación que le permiten adaptarse a cualquier cambio en las condiciones, pero al mismo tiempo ejercer una influencia sobre estas condiciones, de tal manera que el ecosistema tiende hacia un estado estable. Tomando un ejemplo de Di Scipio: en el sistema representado arriba *"demasiadas notas o partículas sonoras (más sobre esto abajo) o de cualquier otro tipo de unidades sonoras atómicas no*

provoca automáticamente una reducción de la amplitud (una compensación perceptual en la dimensión de densidad): la compensación continúa siendo una decisión del intérprete"[\[10\]](#).

La propuesta de Di Scipio consiste en un desplazamiento de la perspectiva. En lugar de "componer de manera interactiva", pasar a "componer la interacción". La idea es crear un sistema que pueda recoger información de las condiciones externas y utilizar esa información para organizar su propia secuencia de estados (*auto organización*). "Cuando un sistema de este tipo entra en una relación no destructiva con su entorno- la casa del sistema, su oikoV-, entonces se establece un *ecosistema*. Un sistema de este tipo puede representarse de la siguiente forma:



Este sistema es un flujo de comunicación circular. Los datos iniciales se introducen en el sistema a través de micrófonos distribuidos por la sala donde se celebra el concierto. Los micrófonos, no obstante, recogen sólo el sonido de ambiente, sonido que ya se encontraba en la sala. Este sonido recibe un tratamiento por parte de procesamientos de DSP. El resultado es sonido, que vuelve a extenderse por la sala. Y aquí se produce la iteración, dado que el sonido es nuevamente recogido por los micrófonos. En esta ocasión, sin embargo, se extraen una serie de datos de este sonido, datos que veremos en el análisis del algoritmo. El sistema utiliza estos datos para regular su propio comportamiento, para regular su propia organización. Las funciones reguladoras ejercen una influencia sobre el sonido, y al mismo tiempo son objeto de una influencia que el sonido ejerce sobre ellos. Estas funciones reguladoras/controladoras serán analizadas más abajo en profundidad. En cuanto al papel de la intérprete, veremos que ella también forma parte del sistema, también ejerce una influencia sobre el sonido y es a su vez influido por éste. El intérprete se comporta también como una función reguladora, también puede participar en la trayectoria del sistema. Veremos asimismo que Di Scipio asigna una serie de instrucciones al intérprete que le permiten contribuir a la regulación del sistema. Todo intercambio entre el sistema y su casa se produce por medio de sonido: el sonido es el *interface*, el mediador.

3.Audible Ecosystemics

Background Noise Studies 3a, 3b

En el artículo aparecido en 2005 “*Klangstaub: die Notwendigkeit einer ästhetischen Orientierungslosigkeit*” (Polvo de ruido: la necesidad de una falta de orientación estética), Agostino Di Scipio describe en palabras propias las dos piezas que hoy analizamos de la serie “Audible Ecosystemics”. Propongo utilizar esta descripción como mapa de nuestro análisis:

“Background Noise Study’ consiste en una red de interacciones sonoras entre una serie de aparatos (micrófonos, altavoces, un ordenador) y el entorno en el que se encuentran. Se localiza una fuente de sonido de ambiente en la sala en la que se desarrolla el concierto y se coloca un micrófono cerca de ella. Otro micrófono se coloca cerca de otra fuente de sonido de ambiente, o bien en el centro de la sala, con el objeto de captar el sonido de ambiente general de la sala. Todo el material de la composición durante todo su desarrollo consiste en estas pequeñísimas informaciones acústicas, y en nada más. Los sonidos recogidos por los micrófonos son amplificados masivamente, y después de unos segundos pueden oírse a través de un grupo de altavoces. Tan pronto como se producen acontecimientos sonoros de una intensidad determinada, o bien cuando la amplificación del sonido de ambiente se ha acumulado hasta un punto en el que el entorno sonoro de la sala se ha enturbiado, entonces un ordenador comienza a producir transformaciones sonoras, que son a su vez difundidas a través de los altavoces. Estas transformaciones incrementan gradualmente la densidad y el volumen en proporción directa al ruido de fondo y a la “contaminación sonora” (Klangverschmutzung) de la sala. Sin embargo, las transformaciones se detienen automáticamente cuando se alcanza un nivel a partir del cual sólo se produce saturación. Esto permite que entren en juego medidas de compensación, o sencillamente las condiciones de inicio (solamente sonido de ambiente). Todo el proceso se repite cuatro o cinco veces con diferentes resultados, ya sea porque el sonido de ambiente se ha incrementado con el tiempo, o porque la interacción entre las máquinas y su entorno dejan tras de sí restos sonoros, que determinan las interacciones que siguen.”^[11]

En primer lugar expondremos la manera de captar el sonido. Después describiremos la función de este sonido como fuente de material, es decir, analizaremos el proceso de DSP. Aquí seguiremos la partitura de [Background Noise Study 3ª \(DESCARGAR FORMATO .pdf\)](#) analizando el flujo en dos partes: la primera parte se ocupa de las señales de control extraídas a partir del sonido en tiempo real; la segunda parte rastrea el discurrir de la señal de audio misma. A esto seguirá una breve descripción de la espacialización del sonido.

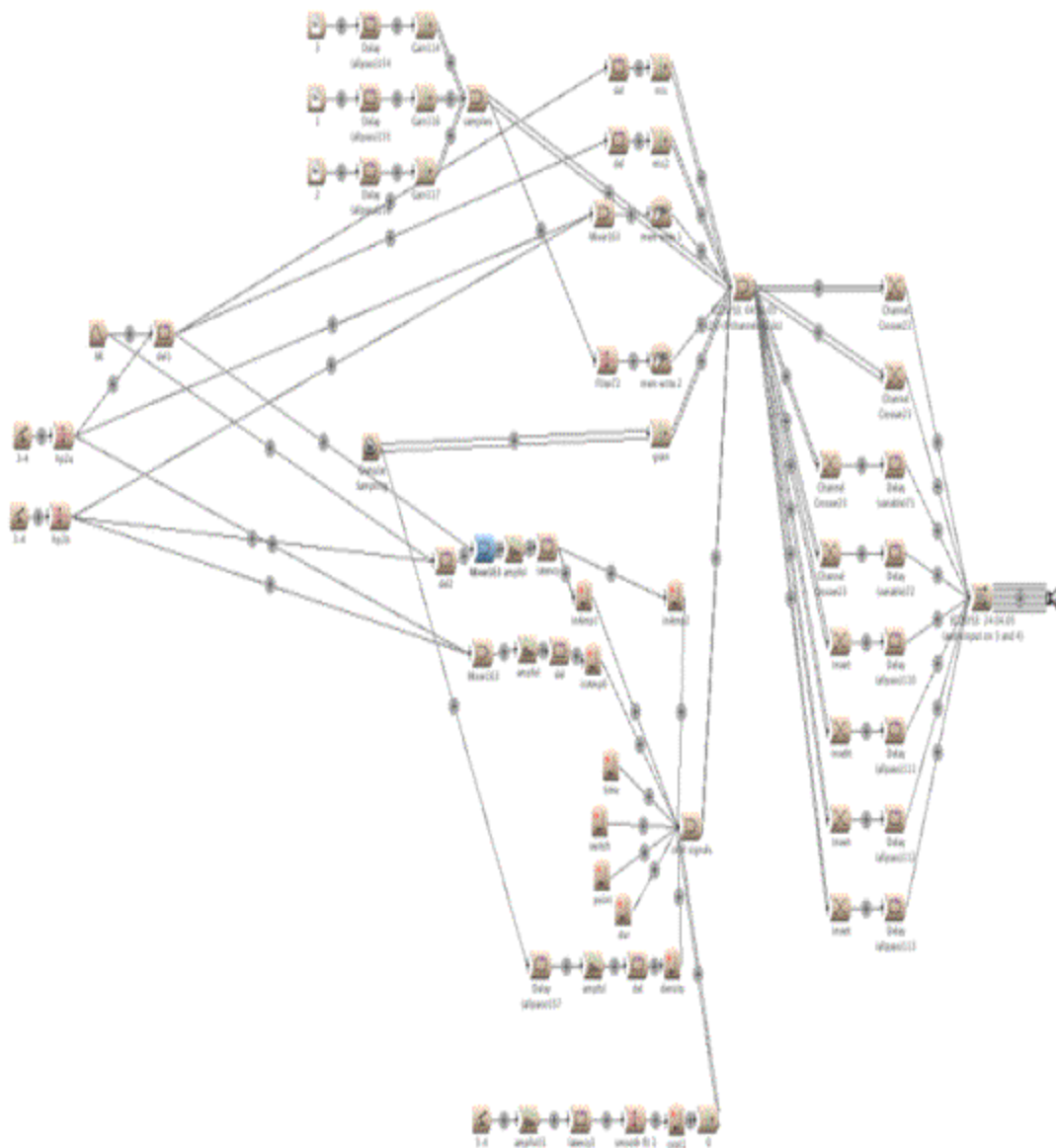
El tratamiento de DSP de Background Noise Study 3b es exactamente el mismo, lo que permite que ambas piezas se interpreten en el mismo evento. Lo particular de esta segunda pieza son las instrucciones que recibe el intérprete (un *intérprete bucal*), que analizaremos también.

3.1. Background Noise Study 3a.

El “intérprete electrónico” encuentra una fuente de ruido de fondo en la sala y coloca un micrófono cerca. “Ruido de fondo” hace referencia a todo sonido que

no es el que se quiere grabar y difundir por un medio. Otro micrófono se coloca o bien cerca de una fuente de ruido de fondo, o de una fuente potencial de ruido de fondo (un lugar en la sala donde es previsible que se produzca ruido de fondo a lo largo de la interpretación), o en el centro de la sala. El emplazamiento de los micrófonos tiene una importancia fundamental en el resultado sonoro. Dependiendo de la colocación del segundo micrófono el espectro sonoro resultante varía extraordinariamente. Pero también la forma de la obra varía: colocar el segundo micrófono cerca de una nueva fuente de ruido de fondo redundante en una mayor densidad sonora, dado que el material de inicio se duplica. La colocación cerca de una fuente de ruido potencial supone un mayor grado de indeterminación, puesto que resulta imposible saber si verdaderamente, y en qué medida, se producirá más ruido (en esta pieza, dicho sea de paso, las toses y silbidos del público son previsibles en un grado muy alto). Al colocar el micrófono en el centro se otorga a la sala una importancia mayor, puesto que este micrófono recoge predominantemente las frecuencias que son reflejadas por las paredes de la sala [\[12\]](#). El nivel de ambos micrófonos es amplificado hasta un nivel que hace audible este ruido de fondo. Ambas señales se introducen directamente en el módulo de DSP; es decir, en la sala no se oye ningún sonido que no haya pasado por el ordenador.

3.1.1. El flujo de DSP: Funciones de control



La figura muestra la realización del algoritmo (el *patch*), realizado en Kyma. El Kyma es un lenguaje (una plataforma) de programación gráfica, como puede verse en el ejemplo. Fue desarrollado por Carla Scaletti, como lenguaje de programación que va unido a una estación de sonido, el *Capybara*. Ambos constituyen una herramienta excelente para el tratamiento de sonido en tiempo real, tanto por la flexibilidad del Kyma como por la capacidad de procesamiento del *Capybara*, que llega a ser espectacular. Puede encontrarse más información en <http://www.symbolicsound.com/>.

La página 6 de la partitura contiene una representación gráfica de la extracción en vivo de las señales de control, señales que dirigen todo el sistema de producción de sonido. Es importante comprender que todos los resultados de

estas funciones son valores numéricos, es decir: aquí no se produce sonido, solamente series de ceros y unos que sirven para dirigir el sistema.

En primer lugar se regula la amplificación (ganancia) de la señal proveniente de ambos micrófonos (en diagramas electrónicos o de DSP suele representarse con una flecha). Ambas señales pasan a continuación por un filtro high-pass[13] que sólo permite pasar frecuencias que son más altas que una frecuencia determinada, la frecuencia de corte (*cut-off*) mientras que atenúa las frecuencias más bajas. El resultado es eliminar el ruido caótico que acompañan a toda señal[14]. Veremos que esta frecuencia de corte viene determinada por el sistema mismo: la frecuencia de corte viene determinada por el tiempo que permanece apagado un interruptor, que a su vez reacciona de manera proporcional a la densidad del sistema. Cuanto más tiempo permanece apagado este interruptor más alta es la frecuencia de corte, lo que tiene como consecuencia una pérdida de densidad del sonido. El sistema tiende entonces a diluirse, dado que tiende a admitir sólo frecuencias más altas (y por tanto inaudibles), y cada vez menos.

Ambas señales se mezclan en una mesa. La señal combinada es enviada entonces en dos vías paralelas:

1. La señal se retarda (*delay*) 20 segundos. El motivo es que toda señal que sale del ordenador necesita ser en primer lugar guardada en un *buffer*, desde donde puede ser leída. Para producir sonidos a tiempo real, el ordenador tiene primero que llenar un buffer mientras nuevos sonidos se van grabando. A esta señal se le asigna una curva de amplitud (curva AR)[15]. A continuación se integra esta curva. En este caso, la integración consiste en un medidor de amplitud (*Amp follower*), que calcula la amplitud de cada sampleado. Esta señal vuelve a retardarse para compensar la latencia[16], y 0.98[17] de la señal ya retardada vuelve a añadirse a la señal (*feedback*). El resultado de este proceso y su inverso se guardan en memoria como "lnAmp1" e "lnAmp2" respectivamente. Estos valores sirven para controlar el sistema.
2. La señal es primero integrada y después retardada 0.001 segundos (correspondiendo a la latencia del sistema). Se guarda el cuadrado del inverso de este valor como "lnAmp0" para utilizarlo como regulador del sistema.

"lnAmp0" cumple dos funciones de control:

1. Se compara a través de una función booleana con un valor límite (*threshold*). Este valor límite varía en cada interpretación. Mientras el valor de "lnAmp0" se mantiene por debajo de este valor límite el sistema continúa funcionando sin perturbación. En el momento en que el valor de "lnAmp0" es mayor que el valor límite, entonces se envía una señal que enciende un interruptor (*gate*). Este interruptor cumple a su vez dos funciones. La primera de ellas será analizada más tarde, dado que

corresponde al flujo de señal de audio. La segunda función cumple a su vez nuevamente funciones de control:

2.a. El interruptor lanza la curva de amplitud AR, es decir, cuando el interruptor se apaga se asigna una curva AR nueva al buffer. Esto significa que la nueva grabación se introduce gradualmente (*fade in*), y después de 20 segundos se resuelve gradualmente (*fade out*). El resultado es que el sonido no aparece de manera abrupta una vez que el sistema se ha apagado, sino que lo hace de manera continua.

2.b. El interruptor regula también una función Lag. Esta función mide el tiempo que el interruptor permanece apagado. La duración es posteriormente comparada mediante una función booleana. Si el valor es mayor que 20, es decir, cuando el sistema permanece apagado más de 20 segundos, entonces la función Lag se limita a dar como resultado 20. La lógica es sencilla: el sistema graba durante 20 segundos, y esto establece el límite del buffer. Si el sistema permanece más de 20 segundos sin recibir un nuevo sonido, entonces la función Lag hace que la flecha del buffer se deslice hasta el final del buffer, colocando al sistema en disposición de recibir nuevos sonidos. Cuando el sistema permanece apagado menos de 20 segundos, entonces la función Lag tiene como resultado el valor mismo.

El valor de la función Lag controla además otros parámetros. Al igual que el interruptor, la función Lag también controla parámetros de la señal de audio, como veremos después. Sin embargo, otras funciones de control son:

- a) La frecuencia de corte del filtro high-pass. Cuanto más tiempo permanece apagado el sistema, entonces más alta la frecuencia de corte. Cada repetición hace que el sonido pierda densidad.
- b) La posición del apuntador (*pointer*) del buffer que marca el lugar desde el cual se granulan las partículas (más sobre síntesis granular abajo). El apuntador sigue una función rampa, es decir, se desliza gradualmente a lo largo del buffer de izquierda a derecha y de abajo a arriba.

3.1.2 El flujo de DSP: señal de audio.

La página 7 de la partitura muestra el flujo de la señal de audio.

Los dos micrófonos colocados en la sala envían su señal, de nuevo, a través de sendos filtros high-pass. Estas señales siguen a continuación dos direcciones:

1. Se mezcla ambas señales. La señal resultante se multiplica por $1 - \ln \text{Amp}1$. Mirando atrás descubrimos que el valor de $\ln \text{Amp}1$ dependía directamente de la amplitud de la señal entrante, al ser multiplicada por $1 -$ el valor del integrador. Todo esto resulta en una amplificación de la señal de audio por su propia amplitud [\[18\]](#). Esta señal amplificada se guarda en un buffer (*memory write 1*).

2. Ambas señales se retardan y se asigna una curva AR a esta señal retardada. Antes de la salida se rebaja la ganancia de la señal, puesto que $\ln \text{Amp1}$ está aquí en proporción inversa a la señal. Es decir, cuanto más volumen muestre la señal original (el ruido de ambiente), entonces mayor es la amortiguación.

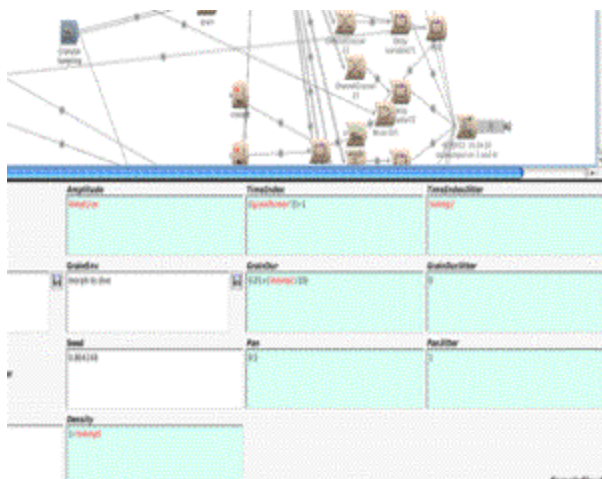
Más tarde veremos cómo la señal de audio es espacializada. Primero debemos analizar qué sucede con la señal guardada en *memory writer* 1. El tratamiento de DSP de este señal recorre los siguientes pasos:

1. Esta señal es leída por tres lectores de samplers, a velocidades diferentes, con lo que la frecuencia de la señal cambia. Cuanto más rápido se lee el sampler, más alta es la señal de salida [\[19\]](#). Cada sampler tiene un cociente diferente y todos los cocientes dependen de la función Lag, discutida más arriba. Cuanto más tiempo permanece el sistema apagado, entonces los lectores leen los samples a mayor velocidad, y por tanto también más altas son las frecuencias resultantes. Las señales de los tres samplers se retardan en $1/3$, $1/2$ y $2/3$ de la duración del buffer (20 segundos) respectivamente. Las tres señales son entonces amortiguadas en

proporción inversa a la amplitud original, para ser después difundidas por la habitación. Esto tiene como consecuencia que el sonido se proyecte por la habitación a intervalos regulares y en distintos puntos de la sala.

2. Se mezclan las tres señales y se las hace pasar por un filtro high-pass. La frecuencia de corte de este filtro depende en esta ocasión de $\ln \text{Amp0}$. Recordemos que $\ln \text{Amp0}$ era el cuadrado de 1 menos la amplitud original, lo que quiere decir que cuanto más alta la amplitud original, más baja (y en relación cuadrática) será el valor de la frecuencia de corte, con lo que se admitirán frecuencias más bajas a través del filtro. Esto tiene a su vez como consecuencia una ampliación del espectro sonoro. El sentido de este procedimiento es que ésta es la señal desde la que se extraen las partículas de síntesis granular. Una paleta de sonido más amplia permite más posibilidades de variación en la granulación, dado que las partículas contienen más frecuencias. La señal filtrada se guarda en un buffer (*memory write* 2), desde donde se procede a la granulación.

3.2. Granulación



La imagen muestra los parámetros que Di Scipio utiliza para la granulación. Antes de analizarlo describiremos brevemente el método de síntesis granular.

3.2.1. Síntesis granular

La síntesis granular fue desarrollada por Curtis Roads, continuando una propuesta de Xenakis (realizada manualmente en *Concret PH*), quien a su vez desarrolló musicalmente una idea de Denis Gabor, físico e ingeniero húngaro, ganador del Premio Nobel en 1971. La idea básica de Gabor es que el análisis del sonido podría seguir un curso análogo al de la física atómica, en el que la energía transmitida por un átomo dejaba de considerarse como un continuo, tomando su lugar los pequeños cuanta de energía que descubrió Max Planck. Gabor ideó una manera de analizar sonidos que no se basa en el análisis de Fourier (continuo), sino en pequeños paquetes de energía que contienen infinitud de frecuencias. La denominada "matriz de Gabor" representa la intensidad de la energía dependiendo de la ubicación en el espectro de cada partícula. La gran ventaja de este análisis consiste en hacer corresponder las dimensiones de espectro y amplitud con respecto al tiempo, que en el análisis de Fourier resultan separadas. Xenakis supo ver la inmensa ventaja de considerar un sonido no como una sucesión de ondas sinusoidales, sino como una aglomeración de pequeñas partículas sonoras, cada una de ellas conteniendo multitud de frecuencias, y en la que la energía de cada partícula podía ser calculada (siempre de manera limitada por el principio de indefinición) de manera autónoma. La realización de la síntesis corrió a cargo de Curtis Roads, aun cuando fue Barry Truax quien comenzó a utilizar métodos estadísticos para aglutinar partículas, lo que permite utilizarla a tiempo real.

La gran mayoría de implementaciones de síntesis granular [\[20\]](#) funcionan de la siguiente manera [\[21\]](#):

El sonido queda recogido en un buffer (siempre y cuando se trabaje con sonidos reales, claro está. La síntesis granular también puede trabajar con sonido puramente sintéticos). Este buffer podemos imaginarlo como una pantalla de ordenador, con una línea horizontal que marca el tiempo y una línea vertical que

designa el espectro. Esta pantalla es recorrida por un apuntador (pointer) de acuerdo a una función; utilizando una que ya conocemos, digamos por ejemplo, una función rampa. El apuntador va extrayendo samples del buffer, es decir, va recortando sonidos de frecuencias diferentes, de duración diferente, etc. Como todo sonido sintetizado, cada uno de estas partículas necesita una curva de amplitud, con lo que a cada partícula se le asigna una. La duración de cada partícula puede ser de incluso 20 milisegundos (inaudible), o de varios segundos. Lo más habitual es que las partículas se traten de manera estadística, formando *nubes* de partículas, dado que las partículas individuales son inaudibles. Los parámetros que el compositor puede determinar para formar sus sonidos son:

- a) Comienzo y final de la *nube*.
- b) Duración de las partículas.
- c) Densidad de la nube: número de partículas por segundo.
- d) Banda de frecuencia de la nube: el compositor determina frecuencias límite, dentro de las cuales se extraen las partículas.
- e) Curva de amplitud de las partículas.
- f) Forma de la onda dentro de la partícula.
- g) Dispersión en el espacio: dependiendo del número de canales con que cuente, el compositor puede distribuir las partículas por la sala.

Cada parámetro ejerce una influencia sobre los demás: la síntesis granular es un método orgánico. Como ejemplo, digamos que la elección de la curva de amplitud tiene un efecto audible en el espectro de la partícula, y por tanto en el color del sonido.

De nuevo, la discusión acerca de la síntesis granular es necesariamente breve. Desde luego na hacemos justicia a tan interesante tema.

3.2.2. Granulación en Audible Ecosystemics

Continuamos analizando el tratamiento de DI Scipio, en este caso el proceso de granulación.

Recordemos: el apuntador se desliza a lo largo de los 20 segundos del buffer, de abajo a arriba y de izquierda a derecha. Donde se halle el apuntador, de allí se extrae una partícula. El apuntador se desliza con una velocidad determinada por una frecuencia, que es la inversa del valor de la función Lag multiplicado por 40 (para situarla dentro del ámbito). Es decir, cuanto más tiempo permanece apagado el sistema, entonces más lentamente se mueve el apuntador. Esto tiene

como consecuencia que las partículas se extraen del mismo material sonoro, lo que significa una reducción de la variación sonora.

El Jitter determina una fluctuación probabilística del apuntador. Determina dos valores a izquierda y derecha del apuntador, dentro de los cuales fluctúa rápidamente. El Jitter viene determinado por el valor de $\ln \text{Amp}2$, lo que corresponde a la amplitud original. Es decir, cuanto más volumen muestra el material, mayor es el ámbito dentro del cual fluctúa el apuntador. Un Jitter mayor trae como consecuencia una realización más "realista" de la partícula, además de atraer una mayor indeterminación. Puede suceder que la partícula sea extraída con el apuntador leyendo los samples *al revés*, de derecha a izquierda. Es un efecto sonoro muy particular.

La duración de la partícula también depende de manera directamente proporcional a $\ln \text{Amp}2$. Di Scipio fija una duración mínima (0.01 seg), a la que se suma el valor de $\ln \text{Amp}2/10$. La variedad en la duración de la partícula tiene como consecuencia una gran variedad de percepción sonora. Partículas muy cortas se perciben casi como simples cambios de presión del aire (como cuando el oído hace "pop" con un cambio de presión), mientras que duraciones más largas sí permiten una percepción espectral.

La densidad de la nube de partículas viene determinada por $1 - \ln \text{Amp}0$. Este último valor, recordemos, era igual a $1 - \text{Amplitud}^2$ (cuadrado). Lo que quiere decir que la amplitud original determina la densidad de manera inversamente proporcional al cuadrado. Cambios en la densidad tienen efectos dramáticos en el espectro sonoro resultante. La densidad hace referencia al número de partículas por segundo. Un número elevado de partículas por segundo produce un sonido continuo, como una masa sonora (*Klangteppich*) donde las variaciones de color se perciben de manera muy rápida e ininterrumpida, y por consiguiente un sonido rico y lleno. Un número reducido es percibido como leves erupciones de sonido (la analogía con el "pop" del oído también es válida aquí).

A cada partícula se le asigna una curva de amplitud. La amplitud de la partícula determina el volumen con el se percibe la partícula, pero también, como dijimos, tiene consecuencias sobre el espectro de la partícula. La amplitud de las partículas se determina mediante dos parámetros:

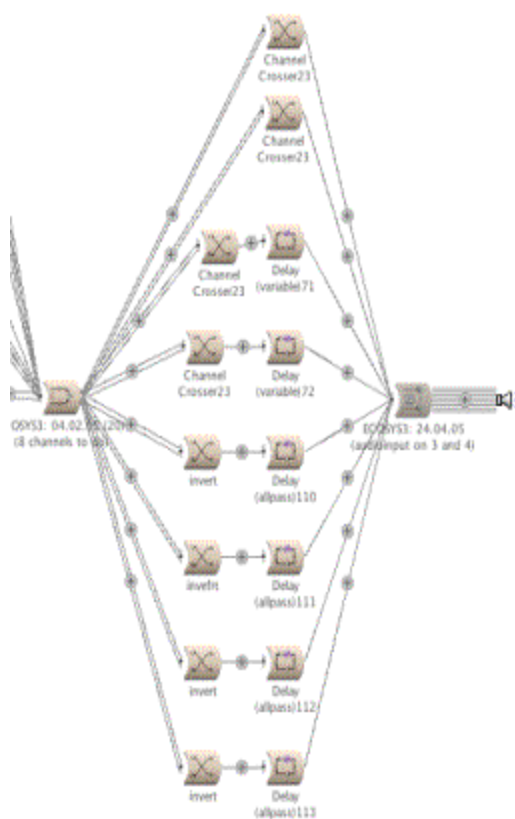
1. $\ln \text{Amp}0$, es decir, la amplitud del material sonoro original.
2. OutAmpGrain resulta un ejemplo interesante de cómo Di Scipio ha llevado a cabo la interacción entre el sistema y su entorno. El sistema analiza su entorno, reacciona, y el resultado de esa reacción vuelve a introducirse en el sistema. El sistema vuelve entonces a analizar el entorno, pero ahora su entorno refleja la reacción anterior del sistema, la reacción se ha integrado en el sistema. La amplitud de la primera nube de partículas la proporciona el intérprete electrónico (a través de un interface). Esta amplitud es integrada por el sistema mediante un medidor de señal que depende de la amplitud, retardada para evitar la latencia y guardada en OutAmpGrain . Este valor vuelve entonces a añadirse a la

mencionada amplitud, pero después de haber sido restado de 1, con lo que se comporta de manera inversa, se comporta como una compensación a la amplitud: cuanto mayor es la amplitud, mayor es la amortiguación. Este tipo de proporciones inversas producen sistemas muy dinámicos. Resulta interesante observar cómo Di Scipio ha escogido la manipulación de los parámetros: aquellos parámetros que muestran un comportamiento más inestable son aquellos que más contribuyen a otorgar variedad al resultado sonoro. De ahí que se refiera a sí mismo con "un compositor de sonidos"[\[22\]](#).

3.3. Espacialización

La página 8 de la partitura contiene el flujo de espacialización de la señal.

La sección del patch que se refiere a la espacialización es:



La espacialización muestra dos características muy habituales en las obras de Di Scipio:

1. Los altavoces están dirigidos contra las paredes. El sonido que se encuentra en la sala nunca es oído directamente por el público. El material está compuesto tan sólo por sonido reflejado, con lo que la sala adquiere una importancia fundamental: la sala determina qué frecuencias se escuchan de manera predominante.

2. La salida de la señal a cada altavoz es retardada por la distancia con respecto a cada altavoz partido por 344 (velocidad de la onda sonora en el aire) multiplicado por el valor de la función Lag. Este retardo tiene dos motivos:
 - a) Retardar una señal por la distancia dividida por la velocidad de la onda tiene como consecuencia que el sonido surge simultáneamente desde los diferentes altavoces. El sonido no se percibe simultáneamente, por supuesto, dado que eso depende de la posición del oyente. La aparición es no obstante simultánea en distintos altavoces, la señal permanece constante por canal. Esta solución ofrece también la posibilidad de hacer aparecer el sonido de manera escalonada, de ahí que Di Scipio multiplique la velocidad de la onda por 2 y 3. Esto se conoce como división virtual de la sala en hileras.
 - b) La señal se retarda asimismo en $100 \cdot 1 - (\text{Lag}/20\text{seg})$. Esto tiene como consecuencia que la señal se retarde un número determinado de samples, número que es proporcional al tiempo que el sistema permanece apagado. Con esto se asegura que los lectores de samples encuentran algo en el buffer que leer.

3.4. Background Noise Study 3b

"(Background Noise Study 3b) utiliza el mismo concepto cibernético situado en entorno diferente; a saber: en el conducto vocal del intérprete. El resultado sonoro tenemos todos aquellos pequeños sonidos producidos involuntariamente en la boca y en la faringe. Ocasionalmente se traen conscientemente a un primer plano leves impulsos y otras pequeñas ondas que se producen involuntariamente"[\[23\]](#).

La partitura de Background Noise Study 3b consiste en una serie de posiciones bucales, transiciones entre ellas, pausas y diferentes técnicas de lengua y garganta. A cada una de estas posiciones se le asigna una duración temporal. La intérprete bucal sostiene en la mano un micrófono muy pequeño, se lo introduce en la boca e interpreta la pieza. El micrófono se amplifica de manera masiva, con lo que recoge y hace audible los mínimos sonidos que se producen al contraer y distender los músculos de la boca y del conducto bucal.

Durante las pausas prescritas en la partitura, la intérprete se saca el micrófono de la boca. Las pausas sirven para que la intérprete se recupere físicamente, humedecerse los labios, respirar con normalidad, tragar saliva, etc.

Lo inhabitual de la interpretación es que la intérprete no produce ningún sonido de manera consciente. De hecho, se le exige permanecer tan silenciosa y relajada como pueda. La interpretación de la obra supone un gran reto, dado que a la intérprete se le hacen conscientes muchos aspectos de la técnica interpretativa vocal sobre los cuales quizá antes no había pensado demasiado.

3.4.1. Interacción sistémica

La intérprete recibe una serie de instrucciones que son una excepción a la indicación general de "silencio". Son sonidos producidos con la boca o la garganta. La intérprete debe mover la mandíbula, producir impulsos leves con la glotis, entrechocar los dientes, chasquear la lengua, mover la lengua entre las paredes dentales interiores, etc.

Estas indicaciones son la única manera de que dispone la intérprete para regular el sistema. La intérprete debe oír el sonido, que ha producido ella misma y ha sido retardado 20 segundos, y reaccionar a él. Esta reacción es totalmente libre: no obstante, debe decidir lo que quiere escuchar a continuación. Las consecuencias de su reacción, no obstante, no son lineales, y por tanto imprevisibles. Di Scipio ofrece a la intérprete dos posibilidades:

1. Puede confiar en su percepción, escuchar los resultados de sus acciones e intentar reaccionar o adaptarse a ellos.
2. Puede intentar dirigir el sistema de marea consciente, controlando todo lo posible las causas de cada sonido.

Existe la posibilidad de que el sistema escape al control de la intérprete. Di Scipio establece una serie de "reglas de emergencia" que están a disposición de la intérprete cuando crea que no tiene el control del sistema. Puede, por ejemplo, introducirse profundamente el micrófono en la boca y permanecer en silencio. Todas las medidas conducen a una saturación del sistema, dado que el sistema reacciona por compensación ante sonidos fuertes o muy diferenciados. Una saturación apaga temporalmente el sistema, con lo que la interpretación puede continuar.

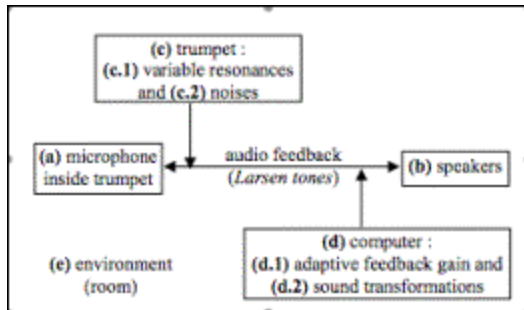
"Se utiliza la violencia para acabar con la acumulación de sonido". La intérprete debe ser consciente de que la cantidad de violencia adquiere una importancia grande, puesto que las condiciones de inicio dependen proporcionalmente de los últimos sonidos captados. La intérprete debe afrontar las consecuencias de su violencia: es posible que su propia violencia se vuelva contra ella, ¡y por duplicado!"

4. Modes of Interference

La filosofía sobre la que se basa "Modes of Interference" es similar a la "Audible Ecosystemics 3b". En esta ocasión, no obstante, el objetivo del material sonoro no es el de hacer audible sonido de ambiente o levísimos fenómenos sonoros producidos por la contracción y relajación de músculos bucales. El sistema autónomo de "Modes of Interference" establece una recursión que se dirige al establecimiento de *feedback*, y más concretamente de tonos Larsen. El sistema produce un bucle entre el micrófono y altavoces que rápidamente incurre en el conocido (y a evitar por todos los medios por cualquier técnico de sonido que se precie) fenómeno de *feedback* (retroalimentación)[\[24\]](#). El intérprete, en este

caso un trompetista, puede intervenir en este proceso, y hasta cierto punto dirigirlo.

Una representación del sistema es [\[25\]](#):



4.1. Tratamiento del sonido

El patch de manipulación de sonido cumple fundamentalmente dos funciones:

1. Mantener el sistema dinámicamente activo. Esto se consigue principalmente mediante medidas de compensación, adaptando el sistema a la saturación constante hacia la que tiende el bucle de retroalimentación.
2. El patch también transforma todos los sonidos que se producen durante el bucle, incluyendo los levísimos sonidos generados por el trompetista. Estas transformaciones vuelven a integrarse al bucle, con lo que las transformaciones también afectan al sonido.

4.1.1. Extracción de parámetros de control

La mayor parte de las funciones de extracción de control del sistema a partir del sonido mismo resultan ya conocidas de "Audible Ecosystemics". El patch está realizado en PureData [\[26\]](#), y, salvo en la diferencia en las funciones construidas en el software, el flujo es muy parecido a lo que ya conocemos.

La función principal es un medidor de amplitud. El inverso de este valor es *AmpScaler*, que es la función principal de adaptación del sistema. Al ser el inverso de la amplitud de la señal, *AmpScaler* actúa siempre de manera inversamente proporcional a la amplitud. El valor de esta función sirve para atenuar de manera continua la señal, evitando así que el sistema se sature.

En Modes of Interference también hay granulación, de manera similar a lo ya visto.

4.2. Interpretación

[La partitura de "Modes of Interference"](#) consiste en una explicación del patch y en una serie de instrucciones para el intérprete. El intérprete no debe producir

ningún sonido propio del instrumento; su labor es más bien controlar el bucle de retroalimentación por medio de dos estrategias básicas:

1. Variar la resonancia del tubo. El micrófono es introducido en el tubo por la campana a gran profundidad, situándolo a unos tres centímetros de la boquilla. El intérprete varía la longitud del tubo mediante combinaciones de pistones. La longitud del tubo cambia la resonancia dentro de la cavidad, y con ello cambia el material que es captado por el micrófono.
2. Introducir sonidos en el bucle por medio de levísimos ruidos, como *tapping* sobre la boquilla, chasquidos de lengua, y, ocasionalmente, soplar aire en la boquilla. Estos sonidos forman parte del proceso de transformación del material, dado que introducen cambios en el bucle de sonido, cambiando así el estado del sistema. El resultado son cambios en las frecuencias Larsen y en las dinámicas del sistema.

La interpretación del trompetista cumple por tanto las funciones de ser fuente de material sonoro y de regulador del sistema.

5. CONCLUSIONES

Afrontar una obra algorítmica resulta siempre complicado, aun más cuando lo hacemos por primera vez. El oyente se pregunta, a veces con perplejidad: ¿dónde está el alma? El intérprete se pregunta, con perplejidad aún mayor: ¿dónde está la música? El compositor se pregunta, muchas veces con una mueca: ¿dónde está el compositor?

Es imposible contestar estas preguntas (¿es imprescindible contestar estas preguntas?). El pensamiento compositivo fascina a unos y deja indiferente a otros; la técnica interesa a unos y aburre a otros; el sonido mismo conmueve a unos y aturde a otros. En ese sentido, las composiciones algorítmicas no son muy diferentes de cualquier otra obra, de cualquier otro género. Algunos compositores sencillamente sienten la necesidad de alejar todo lo posible todos aquellos elementos que no sean la obra misma; quizás con ello crean que pueden concentrarse únicamente en lo que a ellos más obsesiona. En el caso de la música computacional resulta muy sencillo que el compositor colecciona una serie de gestos técnicos (en este caso, de implementaciones) y las vaya distribuyendo en el tiempo. Lo interesante de la música computacional surge precisamente cuando esto se deja a un lado. Cuando el compositor puede decir: "estas son mis herramientas: estas de producción, aquellas de organización". En el momento en que podemos ver con toda claridad cuáles son nuestras herramientas, entonces podemos comprender cuál es la mejor forma de utilizarlas. Di Scipio enumera sus herramientas:

1. Funciones de compensación: a un aumento de un parámetro reacciona el sistema con una reducción de otro parámetro correlativo perceptualmente. Por ejemplo: una reducción de la amplitud provoca un aumento de la densidad.

2. Función de seguimiento: Un parámetro perceptual sigue a otro, hasta que finalmente le alcanza.
3. Redundancia: un parámetro es apoyado, en lugar de contrastado.
4. Simultaneidad: un parámetro perceptual introduce otro, con el que concurre.

Estas herramientas se utilizan de acuerdo a dos estrategias básicas:

1. *Omeostasis*: el sistema muestra tendencia a incurrir o alejarse poco de un estado determinado.
2. *Omeoresis*: el sistema muestra tendencia a incurrir en estados muy variados.

El compositor que conoce su obra es capaz de decidir cuándo utilizar una estrategia y cuándo la otra. Pero, ¿y el oyente?

Imaginemos:

Tú estás sentado en una sala. Te pones cómodo, buscas una posición ideal, te mueves, produces sonidos, inspiras, expiras, te preparas, escuchas...

Y no oyes nada.

¡Pero ahí hay algo! El simple hecho de que *estés* ahí produce sonidos, sonidos que acompañan cada momento de nuestra vida. Sonidos que ya ni siquiera oímos, porque nunca nos abandonan. Son ruido de fondo.

Imaginemos:

Un compositor está sentado en su estudio. Reflexiona sobre sonidos, los oye con su oído interno. Se da cuenta de que los sonidos no son estáticos, que se mueven, que se relacionan entre sí, que construyen una cadena de acciones y reacciones. El compositor intenta reconstruir la lógica de esta cadena. Comprende que para entender esta cadena debe integrarse a sí mismo en ella. Construye un sistema.

El compositor diseña un sistema: determina cómo se comportará el sistema ante cambios en las condiciones externas; cómo el sistema se adaptará a estos cambios; cuánto cambiará el sistema mismo con estas reacciones; etc. Nosotros, el público, somos una parte de este sistema. Nosotros también cambiamos, nosotros debemos adaptarnos también a los cambios en nuestro entorno, a las demás partes del sistema; debemos interrelacionarnos con el sistema.

"La música es una huella que la composición deja a su paso".

Notas:

Di Scipio, A. 1990: Composition by exploration of non-linear dynamic systems. Proceedings of the ICMC, Glasgow, 1990.

Di Scipio, A. 1996: Interactive composing with granular time-shifting of sound. Internet Proceedings of the Journee d'Informatique Musicale, 1996.

Di Scipio, A. 1997a: Towards a critical theory of (music) technology. Computer music and subversive rationalization. Proceedings of the 1997 Int'l Computer Music Conference, Thessaloniki. Online Artikel.

Di Scipio, A. 1997b: Interpreting music technology: from Heidegger to subversive rationalization. Sonus, Vol.18 nº 1. Fall 1997.

Di Scipio, A. 2000: Tecnologia dell'esperienza musicale del novecento. Rivista Italiana di Musicologia, vol. XXXV-NN.1-2.

Di Scipio, A. 2003 : Sound is the interface' Sketches of a constructivistic ecosystemic view of interactive signal processing. 14th CIM conference, Firenze.

Di Scipio, A. 2005: Klangstaub: Die Notwendigkeit einer aesthetischen Orientierungslosigkeit. Positionen, 64:45-48. Berlin.

Di Scipio, A. 2006: Using PD for Live Interaction in Sound. An Exploratory Approach. Conservatory of Naples, Italy.

Di Scipio, A. & Prignano, I. 1996.: Synthesis by functional iteration: a revitalization of nonstandard synthesis. Journal of New Music Research, Vol. 25.

Feenberg, A. 1992.: Subversive rationalization: technology, power and democracy. in Feenberg, Hannay eds. Technology and the politics of knowledge, Indiana University Press.

Feenberg, A. 1995.: Alternative modernity: the technical turn in philosophy and social theory. University of California Press.

Feenberg, A. 1996: Marcuse or Habermas: two critiques of technology. Inquiry 39.

Feenberg, A. 2004: Questioning technology. Routledge, New York.

Heidegger, M. 1953: Die Frage nach der Technik. Artículo online.

Maturana, H. y Varela F.G., 1994: De Máquinas y Seres Vivos. Editorial Lumen, Buenos Aires, Argentina.

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

Roads, C. 2001: *Microsound*. MIT Press, Cambridge.

Ross Ashby, W. 1957.: *An Introduction to cybernetics*. Chapman & Hall, London.

Smith, J.O., 2005: *Introduction to digital filters*.
<http://ccrma.stanford.edu/~jos/filters05/>

[1] No porque sea imposible encontrarlos en estas dos piezas: por supuesto que es posible, ¿podría acaso ser de otra forma? El pensamiento teórico, al menos en el caso de Di Scipio, es generador de ideas musicales, y viceversa. Toda la obra de un artista que reflexiona está necesariamente impregnada de esas reflexiones, o de esa manera de hacer reflexiones, esa *Weltanschauung*. En el caso de Di Scipio esto es extremadamente cierto.

[2] *Vid.* En cuanto a los aspectos técnicos:

Di Scipio 1990, 1996; *Di Scipio & Prignano*, 1996.

En cuanto a los aspectos teóricos: *Di Scipio* 1997a, 1997b (esta revierte a *Heidegger* 1953), 2000, 2005; y *Feenberg* 1992, 1995, 1996, 2004.

[3] Herbert Brün, citado en *Di Scipio* 2005; (traducción del autor).

[4] Maturana & Varela, 1994.

[5] *ibidem*.

[6] „al escribir yo en el pizarrón que los ADN participaban en la síntesis de las proteínas, y que éstas participaban en la síntesis de los ADN, (...) me di cuenta de que era esa circularidad la dinámica productiva molecular constitutivo de lo vivo.“ *ibidem*, pág. 14.

[7] Seguimos aquí a Ross Ashby, 1965.

[8] Ejemplos extraídos de *DI Scipio*, 2003.

[9] *Di Scipio*, 2003.

[10] *Ibid.*

[11] *Di Scipio*, 2005.

[12] Toda sala refleja un determinado número de frecuencias y absorbe otras. Qué frecuencias se reflejan y cuáles se absorben depende de muchos factores: tamaño de la sala, contenido de la sala, reflexión de las superficies presentes, materiales de las paredes, suelo, las personas que se encuentran en la sala,

etc. Las frecuencias reflejadas varían también, claro, dependiendo del sonido emitido.

[13] Un filtro digital es un algoritmo que transforma la señal entrante por medio de diferentes procesos. *Vid* Smith, J.O., 2005.

[14] Este ruido no es necesariamente ruido „sonoro“ (aunque se percibe como ruido blanco una vez que ha pasado por un transformador de digital a analógico), sino ruido de cuantización, que suele acompañar al proceso de muestreo.

[15] „Attack-Release“: una curva de amplitud que determina el ataque y la resolución de un sonido.

[16] La latencia (en sentido digital) es el tiempo que tarda el ordenador en alcanzar el siguiente resultado. Se corresponde al reloj interno del ordenador.

[17] Latencia del sistema (reloj interno)=0.001 seg. Latencia del primer delay=20seg. Nueva latencia= $S/V=0.02$ seg.

[18] $x*1-(1-x)$

[19] Este era el método habitual de síntesis en los primeros sintetizadores para tocar diferentes frecuencias; *vid* Roads, 1996.

[20] Es posible encontrar una lista completa de implementaciones de síntesis granular, además de links a aquellas que son gratuitas en <http://aliasfrecuencias.org/son/text/granular.html>

[21] Una explicación que no incurre en la obligada brevedad de nuestro tratamiento en el propio Roads, 1996, y sobre todo 2001.

[22] Conversación personal.

[23] Di Scipio, notas al programa de estreno, Berlín, Julio 2005.

[24] Otros ejemplos famosos de obras que utilizan feedback como material sonoro son las de Cage, Lucier o Lupone.

[25] Di Scipio, 2006.

[26] PureData (o PD) es un lenguaje de programación gráfico muy similar al Max/MSP. De hecho su creador, Miller Puckette, es uno de los dos creadores de éste. Una ventaja evidente de PD es que es completamente abierto y gratuito. El patch que aquí presentamos puede abrirse, tan sólo hay que bajarse e instalar el programa. Ver <http://www.puredata.org>.