



FACULTAD DE ARTES Y CIENCIAS MUSICALES

CENTRO DE ESTUDIOS ELECTROACÚSTICOS

PROCESAMIENTO EN TIEMPO REAL EN LA OBRA DE LUIGI NONO

Pablo Cetta

INTRODUCCIÓN

El presente artículo trata principalmente aspectos técnicos relativos a la producción y realización de una de las obras más importantes del último período creativo del compositor italiano Luigi Nono (1924-1990) : *Quando stanno morendo (Diario Polaco Nro. 2)*, escrita en 1982 para dos sopranos, mezzosoprano, contralto, flauta baja, violoncelo y procesamiento electrónico del sonido en tiempo real.

La ejecución de la versión original involucra gran cantidad de dispositivos electrónicos (hardware) y personal técnico para lograr el procesamiento del sonido de voces e instrumentos en vivo. A esto se suma la dificultad de obtener tal equipamiento. Pasados veinte años del estreno, creímos que con el desarrollo tecnológico actual era posible concentrar todos esos dispositivos y su operación en un único programa de computación autocontrolado.

La programación del sistema fue realizada en el Centro de Estudios Electroacústicos de la Facultad de Artes y Ciencias Musicales de la U.C.A., sobre una plataforma Macintosh G4 de 733 MHz, usando el software de procesamiento en tiempo real Max-MSP, desarrollado por Miller Puckette en el IRCAM.

La obra fue estrenada en nuestro país por el *Ensamble Dédalus* bajo la dirección de Eduardo Moguillansky, en un concierto realizado el 7 de Agosto de 2002 en el Auditorio del Centro Cultural Recoleta, en coproducción con el Laboratorio de Investigación y Producción Musical (LIPM). En el mismo concierto se interpretaron *¿Dónde estás hermano?* (1982) para dos sopranos, mezzosoprano y contralto, reescritura de parte de la obra anterior, y *...sofferte onde serene...* (1976) para piano y sonidos grabados.

Las obras fueron interpretadas por Mariana Nigro y Ana Santorelli (sopranos), Selene Lara (mezzosoprano), Beatriz Moruja (contralto), Patricia García (flauta baja), Martín Devoto (violoncelo), Florencia Cima (recitante), Blas Gonzalez (teclado MIDI) y Nora García como pianista invitada.

Actualmente, trabajamos en la preparación del *Omaggio a György Kurtág* de Luigi Nono, compuesta entre 1983 y 1986, obra a la que nos referiremos en último término.

UNIDADES DE PROCESAMIENTO

Quando stanno morendo recurre a la transformación del sonido de voces e instrumentos mediante 12 dispositivos, interconectados en 23 configuraciones distintas (programas) que alternan a lo largo de sus tres movimientos. La selección de cada programa y los parámetros de los efectos es indicada con precisión en la partitura.

Los procesos requeridos son los que a continuación se enumeran.

- 2 líneas de retardo (delays)
- 2 cámaras de reverberación
- 2 armonizadores
- 1 dispositivo de espacialización (Halaphon)
- 2 bancos de filtros pasabanda
- 1 dispositivo de retrogradación
- 1 vocoder
- 1 seguidor de envolvente

El sonido se distribuye a través de diez parlantes, que se ubican alrededor de la sala en un círculo exterior (parlantes 1 al 4) y otro interior (parlantes 5 al 8). Los parlante 9 y 10 se colocan alejados del público (detrás del escenario o bien cerca del techo) con el propósito de crear una sensación de distancia.

DESCRIPCIÓN DE LOS EFECTOS E IMPLEMENTACIÓN EN MAX-MSP

Retardo

Se emplean dos líneas con tiempos de retardo fijos de 2 y 5 segundos, con realimentación.

Este proceso, que tradicionalmente se aplica para imitar uno o varios ecos, es utilizado aquí con un criterio diferente: persigue la generación de microintervalos simultáneos entre el sonido directo y el procesado, a partir de un tiempo de retardo relativamente largo sobre eventos de tipo continuante. El efecto resulta enfatizado mediante la indicación “Mikro“, que pone el autor en su partitura a las cantantes, para que ejecuten desviaciones en la altura de la voz que pueden llegar hasta el cuarto de tono.

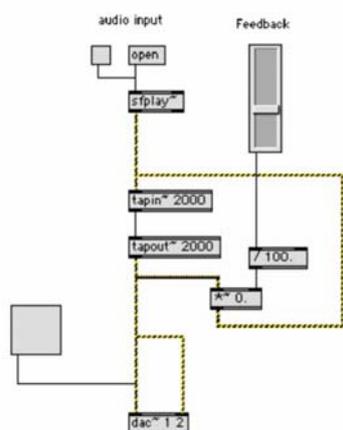


Figura 1

Vemos en la figura 1 la programación en Max-MSP de la línea de retardo de 2000 milisegundos, con realimentación. El almacenamiento de la señal entrante y su posterior reproducción se realizan con las unidades *tapin* y *tapout*. La señal retardada reingresa escalada por un valor de realimentación, establecido con el potenciómetro *Feedback*.

Reverberación

La reverberación artificial es empleada de dos modos diferentes. En el primer caso como *refuerzo*, si la acústica de la sala de conciertos lo requiere, con un tiempo de 2 a 3 segundos. En el segundo caso como *tratamiento* del sonido, integrada compositivamente en los programas 17, 18, 19 y 23 (con un tiempo de 2 a 3 segundos) y en el programa 22 (7 a 10 segundos).

Para esta obra se empleó una unidad *tap.verb* desarrollada por Prairie Intermedia. Cuenta con controles para especificar el tiempo de reverberación (t_{60}), la relación sonido directo-reverberado y la absorción de agudos del ambiente.

Armonización

Se utilizan dos transpositores de altura variable. En el programa 17 se aplican a la flauta y al violoncelo, variando el nivel de transposición hacia la octava inferior (-1200 cents), pero no linealmente sino con una desviación aleatoria. En el programa 19 afectan a la segunda soprano y a la mezzosoprano con una transposición fija de +1129 y +600 cents respectivamente, mientras que en el programa 22 ambos transpositores modifican la altura del violoncelo en -1200 y -2048 cents. En la ejecución original de la obra se utilizó el hardware Publison Infernal Machine 90 Stereo Audio Computer.

La transposición en altura de un sonido, sin que varíe su duración, se obtiene aquí mediante un proceso denominado *phase vocoding*. El mismo, también se aplica para transformar la duración de un evento sonoro sin que se modifique su altura. Sabemos que en los medios analógicos una transposición hacia el grave involucra un estiramiento del sonido y viceversa (en la reproducción de una cinta grabada cuando se disminuye la velocidad, por ejemplo). Los medios digitales, en cambio, permiten controlar ambos parámetros de forma independiente.

En un *phase vocoder*, la señal de entrada es separada en diversos canales, cada uno de los cuales representa una determinada franja del espectro audible. Para cada canal se detecta la componente más significativa y se extraen periódicamente su frecuencia, amplitud y fase como puntos de una envolvente. Esa información alimenta un banco de osciladores sinusoidales que resintetiza el espectro. Los datos de cada canal pueden ser almacenados, ocupando un espacio menor que el de la señal original, o también transmitidos a gran velocidad, de ahí su utilización en telefonía como codificador de la voz (*vocoder = voice-encoder*).

La función de transposición se logra modificando los valores de frecuencia para cada oscilador. Multiplicando esos valores por 0.5, por ejemplo, obtenemos una transposición a la octava inferior sin que la duración del evento sonoro se vea alterada en lo más mínimo.

La unidad *pvoc* utilizada, desarrollada por Eric Lyon, posee argumentos que especifican las frecuencias que determinan el registro de operación del banco de filtros. Mientras más amplia sea la banda de análisis, mayor será el tiempo de procesamiento, y por consiguiente, el consumo de recursos de la computadora. Otros argumentos indican el índice de transposición y el umbral a partir del cual una componente debe ser considerada significativa en el espectro, de acuerdo a su amplitud relativa.

Retrogradación

La realización de un proceso de retrogradación en tiempo real presenta como problema la imposibilidad de conocer el futuro, para comenzar por él y dirigirnos hacia el pasado. Por lo tanto, sólo es posible por pasos, fragmentando la señal de audio en un pequeño bloque, retrogradándolo y poniéndolo a continuación del bloque anterior. Nono estableció la duración de cada bloque en 1.5 segundos, y para este proceso utilizó un dispositivo Publison DHM 89 B2.

La implementación en Max-MSP, utiliza dos *buffers* (unidades de almacenamiento de la señal) de 1600 milisegundos cada uno. Cuando la señal ingresa, se graba un segundo y medio en el primer *buffer*. A continuación se comienza a grabar en el segundo de ellos, y mientras esto sucede, se reproduce el contenido del primero de atrás para adelante (retrogradado). Posteriormente se reproduce el segundo mientras el primero graba, y así sucesivamente. Para evitar discontinuidades en la forma de onda resultante se realiza un solapamiento y un *crossfade* de 100 milisegundos entre las señales reproducidas por ambos *buffers*.

Sobre la parte derecha de la figura 2 se puede observar la unidad *metro*, que sirve de reloj del proceso, y debajo, las dos unidades que graban de forma alterna. A la izquierda, las dos unidades de reproducción, y lo necesario para procesar el *crossfade* entre ambos registros.

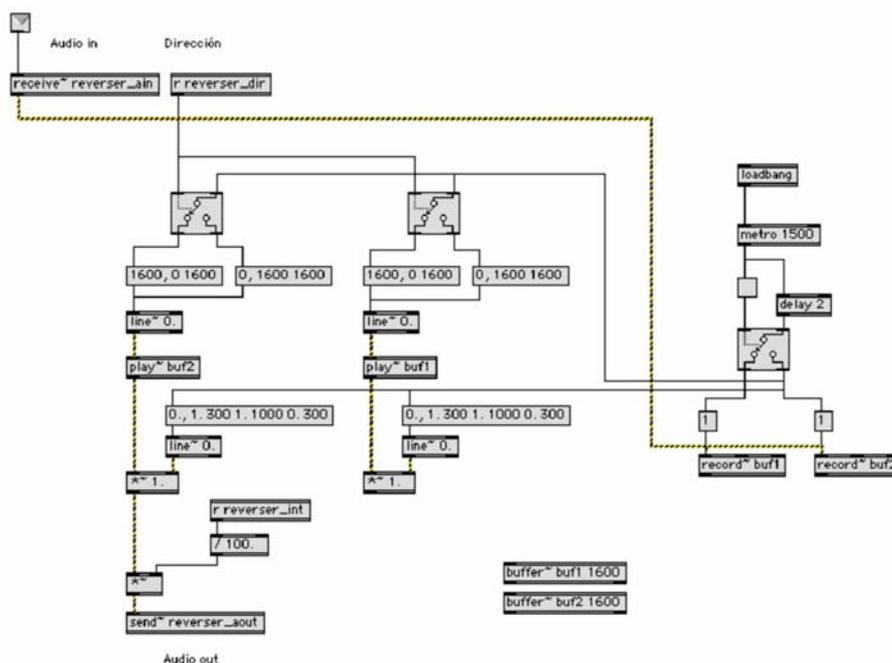


Figura 2

La retrogradación por pasos es empleada en el segundo movimiento de la obra afectando el recitado de la contralto. El compositor pide una alternancia *ad libitum* entre el sonido directo y el retrogradado.

Espacialización

La versión original recurre a un hardware de localización espacial del sonido denominado Halaphon, que produce trayectorias circulares de una señal de audio controlando la intensidad relativa de un grupo de parlantes. El período (tiempo para completar un giro de 360°) puede asumir un valor constante, o bien variar de acuerdo a la intensidad de otra señal de audio que ingresa por algún micrófono.

En la obra se emplean cuatro programas, que a continuación se describen:

Programa 1 (H1): giro entre parlantes 1, 2, 3 y 4. El período es controlado por la intensidad del soplo del/la intérprete de flauta. Cuando la dinámica es fuerte o fortísimo, el período es de aproximadamente 0.4 segundos; en ausencia de soplo, el sonido debe permanecer estático.

Programa 2 (H2): dos giros simultáneos. Uno entre los parlantes 4, 3, 2 y 1 (en sentido antihorario) con un período de 5 segundos. El otro entre parlantes 1, 2, 3 y 4 (en sentido horario) con un período de 8 segundos.

Programa 3 (H3): dos giros simultáneos. Uno entre los parlantes 5, 4, 3, 6, 2 y 1 (en sentido antihorario) con un período de 7 segundos. En nuestra versión fue simplificado a los cuatro parlantes del círculo exterior (1, 2, 3 y 4). El otro giro entre los parlantes 1, 2, 6, 3, 4 y 5 (en sentido horario) con un período de 10 segundos, también simplificado al arreglo exterior.

Programa 4 (H4): dos giros simultáneos. Uno entre los parlantes 1, 2, 3 y 4 (en sentido horario) con un período de 8 segundos. El otro entre parlantes 5, 6, 7 y 8 (en sentido horario) con un período de 10 segundos.

Nuestra versión en Max-MSP utiliza sólo ocho canales (número máximo de canales de salida de la placa de audio). Las señales de los canales 9 y 10 fueron simuladas a través de 3 y 4, con reverberación para dar la sensación de lejanía.

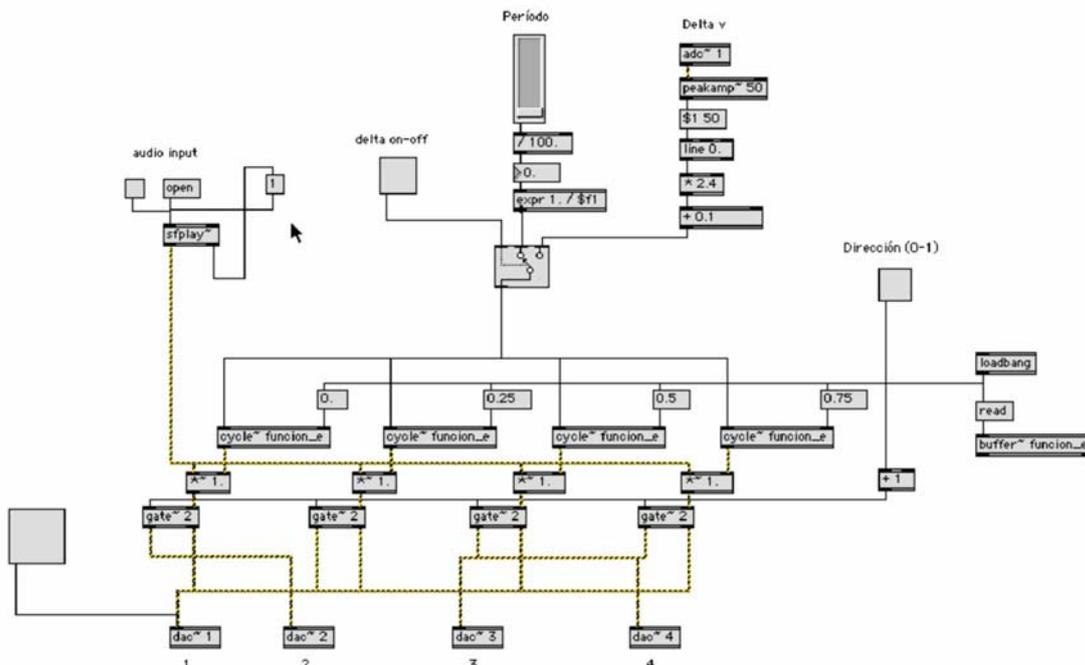


Figura 3

La figura 3 representa a uno de los dos espacializadores empleado. El período se puede controlar por medio de un valor fijo o a través de un seguidor de envolvente. En el segundo caso, el seguidor toma muestras de amplitud de la señal que ingresa por un micrófono -cada 50 ms- y construye rampas entre estos puntos; los valores son luego escalados y determinan el período de giro para cada instante.

La ganancia de cada uno de los cuatro parlantes de un círculo –interior o exterior- involucrados en el giro virtual de la fuente sonora se obtiene mediante:

$$g_n(\theta) = \cos(\theta - \theta_n) \quad \text{si } |\theta - \theta_n| < 90^\circ \quad \text{ó bien } 0$$

donde g_n es la ganancia del parlante n , θ es el ángulo de posicionamiento de la fuente y θ_n es el ángulo de ubicación del parlante.

Para evitar la realización de estos cálculos se implementa la función que se observa en la figura 4. Los valores de la función –cuya amplitud varía entre 0 y 1- determinan la amplitud relativa que debe poseer cada parlante para generar un movimiento circular del sonido.

En la parte inferior del gráfico observamos cuatro osciladores (*cycle*) que leen cíclicamente la misma función previamente almacenada, pero a distintas fases iniciales. Finalmente, las unidades *gate* permiten configurar las salidas de los osciladores y determinar el sentido de giro –horario o antihorario.

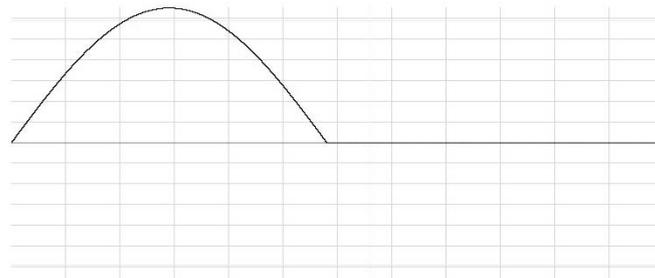


Figura 4

Filtros

El banco 1 está conformado por cinco filtros pasabanda. El ancho de banda está determinado por las siguientes frecuencias mínimas y máximas, con una pendiente de 60 dB por octava.

Filtro	F min (Hz)	F máx (Hz)
1	90	381
2	1017	2713
3	3223	3419
4	4566	4835
5	6458	6838

El banco 2, en cambio, posee solamente dos bandas

Filtro	F min (Hz)	F máx (Hz)
1	89	300
2	675	1012

En la figura 5 se observa la configuración del primer banco, formado por las unidades *reson*, con control de ganancia y Q.

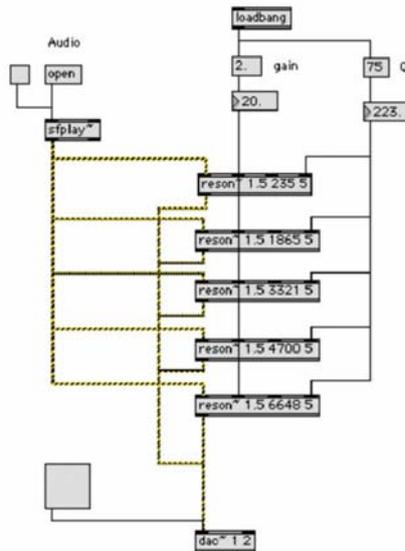


Figura 5

Vocoder

El vocoder utiliza dos bancos idénticos de filtros. Uno para separar la señal de entrada (señal de análisis) en un número determinado de bandas, el otro para separar en igual número de canales otra señal usada como fuente de excitación (señal de síntesis). Para cada banda de entrada se determina la amplitud promedio, y este valor es multiplicado por la amplitud de la banda respectiva de la señal excitadora. De este modo, la voz o el sonido de un instrumento puede servir para determinar una suerte de envolvente espectral a aplicar a cualquier otro sonido (ruido rosa, en el caso de esta obra).

En Max-MSP se simula la configuración del vocoder original, formado por dos bancos de 48 filtros cada uno, para análisis y síntesis. La distancia de la frecuencia central de cada filtro equivale a un intervalo de segunda mayor. La frecuencia del filtro 1 es 65.4 Hz y la frecuencia del filtro i está dada por:

$$f_i = f_{i-1} * 2^{1/6}$$

Este vocoder original utiliza para la obra las bandas 22 a 36. Es por ello que en la versión de Max-MSP sólo se recurre a 15 filtros, partiendo de un modelo incluido en este software, el primero con una frecuencia central de 740 Hz y el último de 3730 Hz.

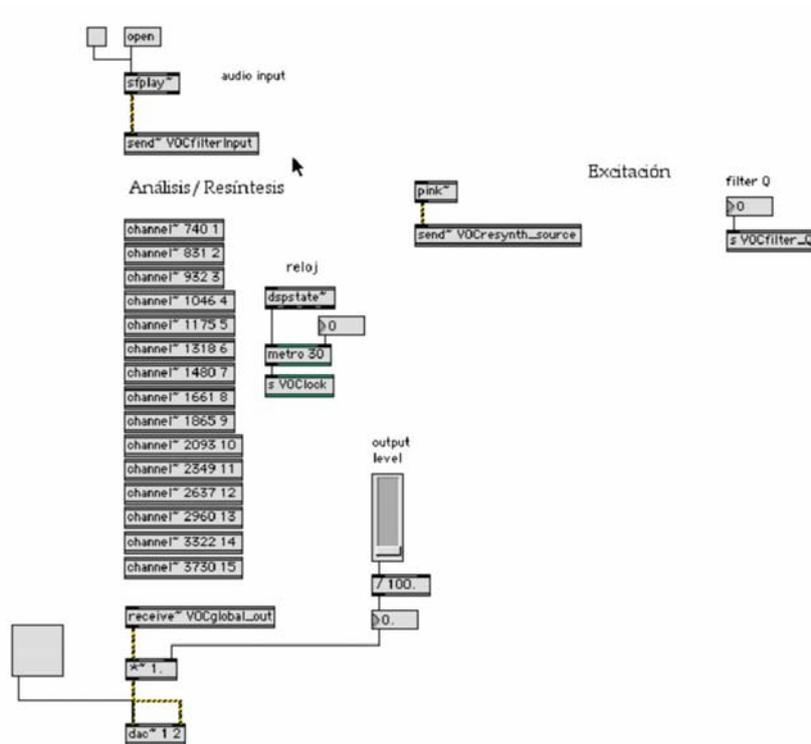


Figura 6

Gate

La amplitud de una señal es controlada por la amplitud de otra. Como ejemplo, la intensidad de la voz amplificada de una cantante es controlada por la intensidad de la voz de otra cantante. Esto puede parecer banal, pues podríamos pedirle a la primera cantante que modificara directamente el nivel de su propia voz, sin embargo, creemos se persigue una variación controlada entre el sonido directo y el amplificado y un contraste sutil en la percepción del timbre en función de la intensidad (la intensidad aumenta, pero el timbre de la voz no se modifica).

Originalmente esto se realizó con amplificadores controlados por tensión (VCA), y en nuestra versión, con el seguidor de envolventes del gráfico siguiente (ver figura 7). La unidad *peakamp* detecta cada 50 ms los picos de amplitud de la voz de control y los une formando una envolvente. Estos valores multiplicados por la amplitud de la voz a procesar dan el resultado buscado.

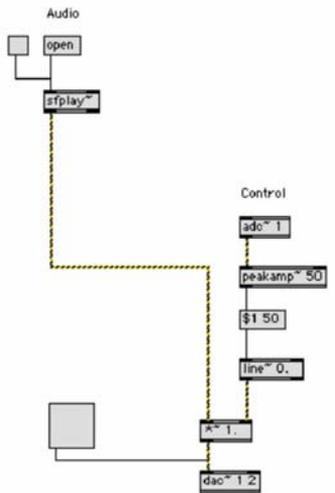


Figura 7

CADENAS DE EFECTOS

Como vimos anteriormente, la obra emplea 23 programas distintos. La figura 8 muestra las conexiones del programa 17, que domina la totalidad del segundo movimiento. Emplea ambos armonizadores, ambos retardos, retrogradación (reverser), reverberación y espacialización con el programa H2.

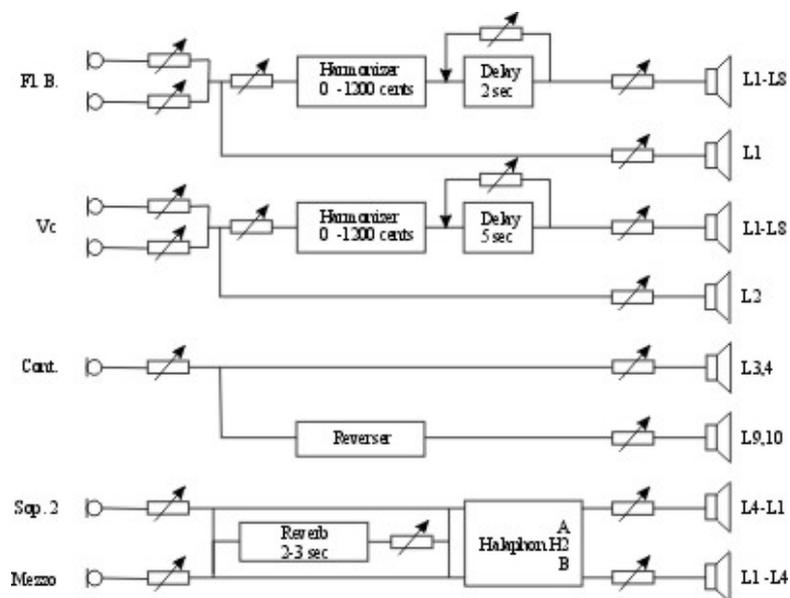


Figura 8

Cada programa se configura cambiando las entradas y salidas de los procesadores involucrados mediante una matriz de 25 filas (entradas) por 25 columnas (salidas). Cada programa fue configurado manualmente, utilizando la unidad *matrix*, y almacenados los valores de conexión en un vector.

En la figura siguiente se muestra el display de la matriz de interconexiones utilizado, que luego es retirado del programa principal para optimizar a éste en velocidad (todas las representaciones visuales deben evitarse dado que la velocidad de procesamiento es crítica cuando se trabaja en tiempo real).

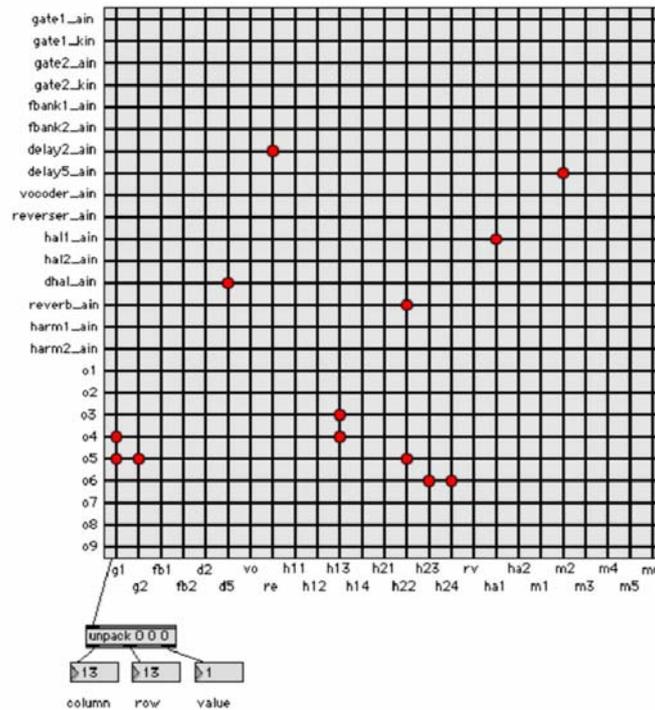


Figura 9

En la figura 10 podemos ver a la unidad *matrix* que corresponde al tercer movimiento de la obra. Se aprecian las entradas por la parte superior y las salidas por la inferior. La entrada *matriz_input* recibe las órdenes de conexión entre procesadores, micrófonos y parlantes. La unidad *matrix* realiza un *crossfade* automático entre un programa y el siguiente a fin de evitar clicks y discontinuidades en el sonido.

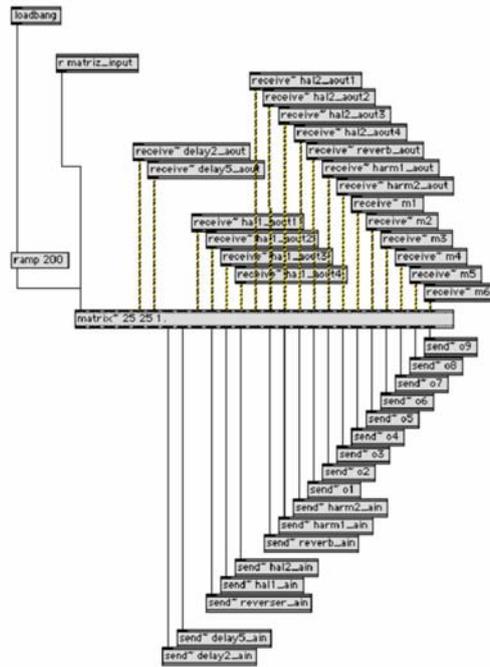


Figura 10

AUTOMATIZACIÓN DE LOS PROCESOS

Dado que los cambios de programa ocurren con mucha frecuencia, incluso varios por compás, se optó por incluir otro instrumentista a cargo de un teclado MIDI para que realizara esta tarea. Se creó una partitura donde cada nota se corresponde con un programa particular, de modo que la “melodía” interpretada nunca suena, sino que sirve de control para el cambio de programas.

La figura 11 muestra la implementación MIDI en Max-MSP, que detecta la nota ejecutada y la traduce a un número de programa a aplicar.

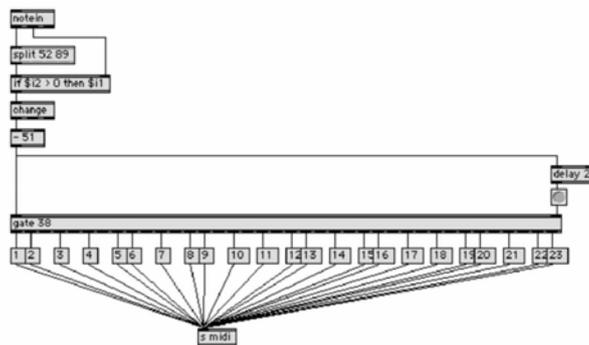


Figura 11

Cada vez que se cambia de programa, no sólo se interconectan los procesos, sino que se envían los parámetros de configuración a cada procesador: tiempo de reverberación, período de cada giro del espacializador y sentido, etc.

```

r_levels18
;
deltav_on_off 0;
reverb_init 70;
reverb_160 3;
reverb_mbx 50;
hall_periodo 10;
hall2_periodo 7;
o11 127;
o12 127;
o13 125;
o14 125;
o15 0;
o16 105;
o17 105;
o18 105;
o19 100;

r_levels19
;
transp_harm1 1.9196;
transp_harm2 1.4142;
reverb_init 70;
reverb_160 3;
reverb_mbx 50;
o11 0;
o12 0;
o13 125;
o14 125;
o15 0;
o16 0;
o17 0;
o18 92;
o19 92;

r_levels20
;
deltav_on_off 0;
hall_periodo 10;
hall2_periodo 7;
o11 0;
o12 0;
o13 127;
o14 127;
o15 108;
o16 113;
o17 118;
o18 118;
o19 117;

r_levels21
;
deltav_on_off 0;
hall_periodo 10;
hall2_periodo 7;
o11 0;
o12 0;
o13 127;
o14 127;
o15 108;
o16 113;
o17 113;
o18 113;
o19 117;

r_levels22
;
transp_harm1 0.3;
transp_harm2 0.4;
deltav_on_off 0;
hall_periodo 8;
hall2_periodo 10;
reverb_init 89;
reverb_160 8;
reverb_mbx 50;
o11 117;
o12 117;
o13 117;
o14 117;
o15 115;
o16 115;
o17 115;
o18 115;
o19 157;

r_levels23
;
deltav_on_off 0;
hall_periodo 10;
hall2_periodo 7;
reverb_init 70;
reverb_160 3;
reverb_mbx 50;
o11 91;
o12 91;
o13 0;
o14 0;
o15 92;
o16 92;
o17 92;
o18 92;
o19 0;

```

Figura 12

En la figura 12 podemos observar los valores de configuración para los programas 18 al 23 (que corresponden al tercer movimiento) y al final de cada lista los niveles de salida para cada canal de audio.

Veamos, por último, la pantalla principal del programa representada en la figura 13, que corresponde al primer movimiento de la pieza. Se aprecian diversos *faders* y vómetros de entrada y salida, el acceso a los procesadores en la parte inferior, e indicadores de programa actual y de porcentaje de utilización de la CPU.

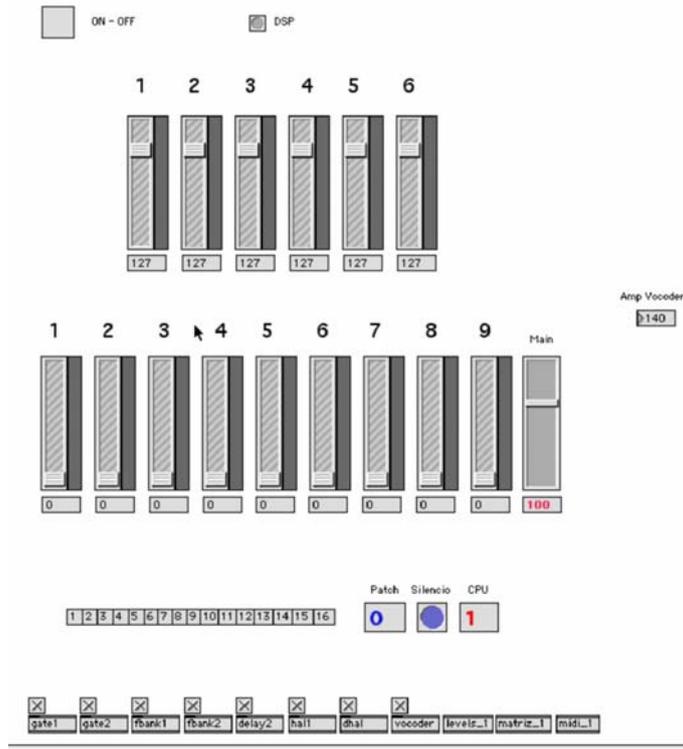


Figura 13

OMAGGIO A GYÖRGY KURTAG

Esta obra fue escrita entre 1983 y 1986 para contralto, flauta, clarinete en si bemol, tuba y electrónica en tiempo real. En esta ocasión, el autor emplea una única configuración para toda la pieza con procesos extractados de la obra anterior: cuatro líneas de retardo, cuatro reverberadores, dos armonizadores, un filtro y tres espacializadores. La reproducción es en seis canales.

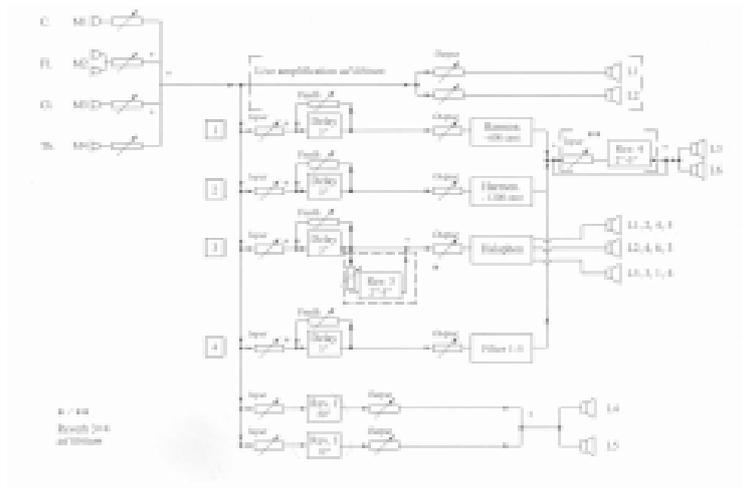
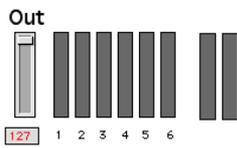
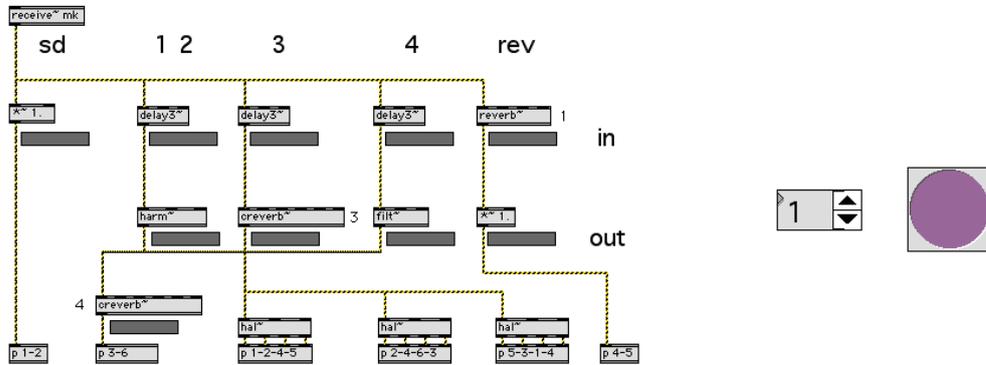
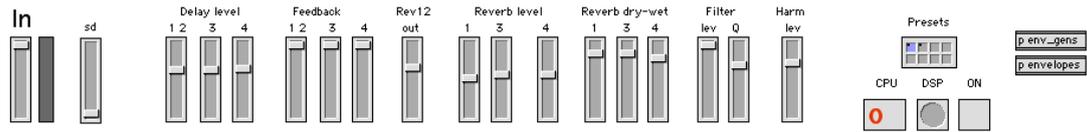


Figura 14

Se observa en la Figura 14 la distribución de las señales sumadas de los cuatro micrófonos: sonido directo, líneas de procesamiento numeradas del 1 al 4 (dos con retardo y transposición, una con retardo, reverberación y espacialización y otra con retardo y filtros) y una línea de reverberación con tiempos de 12 o 50 segundos. En la partitura se indica el recorrido que en cada momento deben efectuar las señales -por alguna de las líneas en particular, o por dos líneas simultáneas (1+3 ó 2+4)-, y se especifican mediante reguladores las evoluciones dinámicas de entrada, de salida y del nivel de realimentación de las líneas de retardo.

IMPLEMENTACIÓN EN MAX-MSP

La Figura 15 muestra la implementación en MSP del diagrama de procesamiento. No se describen las unidades por ser similares a las vistas en la obra anterior, sólo cambia la disposición. En la Figura 16 se observa el subpatch de automatización de las envolventes de entrada, salida y realimentación de las líneas de procesamiento, y en la figura siguiente, parte de los valores para generar las envolventes y asignar parámetros a los procesadores.



ALL 0 TEST 12 TEST 3 TEST 4 TEST REV TEST 1-3 TEST 2-4 ALL 1

OMAGGIO A GYORGY KURTAG
Luigi Nono

Figura 15

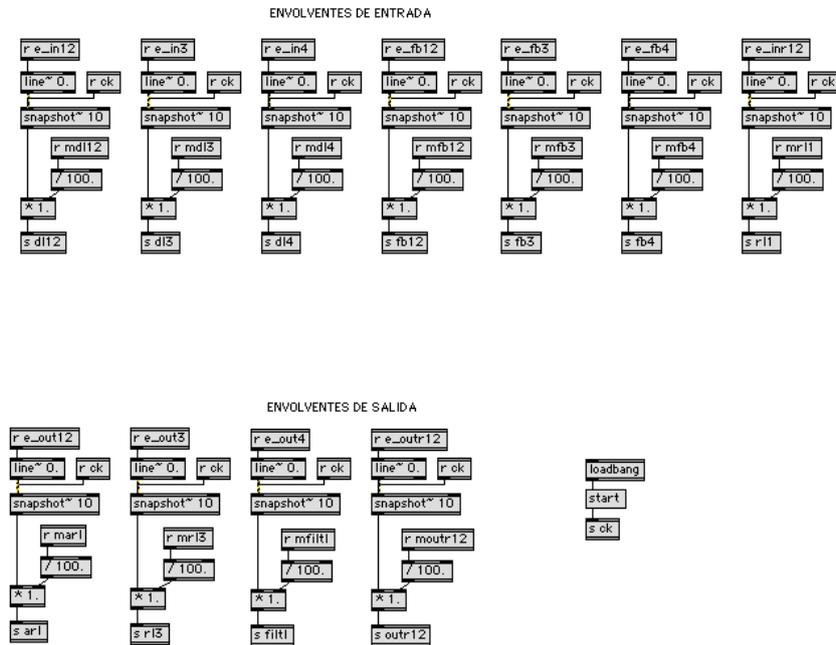


Figura 16

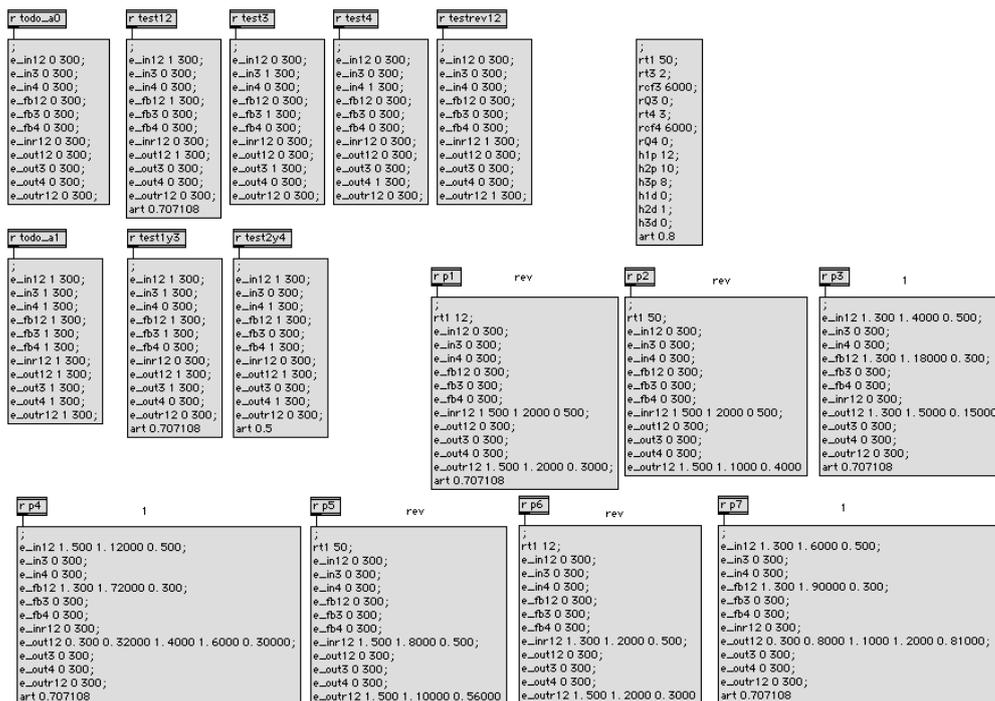


Figura 17

CONCLUSIONES

Al comenzar nuestro trabajo descubrimos la existencia de una suerte de culto en Europa por la realización de las obras de Luigi Nono con el equipamiento original usado por el compositor. Y esto un poco me recuerda a la antigua discusión sobre el valor de la interpretación de las obras clásicas con instrumentos de época o los modernos que hoy poseemos. Sin intención de tomar posición por uno u otro bando, considero que el resultado de nuestra labor ha sido muy satisfactorio, no sólo en los aspectos técnicos sino también los musicales. La posibilidad de automatizar, corregir y memorizar los procesos de transformación sonora con un elevado grado de precisión ha redundado en beneficio de la realización.

BIBLIOGRAFÍA

Moore, F. Richard. *Elements of Computer Music*. MIT Press. Massachussets. 1991.
 Nono, Luigi. *Quando stanno morendo*. *Diario Polacco N. 2*. Ricordi. Milan. 1999.
 Nono, Luigi. *Omaggio a György Kurtág*. Ricordi. Milan. 1996.

Artículo publicado en:

Cetta, P. (comp.). *Altura-Timbre-Espacio*. Cuaderno de Estudio N° 5. IIMCV. Educa. 2004.