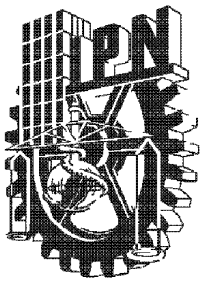

INSTITUTO POLITÉCNICO NACIONAL

CENTRO DE INVESTIGACIÓN EN COMPUTACIÓN

LABORARIO DE PROCESAMIENTO
DE LENGUAJE NATURAL

MODELO GENERATIVO DE COMPOSICIÓN MELÓDICA CON EXPRESIVIDAD

T E S I S
QUE PARA OBTENER EL GRADO DE
DOCTOR EN CIENCIAS DE LA COMPUTACIÓN
P R E S E N T A
M. EN C. HORACIO ALBERTO GARCÍA SALAS

DIRECTORES DE TESIS
DR. ALEXANDER GELBUKH
DR. FRANCISCO HIRAM CALVO CASTRO

México D.F. 2012

SIP-14

INSTITUTO POLITÉCNICO NACIONAL
SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

ACTA DE REVISIÓN DE TESIS

En la Ciudad de México, D.F. siendo las **8:30** horas del día **2** del mes de **Julio** del **2010** se reunieron los miembros de la Comisión Revisora de Tesis, designada por el Colegio de Profesores de Estudios de Posgrado e Investigación del:

Centro Investigación en Computación

para examinar la tesis titulada:

"MODELO GENERATIVO DE COMPOSICIÓN MELÓDICA CON EXPRESIVIDAD"

Presentada por el alumno:

GARCÍA	**SALAS**	**HORACIO ALBERTO**
Apellido paterno	Apellido materno	Nombre(s)

Con registro: | A | 0 | 7 | 0 | 2 | 5 | 3 |

aspirante al grado de:

DOCTORADO EN CIENCIAS DE LA COMPUTACIÓN

Después de intercambiar opiniones, los miembros de la Comisión manifestaron ***APROBAR LA TESIS***, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

LA COMISIÓN REVISORA

Presidente

Dr. Sergio Suárez Guerra

Secretario

Dr. Luis Arturo Urbeza Rodriguez

Primer Vocal
(Director de Tesis)

Dr. Alexandre Felixovich Gueiboukh Kahn

Segundo Vocal

Dr. Francisco Hiram Calvo Castro

Tercer Vocal

Dr. Jesús Manuel Olivares Ceja

PRESIDENTE DEL COLEGIO DE PROFESORES

Dr. Luis Alfonso Villa Vargas

INSTITUTO POLITÉCNICO NACIONAL
SECRETARÍA DE INVESTIGACIÓN Y POSGRADO
DIRECCIÓN

INSTITUTO POLITÉCNICO NACIONAL

SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

CARTA CESIÓN DE DERECHOS

En la Ciudad de México el día 15 del mes Diciembre del año 2011, el (la) que suscribe Horacio Alberto García Salas alumno (a) del Programa de Doctorado con número de registro A070253, adscrito a Centro de Investigación en Computación manifiesta que es autor (a) intelectual del presente trabajo de Tesis bajo la dirección de Dr. Alexander Gelbukh y cede los derechos del trabajo intitulado Modelo Generativo de Composición Melódica con Expresividad, al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a la siguiente dirección itztzin@gmail.com. Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.

Horacio Alberto García Salas
Nombre y firma

RESUMEN

Nuestra propuesta es un modelo para la generación de música con base en un enfoque lingüístico. Decimos que ciertos tipos de música se pueden modelar mediante matrices evolutivas, éstas son una forma de representación del conocimiento equivalente a gramáticas generativas probabilísticas. Las melodías musicales forman el corpus de entrenamiento, en donde cada una de ellas es considerada una frase de un lenguaje. Implementando una técnica no supervisada inferimos una gramática de este lenguaje. No hay reglas predefinidas. La generación de música se basa sobre el conocimiento musical representado por matrices probabilísticas. La información que contienen estas matrices es equivalente a las reglas de gramáticas generativas probabilísticas, que son utilizadas para generar nuevas melodías o frases musicales. Nuestro modelo está en constante aprendizaje, expandiendo su conocimiento para generar música. Con cada composición musical aprendida el modelo se reestructura para adaptarse al nuevo conocimiento.

Entendemos la música como un flujo de información que causa diferentes sensaciones. La música es un lenguaje constituido por secuencias de notas, símbolos o elementos léxicos que representan sonidos y silencios dispuestos a través del tiempo. Cada nota tiene ciertas características tales como la frecuencia, tiempo e intensidad. Nos enfocamos en estudiar el comportamiento de dos de estas variables, la frecuencia y el tiempo. Así que, consideramos una melodía como secuencias de frecuencias y de tiempos.

De cada una de estas variables se construye una función probabilística. Una secuencia de notas tiene cierta probabilidad de aparecer en una melodía. Existen ciertas secuencias que aparecen con mayor regularidad, lo que forma patrones característicos de cada composición musical. Utilizamos estas probabilidades para generar composiciones musicales de forma automática.

La expresividad es un mecanismo de supervivencia que depende de la viveza en la transmisión e interpretación de los sentimientos y emociones. En música, la expresividad transmitida a través de los instrumentos musicales depende de diferentes variables como la frecuencia, el tiempo y la intensidad.

Cada nota de una melodía es un símbolo acompañado de distintos rasgos o descriptores semánticos, que le dan significado de un sonido largo o corto, alto o bajo, intenso o suave, de una guitarra o de un piano. Con nuestro modelo es posible representar cada una de estas variables mediante matrices o gramáticas que reflejen su comportamiento probabilístico y utilizarlas para simular la expresividad en la generación de música.

Nuestra propuesta es una alternativa para la generación de música, que nos permite crear música en tiempo real y dejar de escuchar únicamente melodías pregrabadas.

ABSTRACT

We suggest a model to generate music following a linguistic approach. Musical melodies form the training corpus, each of them being considered a phrase of a language. Implementing an unsupervised technique we infer a grammar of this language. This grammar has no predefined rules. Music generation is based on music knowledge represented by probabilistic matrices. The rules of the grammars behind these matrices are used to generate new music phrases or melodies. Our model is in constant learning expanding its knowledge for generating music. With each musical composition learned the model restructures to adapt to the new knowledge. People who have heard our melodies consider them music and many of them like them.

We understand music as an information flow that causes different sensations. Music is a language made by notes, symbols or lexical elements sequences that represent sound and silences through time. Each note has some features like frequency, time and intensity. We focus to study two of these variables behavior, frequency and time. So we consider a melody formed by frequencies and times sequences.

For each of these variables we build a probabilistic function. A notes sequence has a probability of being part of a melody. There are some sequences with more probability what make characteristic patterns of each musical composition. We use these probabilities to generate music automatically.

Expressivity is a survival mechanism that depends on transmission and interpretation vividness of feelings and emotions. In music expressivity transmitted through musical instruments depends on variables such as frequency, time and intensity.

Each melody note is a symbol with several features or semantic descriptors that give the meaning of a long or short sound, low or high, intense or soft, of a guitar or of a piano. With our model is possible to represent each of these variables using matrices or grammars that reflect their probabilistic behavior and use them to simulate expressivity in music generation.

AGRADECIMIENTOS

Agradezco la oportunidad de vivir en este hermoso planeta. A mi mamá y papá, de quienes llevo su recuerdo en mi corazón. A mis hermanos, quienes han hecho de mi la persona que soy. Agradezco al Doc Galindo, un ser de gran poder y mucha luz, cuyas enseñanzas me han mostrado los senderos de la informática y más allá de los bordes infinitos. Le agradezco con todo mi corazón al Dr. Gelbukh por su gran visión y conocimiento. Agradezco a todas las personas que contribuyeron directa o indirectamente en la creación de este trabajo. Agradezco a mi *alma mater studiorum* el Instituto Politécnico Nacional por haberme formado en el camino del conocimiento.

Esta tesis se la dedico a Zanya, mi bebesita hermosa. Un ser brillante, con mucha alegría.

<<Itztli>>

.

ÍNDICE

ÍNDICE

DE FIGURAS

ÍNDICE

DE TABLAS

INTRODUCCIÓN

En esta tesis se trata el problema de la composición musical, es decir, la producción de piezas musicales originales, basadas en reglas de composición que son propias de un compositor. La acumulación de composiciones musicales que siguen reglas similares ha dado origen a los géneros musicales.

La música representa una forma de expresión, cuyos elementos básicos son sonidos y silencios dispuestos a través del tiempo. El arte de la composición musical radica en crear una secuencia armoniosa de estos elementos.

La expresividad musical que se transmite por medio de los instrumentos musicales depende de múltiples variables tales como la frecuencia, el tiempo, la intensidad, el timbre del instrumento, etc. Estos son los parámetros que un compositor y un intérprete pueden modificar para transmitir una u otra emoción. Para llevar a cabo el proceso de composición musical de forma automática desarrollamos un modelo que contempla la expresividad musical con base en dos de estas variables. Con el fin de acotar nuestra investigación, en este trabajo proponemos modelar sólo la frecuencia y el tiempo de las melodías. Sin embargo, de la misma manera en la que modelamos las frecuencias y los tiempos, se pueden modelar las otras variables que conforman una pieza musical.

Se han hecho diversos desarrollos con el fin de automatizar la generación de música, algunos de ellos de manera mecánica como en el caso de las pianolas. Sin embargo, con la aparición de las computadoras, apareció un instrumento musical capaz de generar una miríada de sonidos diferentes, una herramienta con la cual experimentar y desarrollar en tiempo real diferentes modelos de composición musical, por ejemplo, aplicando modelos matemáticos, sistemas con base en el conocimiento, gramáticas, métodos evolutivos, sistemas que aprenden, sistemas híbridos, [Papadopoulos 1999], procesos estocásticos, máquinas de estado (autómatas celulares), sistemas expertos, redes neuronales, música fractal, algoritmos genéticos y música genética (DNA) [Järveläinen 2000].

La música es novedosa en cuanto a que existe una gran cantidad de melodías diferentes y cada una de ellas tiene cierta estructura que difiere de una melodía a otra, de un autor a otro y esta diferencia es mayor entre los diferentes géneros de música. Así que cuando se busca modelar el proceso de composición musical, se trata de generar música novedosa, pero parecida a la que el oído humano está acostumbrado a escuchar.

Algunos de los modelos de composición musical existentes se basan en procesos estocásticos para generar la música y para cumplir con la estructura, se agregan reglas que restringen las composiciones. Sin embargo, mientras más reglas menos novedosa es la música generada y a menores restricciones más novedosa es la música, pero menor estructura presenta al oído humano [Todd et al. 1999]. Así que el problema es encontrar el punto adecuado entre la aleatoriedad y la regularidad, de manera que se cumplan con las características que presentan las composiciones musicales humanas.

En esta tesis nos planteamos el objetivo de encontrar en forma automatizada las reglas de composición de un conjunto de piezas musicales. Las reglas encontradas se emplean entonces, en la composición de piezas musicales novedosas, apoyadas en un proceso estocástico.

En el capítulo 1 se describen los objetivos y alcances de esta tesis. En el capítulo 2 se desarrolla el estado del arte de la composición musical. En el capítulo 3 se desarrolla el marco teórico. En el capítulo 4 se presenta el desarrollo de nuestro modelo. En el capítulo 5 se muestran algunos resultados. En el capítulo 6 las conclusiones y el trabajo futuro. El último apartado son las referencias. Finalmente, se presentan los anexos.

CAPÍTULO 1

EL PROBLEMA DE LA COMPOSICIÓN MUSICAL

En este capítulo se describe el problema que se aborda en esta tesis, los objetivos y los alcances.

1.1 Objetivos

A continuación se presentan el objetivo general y los objetivos particulares.

1.1.1 Objetivo General

Encontrar en forma automatizada las reglas de composición musical, a partir de un conjunto de piezas musicales y utilizarlas para crear música de manera automática.

1.1.2 Objetivos Específicos

Extraer los rasgos significativos que caracterizan en frecuencia y tiempo a una melodía.

Generar melodías con parámetros ajustables de acuerdo a un tipo de composición.

Desarrollar una herramienta que sirva de apoyo en la composición musical.

1.2 Justificación

La ciencia de la computación busca describir expresiones artísticas como la música a través de modelos computacionales. Nuestra investigación está relacionada con el desarrollo de modelos que permitan automatizar el proceso de composición musical. Representa dos lados de la inteligencia artificial. Por un lado, una labor científica al intentar modelar la creatividad humana en la composición musical. Por otro lado, inteligencia artificial aplicada al desarrollar una herramienta computacional para la creación musical automática.

Estas son algunas aplicaciones de nuestro modelo:

- Tener un sistema compositor personal.
- Crear música innovadora con base en patrones de diferentes estilos de música.
- Contar con herramientas que nos asistan en el proceso de composición musical.
- Tener otras alternativas para crear y escuchar música.
- Tener máquinas dedicadas a la generación de música en vivo para restaurantes, oficinas, tiendas, etc. con música creada en tiempo real.
- Contar con herramientas que permitan que los niños tengan contacto directo con el proceso de composición musical.
- Generar música personalizada.
- Utilizar otros medios en la interacción hombre-máquina.

1.3 Aportaciones

- Se propone una alternativa para encontrar reglas de composición musical en forma automatizada a partir de un conjunto de piezas musicales de uno o varios géneros.
- Hemos desarrollado una herramienta informática que compone música automáticamente.
- Hemos desarrollado un sistema evolutivo que se reestructura a sí mismo con cada melodía que aprende.

  Se propone la representación de la música como una secuencia de producciones de una gramática con producciones de tipo 4 (gramáticas regulares).
- Hemos modelado la música desde un enfoque lingüístico. La modelamos como un conjunto de frases formadas por secuencias de símbolos que constituyen un lenguaje.
- Hemos propuesto las matrices evolutivas como modelo informático para representar el conocimiento musical.

- Hemos propuesto una forma de caracterizar la música por medio de matrices evolutivas.
- Hemos desarrollado una aplicación que genera, a través de aprendizaje no supervisado y aproximación pobre en conocimiento, reglas probabilísticas que representan el lenguaje musical.
- Hemos propuesto un modelo del proceso de composición musical que simula la expresividad que un compositor humano imprime en sus creaciones.
- Hemos generado una herramienta que modela de manera independiente, las variables del sonido que representan las notas de una melodía a través de gramáticas, lo que permite mezclar rasgos de distintos géneros musicales.

CAPÍTULO 2

MARCO TEÓRICO

2.1 Música

La *música* es una forma de expresión artística de sonidos y silencios dispuestos a través del tiempo, generalmente, agradable a los seres humanos. El *arte de la composición musical* consiste en crear una secuencia armoniosa de sonidos y silencios siguiendo reglas propias del autor. Las reglas se agrupan en *géneros musicales*, de esta forma se tiene la música clásica, rock, jazz, blues, country, entre otros.

La *expresividad musical* es la forma y matices con la que se interpreta una pieza musical. En la música instrumental, es la manera en que se hacen sonar los instrumentos musicales para darle matices a una interpretación y causar efectos en un escucha. Los efectos producidos pueden ser: entusiasmo, melancolía, tristeza, añoranza, tranquilidad, entre otros. Los compositores de música utilizan tres recursos expresivos: la *dinámica*, la *agógica* y las *articulaciones*. La dinámica representa los matices de una melodía, es decir, los diferentes grados de intensidad de las notas. En una partitura, la dinámica se indica de acuerdo a la tabla 1.

Tabla 1. Dinámica de la música

Nombre	**Símbolo**	**Significado**
pianíssimo	*pp*	muy suave
piano	*p*	suave
mezzo piano	*mp*	poco suave
mezzo forte	*mf*	poco fuerte
forte	*f*	fuerte
fortissimo	*ff*	muy fuerte

La agógica indica el tempo de la melodía, es decir, la velocidad con la que debe ser interpretada una melodía.

Tabla 2. Agógica de la música

Nombre	Significado
Lento	muy despacio, con dolor
Adagio	despacio, nostálgico
Andante	que camina, vivo
Allegro	alegre
Presto	rápido, muy vivo
Prestissimo	rapidísimo, con mucha energía

Las articulaciones se utilizan para cambiar entre un sonido y otro, por ejemplo, el *legato* significa la unión del sonido de dos notas. Si se trata de dos notas iguales, el legato hace que suenen como una nota de mayor duración. El *staccato* que indica que la nota debe sonar a la mitad de su duración, convirtiendo a la otra mitad en un silencio.

El *ser humano percibe la música y la interpreta* de acuerdo con sus factores culturales, individuales y circunstanciales. Los factores culturales son aquellos relacionados con el entorno sociocultural en la que se ha desarrollado el individuo. Los factores individuales son propios de cada persona como sus recuerdos, conocimientos, experiencia, gustos, preferencias y otros. Las circunstancias son las condiciones que rodean al individuo cuando escucha una pieza musical. Una misma pieza musical puede resultar triste para una persona y alegre para otra.

Con base en lo anterior, los sentimientos manifestados por un ser humano (la expresividad sentimental) ante la interpretación de una pieza musical, son subjetivos porque carecen de una base determinística, principalmente porque los factores individuales están fuera de control, incluso del propio ser humano.

Para describir la música se han desarrollado las partituras, un lenguaje escrito que permite trasmitir, los sonidos que componen una pieza musical. Con este lenguaje se describen diferentes modulaciones (cambios de escalas), ritmos (cambios de tiempos), armonías (mezcla de los tonos).

2.1.1 Notas

El tiempo y la frecuencia de las notas son características fundamentales del mismo objeto al que generalmente se hace referencia simplemente como *nota*. Una nota musical representa un sonido o un silencio con cierta duración. En nuestro modelo computacional, una nota

representa una unidad léxico-semántica cuyos rasgos de expresividad son las variables tales como su frecuencia, su tiempo y su intensidad.

La forma en la que un compositor genera la secuencia de notas que forman una melodía, es de acuerdo a su imaginación. En el caso de nuestro modelo, la información contenida en las gramáticas que se generan a partir de ejemplos de música, representa la imaginación de un compositor humano.

2.1.2 Tiempos

Los tiempos de una melodía hacen referencia a las duración de cada una de la secuencia de notas que la forman. En los tiempos de una melodía intervienen, principalmente, dos factores: uno el tempo y el otro, el valor de cada una de las notas que puede ser de 1, ½, ¼, 1/8, 1/16, 1/32, 1/64 y con las computadoras se llega a usar el 1/128. El tempo, generalmente se mide en BPM (*beats per minute*).

El compás de una melodía se establece al principio de su partitura, aunque, puede variar con el transcurso de ésta. El compás indica cuántas notas y de qué valor deben ser tocadas por cada compás. Por ejemplo: un compás de 4/4 indica que se tocan 4 notas de ¼ en cada compás. En general, la música occidental se representa mediante notas con tiempos que adquieren valores de potencias de $1/2^x$, donde $x \in Z$ y $0 \leq x \leq 7$.

2.1.3 Tempo

El tempo indica el *beat* de una melodía y al igual que un reloj de una computadora se utiliza para sincronizar sus procesos, el *beat* se utiliza para sincronizar los instrumentos empleados en la interpretación de una melodía. El *beat* se mide en BPM (Beats per Minute) y es directamente proporcional a la velocidad con la que es interpretada una melodía, por lo que al aumentar o disminuir el *beat* la melodía se acelera o desacelera.

Que tan rápida o lenta es interpretada una melodía es una de las variables que afectan directamente su expresividad. Los tempos se clasifican de acuerdo a su valor en: Grave (40), Largo (45), Larghetto (50), Lento (55), Adagio (60), Adagietto (65), Andante (70), Andantino (80), Moderato (95), Allegretto (110), Allegro (120), Vivace (145), Presto (180), Prestissimo (220). El número entre paréntesis indica los BMP.

Una melodía al ser interpretada a mayor número de BPM, hace que la duración de las notas sea menor, aunque su valor sea el mismo. El tempo que se escoja para una melodía influye en las duraciones individuales de las notas que la componen. Por ejemplo una nota de valor ¼ tocada a 60 BMP dura 1 seg. y si es interpretada a 120 BMP dura 0.5 seg. En algunas ocasiones, estas diferencias en el *beat* de una melodía, son utilizadas por los músicos para transformar algunas de las notas de mayor duración en varias notas de menor duración.

2.2 El enfoque lingüístico

La aparición de las computadoras ha modificado la manera de concebir al mundo. Una computadora puede ser considerada un instrumento musical sofisticado que permite generar una infinidad de sonidos diferentes, abriendo un camino con grandes posibilidades para la creatividad humana. Actualmente se desarrollan diversos algoritmos para modelar la realidad, lo que ha permitido automatizar diversas actividades humanas, entre ellas la musical.

La aplicación de ciencias como la del procesamiento del lenguaje natural a los campos del arte como la música, está abriendo nuevas áreas de investigación en donde se estudia la música en términos lingüísticos, con el fin de desarrollar compositores automáticos de música. En nuestro trabajo, proponemos un modelo para la creación de nuevas formas de música. Mediante nuestro modelo se describe la música desde un enfoque lingüístico y se ha aplicado en la creación de un sistema compositor. Un sistema compositor con la capacidad de crear nuevas formas de música con base en el conocimiento aprendido a partir de melodías musicales. Un compositor automático que además, puede generar nuevos rasgos para una composición pregrabada, de manera que cada vez que sea interpretada suene con diferentes rasgos de expresividad, equivalente a que un músico la interprete "en vivo".

La música es un lenguaje formado por cadenas de símbolos. Una manera de modelar el proceso de composición musical es mediante matrices evolutivas, en donde cada nota representa un sonido cuyos rasgos de expresividad son todas las variables que intervienen en la composición, tales como la frecuencia, el tiempo y la intensidad. Cada una de estas notas tiene cierta probabilidad de aparecer en una melodía y existen ciertas secuencias de

notas que aparecen con mayor regularidad, lo que identifica los patrones probabilísticos característicos de cada composición musical. A partir de estos patrones se forman las reglas que se utilizan para generar una composición musical de manera automática.

El objetivo de nuestra investigación es modelar el proceso de composición musical y generar música de manera automática. Esto es, generar una secuencia de sonidos y silencios que sean agradables al oído del ser humano. Consideramos a la música como un flujo de información que provoca diversas sensaciones en el ser humano.

Las reglas expresadas por medio de gramáticas y datos estadísticos, se utilizan para generar una secuencia de notas en las que reflejan los patrones aprendidos de manera no supervisada. El resultado es una composición musical creada por una máquina.

Algunos desarrollos del área de procesamiento de lenguaje natural buscan automatizar el uso de los lenguajes, con el fin de que las computadoras puedan utilizar interfaces más amigables que faciliten el proceso de comunicación hombre-máquina. Se han hecho diversos desarrollos en esta área, por ejemplo, las máquinas de búsqueda que facilitan encontrar información en el internet.

Muchos de los avances que se presentan en esta área se deben a la introducción de las gramáticas generativas. Las gramáticas son una herramienta lingüística que ha sido utilizada para describir los lenguajes. A través de reglas de producción se describen los elementos, estructuras y significados de las oraciones que constituyen un lenguaje. Computacionalmente, se han utilizado para el desarrollo de programación y reconocimiento y generación automática de los lenguajes naturales.

La aplicación de las gramáticas y las técnicas desarrolladas en el procesamiento de lenguaje natural, se pueden extender para representar una amplia gama de lenguajes naturales, que no necesariamente están formados por palabras. Desde este punto de vista, un lenguaje es aquello que se puede percibir y representar por una secuencia de símbolos [Galindo 1995], por ejemplo, el lenguaje musical que es una secuencia de símbolos que describe sonidos y silencios. Entonces, se puede hablar del procesamiento del lenguaje musical y entre otras aplicaciones, automatizar el proceso de composición musical.

2.3 Afectividad computacional

Las investigaciones con respecto a la afectividad incluyen desarrollos en diversas áreas. Algunas de éstas tienen que ver con la importancia que tiene la comunicación visual entre los humanos, por lo que se han desarrollado modelos que permiten reconocer emociones a través de las expresiones faciales humanas [Yacoob et al. 1996] [Essa *et al*. 1997] [Pantic *et al*. 2000] [Tian *et al*. 2001]. Otros modelos se han desarrollado con el fin de reconocer las caras de las personas no importando la orientación que presenten [Feng *et al*. 2000]. Por otro lado, también se encuentra el desarrollo de modelos que permiten estar monitoreando el estado de ánimo de los niños, con el fin de saber si están aprendiendo o no lo que ven a través de una pantalla de computadora. Por medio de sensores de presión y de videos, a partir de las posturas que los niños presentan mientras están sentados en una silla, se determina que tan alertas están con respecto a lo que están estudiando [Mota *et al*. 2003].

El desarrollo de humanos virtuales que permitan establecer una conversación [Egges *et al.* 2003], incluye investigaciones con respecto a la personalidad, estado emocional y estado de ánimo, en donde estas características humanas se representan mediante vectores que dan como resultado el comportamiento de un agente inteligente. En las personificaciones de los agentes virtuales inteligentes es necesario incluir el componente afectivo, con el fin de que sean entendidos por el usuario [Aylett 2004]. Las caricaturas son una de las áreas donde estas investigaciones tienen aplicaciones directas, principalmente en sus rostros. Para lograr que los agentes se comporten como en la vida real, hay que tomar en cuenta diversos factores cognitivos como: la importancia de los objetivos, la probabilidad de éxito o fracaso, el agrado que un agente siente por otro y factores no-cognitivos que tienen que ver con las posturas, las expresiones faciales, los estados de ánimo, la memoria emocional [Lee 1999].

Silfridge [1959] propone una teoría a la que denomina pandemonio basada en las teorías psicológicas del comportamiento de estímulo-respuesta. Con base en ésta, Jackson [1987] publica "idea para una mente" en donde los "demonios", que son lo que actualmente se denominan agentes, actúan de manera autónoma, induciendo ocasionalmente a otros agentes a la acción. De su investigación le surge la pregunta de si la mente humana es sólo una computadora que se encarga de todo el procesamiento o muchas de ellas, que

interactúan de forma independiente? A partir de estos trabajos se han desarrollado diferentes modelos, por ejemplo aquellos en los que se modelan las emociones, los estados de ánimo y los temperamentos por medio de agentes que luchan para determinar el estado de ánimo general del modelo [Velásquez *et al*. 1997] o para representar la arquitectura de las emociones [McCauley 1998].

La generación de voz sintética es otra de las áreas dedicadas al reconocimiento y generación de emociones, cuyas aplicaciones directas se encuentran en los convertidores de texto-voz [Francisco *et al*. 2005]. El desarrollo de agentes conversacionales que con base en un ambiente multi-agente determinan el grado de la intensidad de las expresiones emocionales [Bui *et al*. 2002], algunos disponibles a través de la internet con capacidades de procesamiento de lenguaje natural y manipulación de emociones para establecer conversaciones de diferentes temas con los usuarios [Tatai *et al*. 2003].

En cuestiones afectivas el problema es que ni siquiera los seres humanos pueden reconocer al 100% las emociones, ni de otras personas e incluso ni las propias, pese a que son parte de los rasgos característicos de los seres humanos [Picard *et al*. 2001].

En el caso de la música, frecuentemente, no se tiene al compositor o al intérprete al cual se le puedan medir rasgos como el tono de voz, la expresión facial, cambios en los gestos, en la respiración o en la tensión muscular, por mencionar algunos, que podrían dar indicios para determinar su comportamiento afectivo, que tiene influencia directa en la expresividad de la composición musical. Cuando se trata de música, generalmente, sólo se cuenta con las partituras o archivos sonoros, en los que se pueden escuchar los rasgos que dan expresividad a la composición. Entre estos se encuentran los tonos (frecuencias), las escalas utilizadas, los tiempos y las intensidades (amplitud).

CAPÍTULO 3

ESTADO DEL ARTE

Existen diversas razones por las que se lleva a cabo la investigación al respecto de la composición musical con la ayuda de computadoras. Por un lado se busca modelar la *composición* que implica desarrollar modelos que lleven a cabo el proceso de composición musical por medio de algoritmos computacionales; *Ingeniería de software* que es el desarrollo de herramientas para ayudar a los compositores a hacer composición musical; *Musicología* área en la que se desarrollan modelos de los diferentes estilos de música, implementando las diferentes reglas que los distinguen; y *ciencia del conocimiento* con la que se modela el conocimiento musical [Pearce 2002].

La composición algorítmica se lleva a cabo a través de ciertos pasos, en los cuales se involucran instrucciones con el fin de obtener una composición musical [Papadopoulos 1999]. La idea básica es hacer un sistema que genere música de manera automática y se pueden utilizar las computadoras como una herramienta que permite la implementación de diferentes modelos de composición musical.

En la década de 1950, Lejaren Hiller compuso "*The Illiac Suite for String Quartet*" que se considera la primer pieza musical compuesta basada en algoritmos de computadora. Esta pieza tiene cuatro movimientos. El primero se compuso a través de la generación de notas aleatorias y decidiendo de acuerdo a reglas de cierto estilo musical, cuáles eran aceptadas. El segundo movimiento se generó de la misma manera, sólo incluyendo reglas de otros estilos de música. El tercer movimiento se basó en estructuras definidas. El cuarto movimiento se compuso utilizando cadenas de Markov. El resultado fue una lista de números que se tradujeron manualmente a notas musicales. En la década de 1960, junto con Robert Baker, desarrolló el sistema MusiComp basado en estas ideas.

En la década de 1950, Iannis Xenakis compuso manualmente utilizando fórmulas estocásticas. Los elementos como el timbre, la intensidad y la duración de las notas fueron

arreglos generados basados en la distribución de Poison. En la década de 1960 desarrolló SMP (Stochastic Music Program) en donde aplicó mediciones estadísticas sobre el comportamiento del movimiento de partículas de los gases. La generación de música está basada en el movimiento browniano de las partículas.

En la década de 1960, Gottfried Michael König desarrolló el software de composición PR1 (Proyect 1). Su trabajo se basa en la teoría de la música serial, que establece el uso de secuencias con estructuras definidas, a las que se les aplican transformaciones como repetición (en este trabajo no se permite la repetición), inversión, transposición y otras. Estas estructuras se seleccionan de forma aleatoria basadas en 7 principios de selección y se aplican a 5 parámetros: instrumento, ritmo, armonía, registro y dinámica. El resultado es una lista de notas que tiene que ser transcrita a notación musical.

En la década de 1970, Barry Truax desarrolló el sistema POD (Poison Distribution). Este sistema a diferencia de anteriores, en donde se tenía como salida números o partituras, introduce el concepto de audio digital, en donde se utilizan dispositivos de síntesis de audio para escuchar la salida del sistema. Los elementos básicos de control son lo que llama "tendecy mask", que definen regiones de frecuencia contra tiempo. En los límites de estas regiones el sistema genera eventos de acuerdo a la distribución de Poison. El usuario puede modificar parámetros como el tempo, la dirección, el grado de traslape y otros. El resultado es música monofónica.

Una forma de desarrollar compositores para la creación de música de manera automática, es a través del modelado de diferentes tipos de señales, entre los que destacan los ruidos blanco, browniano y rosa. En el apéndice A mostramos algunos algoritmos para la generación de música utilizando este tipo de señales [García 1999]. También es posible utilizar estos ruidos para la enseñanza de las matemáticas [Bulmer 2000]. El estudio del ruido 1/f o rosa es de particular importancia ya que se presenta en diversos fenómenos. La música humana presenta características de este tipo de ruido [Gardner 1978].

Los sistemas de composición algorítmica se desarrollan de tres maneras [Todd et al. 1999]: sistemas que utilizan reglas definidas, los que aprenden y los que evolucionan. Hiller e Isaacson [1959] desarrollaron un modelo computacional para la composición algorítmica, basándose en cadenas de Markov. Otra herramienta utilizada en la generación de música

son las redes neuronales, por ejemplo Harmonet desarrollado en 1991 [Hild 1992], emplea *connectionist networks* para el procesamiento de música. Este modelo ha sido entrenado para producir corales al estilo de J. S. Bach.

Otra forma en la que se ha generado música de manera automática es a través de algoritmos genéticos. Biles [2001] desarrolla GenJam un modelo basado en un algoritmo genético que imita a un músico novato de jazz aprendiendo a improvisar. Birchfield [2003] desarrolló un algoritmo genético coevolutivo con el objetivo de producir música con emoción, significado y forma, basado en el modelo de emoción de Meyer [1956]. Este modelo propone que el significado que se le da a la música es debido a que un evento musical (tono, frase o toda una sección) provoca que el oído humano espere otro evento y por lo tanto, esta expectativa se puede utilizar para generar emociones.

Todd et al. [1999] desarrollaron un algoritmo genético con el fin de modelar la composición musical basándose en procesos de coevolución. Por un lado, se tienen "machos" que son los compositores, a los que se les pueden definir reglas o las pueden aprender por ejemplo, para la generación de ritmo. Por otro lado, se tienen "hembras" que son quienes van a juzgar las composiciones de los "machos" y de acuerdo a su gusto por la composición decidirán si aceptan o no al "macho". A través de la reproducción se van generando nuevas piezas musicales.

Blackwell [2007] utilizó el modelado de enjambres para obtener composiciones musicales de forma automática. Básicamente, se trata de modelar aquellas sociedades como las abejas, hormigas o avispas en las que los individuos cooperan para construir de forma organizada sus enjambres. Se modelan con propiedades emergentes de sistema complejos, en donde los procesos con los que se organizan estas sociedades emergen de forma espontánea y es modelando esta propiedad con la que se genera la música.

Actualmente, las investigaciones se enfocan a la generación de la música automáticamente tomando en cuenta el contenido afectivo de la música. Mántaras *et al*. [2006] proponen un modelo con el fin de modificar el tiempo melodía musical, preservando su expresividad, utilizando la técnica de razonamiento basado en casos.

En [Legaspi *et al*. 2007] se propone modelar la afectividad utilizando un algoritmo genético cuya función ideal se determina con base en fragmentos de partituras etiquetados a mano

con el fin de establecer pares afectivos bipolares que califican la música, con lo que denominaron CAUI (*Constructive Adaptive Music Interface*). Con base en fragmentos se componen las melodías, en donde se les introducen variantes de forma aleatoria y sobre la base de teoría musical básica.

Se han desarrollado otros modelos del conocimiento musical con el objetivo de establecer su estructura, ayudar en su trascripción, hacer resúmenes y en su recuperación. Por ejemplo, Namunu *et al*. [2004] llevaron a cabo el análisis de la estructura de la música basado en su contenido, con el fin de entender su significado. Kosina [2002] hizo el reconocimiento automático del género de música basado exclusivamente en el contenido de audio de la señal.

El reconocimiento del genero de cierto tipo de música es una tarea que se lleva a cabo fácilmente por un ser humano, sin embargo, que una computadora haga este reconocimiento no es tan sencillo. Aymeric *et al*. [2004] hicieron la recuperación automática de información musical a partir de cierta información almacenada en señales acústicas. Melucci *et al*. [1999] tratan de resolver el problema de recuperación de información musical, utilizando el método de detección de límites locales (*local boundaries detection method*) desarrollado por Cambouropoulos.

Blackburm *et al*. [1998] presentan un sistema para navegar a través del contenido de una base de datos musical, que incluye archivos de audio (MIDI). La búsqueda se hace basándose en el contorno de la música, con una representación de los cambios relativos en las frecuencias de una melodía, independientemente del tono o del tiempo. [Peña 2005] presenta el desarrollo de un lenguaje formal llamado KL (Kernel Language) con el fin de proporcionar un laboratorio para la simulación de una orquesta digital, el procesamiento de partituras y la generación de los sonidos de cada instrumento, por medio de músicos virtuales conectados vía midi.

McAlpine *et al*. [1999] desarrollaron el sistema llamado CAMUS 3D (Cellular Automata MUSic in 3 Dimensions) para modelar el proceso de composición basado en la propagación de patrones. En su trabajo desarrollaron dos tipos de algoritmos: estocásticos y autómatas celulares.

Para generar composiciones utilizan una extensión tridimensional del autómata del "juego de la vida" y el autómata "Demon Cyclic Space". En el autómata del "juego de la vida", cada célula considerada viva (color negro) representa un evento musical y sus coordenadas *x*, *y*, *z* representan los intervalos de las notas de un acorde Fig. 1. Estas notas no son necesariamente distintas y pueden o no sonar simultáneamente. El autómata celular se inicializa con una configuración de celdas definida por el usuario y las reglas con las que evoluciona son las del "juego de la vida".

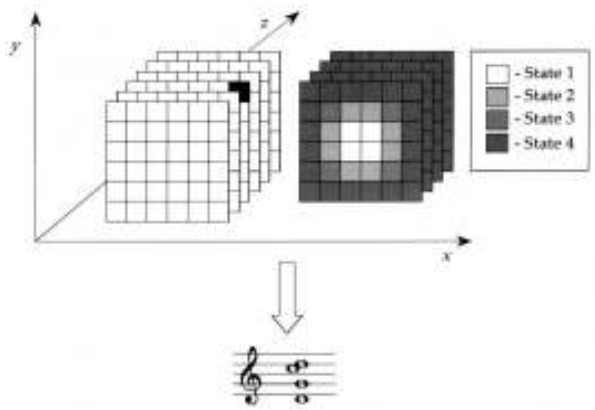

Fig. 1. Autómatas celulares de CAMUS 3D (tomada de [McAlpine *et al*. 1999])

Una vez definidas las notas (en forma de contenido interválico), se determina la nota fundamental, lo cual se puede hacer de dos maneras: 1) de manera manual, siguiendo una lista de acordes del usuario o 2) de manera automática, utilizando una selección estocástica, en donde se elige la nota con base en pesos asignados por un usuario.

Pata evitar que la composición sea una secuencia de acordes no placenteros, implementaron un proceso estocástico que consulta una tabla definida por el usuario, de la cual se escoge entre 24 estados probables.

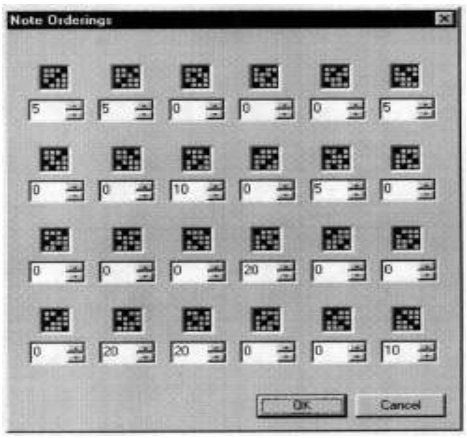

Fig. 2. Probabilidad del orden de las notas de CAMUS 3D (tomada de [McAlpine *et al*. 1999])

El autómata Demon Cyclic Space se utiliza para determinar la instrumentación, es decir, los instrumentos musicales con los que se interpretarán las notas. Este autómata se inicializa con estados aleatorios. El instrumento con la que se emite la nota que corresponde a uno de los canales midi, se elige de acuerdo a la celda "viva" que le corresponde en el autómata del "juego de la vida". El usuario puede cambiar las reglas de evolución de los dos autómatas.

Para calcular la duración de las notas, se utiliza una cadena de Markov de primer orden. Las probabilidades de la cadena se especifican por el usuario mediante una matriz de transición. El usuario utiliza dos métodos para especificar las probabilidades: mediante una interfaz gráfica y otro, que permite el acceso directo a los valores numéricos. En la tabla 3 y la tabla 4 se presenta una comparación entre el sistema CAMUS y Zanya el sistema desarrollado mediante la aplicación de nuestro modelo.

Tabla 3. Comparación CAMUS 3D y Zanya

	CAMUS 3D	Zanya
Algoritmos utilizados	Estocásticos y autómatas celulares	Estadístico y estocástico
Composición	manual: lista de notas del usuario automática: selección estocástica (usuario da las probabilidades).	Selección estocástica basada en funciones probabilísticas aprendidas.
Salida	Genera arreglos con base en las reglas del autómata del "juego de la vida" y el autómata "Demon Cyclic Space".	Genera melodías automáticamente con base en patrones aprendidos.
Tiempos de las notas	Se generan con base en una matriz de transición, con probabilidades determinadas por el usuario.	Se generan de acuerdo a las probabilidades aprendidas y representadas mediante una matriz evolutiva.
Forma de generar	Es un sistema de propagación de	Es un generador de patrones con

patrones	patrones de acuerdo a restricciones predefinidas.	restricciones aprendidas.
Mecanismo	Es una interpretación acústica del comportamiento del autómata celular 3D.	Es un generador de música basado en el comportamiento de los músicos.
Repetición de melodías	Puede repetir la misma melodía, pero no el mismo sonido. El autómata es determinístico y puede repetir la misma secuencia de acordes e instrumentaciones si es inicializado con los mismos valores. Los procesos estocásticos hacen que las notas (pitches), el orden de las notas y su duración, suenen completamente diferentes.	Si puede repetir exactamente la misma melodía y se pueden producir composiciones diferentes controlando el generador de números aleatorios.
Forma de aprendizaje	El usuario puede cambiar manualmente las reglas de evolución de los dos autómatas.	El usuario influye en los patrones generados proporcionando ejemplos.
Tipo de música generada	Genera secuencias de acordes predefinidos.	No genera secuencias predefinidas.
Objetivo	Armonización Parcialmente composición	Composición
Técnicas	Autómatas celulares "juego de la vida" Procesos estocásticos	Procesos estadísticos Procesos estocásticos
Entrada	Parámetros para los autómatas Parámetros para las funciones estocásticas	Melodías
Salida	Armonizaciones y composiciones	Composiciones
Variables usadas	Estados de los autómatas Probabilidades de la cadena de Markov	Frecuencia y tiempo
Iteración	Humana	Semiautomatizada

En la tabla 3, se presentan dos propuestas relacionadas con la música por computadora. Camus 3D que armoniza una melodía compuesta por el usuario o bien generada de acuerdo a una función de probabilidad, en donde los pesos los asigna el usuario. Para armonizar la melodía se apoya en dos autómatas celulares y en una cadena de Markov para determinar las duraciones de las notas. Y Zanya, el sistema desarrollado con base en nuestro modelo, se apoya en una matriz evolutiva para aprender, estadísticamente, los patrones musicales de forma no supervisada y generar música estocásticamente con base en los patrones aprendidos. CAMUS 3D es una sonificación de un proceso matemático que requiere que un usuario establezca diversos parámetros y Zanya que genera composiciones musicales

basado en las reglas que infiere a partir de música humana y sólo se le necesita proporcionar melodías musicales.

En la tabla 4 presentamos una comparación entre algunos otros sistemas de composición musical.

Tabla 4. Comparación entre varios sistemas compositores

	CAMUS 3D	Lejaren Hiller MusiComp	Iannis Xenakis SMP Stochastic music program	Michael König PR1	Zanya
Objetivo	Armonización Parcialmente composición	Composición	Composición	Composición	Composición
Técnicas	Autómatas celulares "juego de la vida" Procesos estocásticos	Aleatorio Procesos estocásticos	Fórmulas estocásticas Distribución de Poison	Teoría música serial Procesos estocásticos	Procesos estadísticos Procesos estocásticos
Entrada	Parámetros para los autómatas Parámetros para las funciones estocásticas	Números aleatorios Reglas: permitido, prohibido o requerido	Mediciones esadísiticas del comportamiento de las partículas de gases (movimiento browniano)	Números aleatorios Cantidad de eventos a generar Tempo	Melodías
Salida	Armonizaciones y composiciones	Números	Secuencias de secciones	Lista de notas	Composiciones
Variables usadas	Estados de los autómatas Probabilidades de la cadena de Markov	Frecuencia Probabilidades cadena de Markov		Instrumento, ritmo, armonía, registro y dinámica	Frecuencia y tiempo
Iteración	Humana	Manual		Semiautomática	Semiautomática

En la tabla 4 presentamos algunos de los sistemas desarrollados para modelar el proceso de composición musical. La principal aportación de nuestro trabajo, que diferencia nuestro modelo de aquellos presentados en la tabla 4, es en cuanto a la generación de reglas. La mayoría de los modelos utilizan reglas definidas por el usuario, ya sean mediante la asignación de probabilidades, la utilización de una función de probabilidad o utilizando las reglas de algún proceso matemático. Nuestro modelo infiere las reglas de forma no supervizada, a partir de la música existente, por lo que la música que genera nuestro sistema se centra en la generación de patrones que se encuentran en la música "natural".

CAPÍTULO 4

DESARROLLO

El proceso musical sucede en tres planos: mental, interpretativo y auditivo [Miranda *et al*. 2006]. En el plano mental se lleva a cabo el proceso de composición, cuyo resultado es una partitura conteniendo la información musical que posteriormente será utilizada en el plano de interpretación. El plano de la interpretación se lleva a cabo por el intérprete, quien se encarga de procesar la información de la composición musical, traduciéndola en sonido. Finalmente en el plano del oyente se lleva a cabo el proceso de escuchar el resultado de los procesos de composición e interpretación, cuya evaluación dependerá del auditorio. En la Fig. 3 se muestra este modelo.

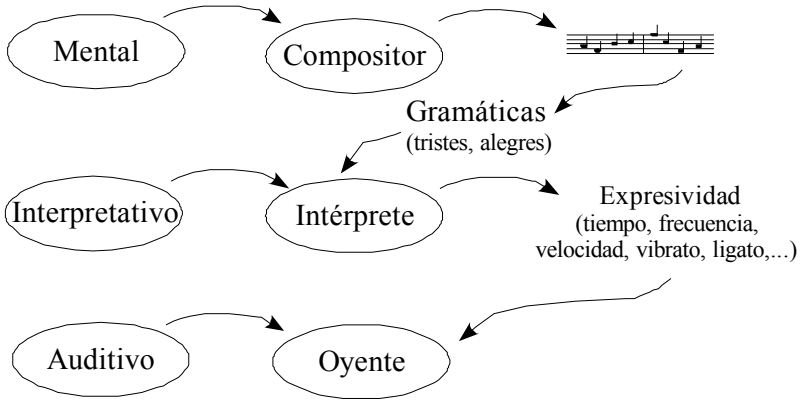

Fig. 3. Modelo del proceso de composición musical

En este trabajo se propone un modelo del plano mental y del interpretativo. El proceso de composición musical implica la concepción de una idea, la cual es transformada en una secuencia de sonidos. La forma de la idea tiene que ver con las características personales del compositor, que van desde su herencia genética, la música y sonidos que escucha en el vientre materno, las emociones que la madre le transmite a su hijo al escuchar cierto tipo de música, los estados de ánimo que experimenta durante su crecimiento y que asocia a ciertos sonidos y muchas otras variables que influencian su forma de pensar y sentir en términos musicales.

Para transformar sus ideas en una obra musical, un compositor hace uso de su creatividad y de su conocimiento. La forma de combinar las notas de manera armoniosa, el timbre de los distintos instrumentos para darle color y la métrica que decide utilizar, son sólo parte de las variables que le permiten formar las estructuras musicales con las que transmite sus pensamientos y emociones a través de secuencias de sonidos.

El resultado del proceso de composición es una partitura en donde se representan las variables musicales de manera tal que, la composición musical pueda ser compartida y en su momento interpretada. Desafortunadamente, en una partitura no se pueden reflejar todas las características que el compositor tiene en mente cuando compone la obra.

La interpretación se puede llevar a la práctica por el mismo compositor o por un intérprete. En el caso de que sea el mismo compositor quien interprete, mejor podrá compartir sus ideas a su audiencia. Sin embargo, si su obra musical requiere de más de un instrumento, como en el caso de las obras sinfónicas, otros músicos tendrán que intervenir para su interpretación, aportando a la obra sus características personales.

El músico intérprete también tiene una historia personal que le permite entender la música de manera particular y esto se ve reflejado al momento de tocar su instrumento. Por ejemplo, en el caso de los cantantes, donde su instrumento es la voz, pueden interpretar una misma canción y cada uno de ellos imprimirá su sello personal a la melodía, haciendo que suene diferente a pesar de ser la misma. Lo mismo ocurre con cualquiera de los instrumentos musicales.

La solución que proponemos para modelar el proceso de composición musical está basada en el concepto de s*istemas evolutivos*, que plantea que los sistemas evolucionan como resultado del cambio constante producido por el flujo de materia, energía e información [Galindo 1991]. Los sistemas evolutivos tienen la capacidad de interrelacionarse con el medio que los rodea, utilizando funciones que les permiten aprender y adaptarse a los cambios constantes que se presentan ante ellos, encontrando las reglas que rigen su comportamiento. Estas reglas pueden ser expresadas en forma de gramáticas. El esquema de nuestro modelo se muestra en la Fig. 4.

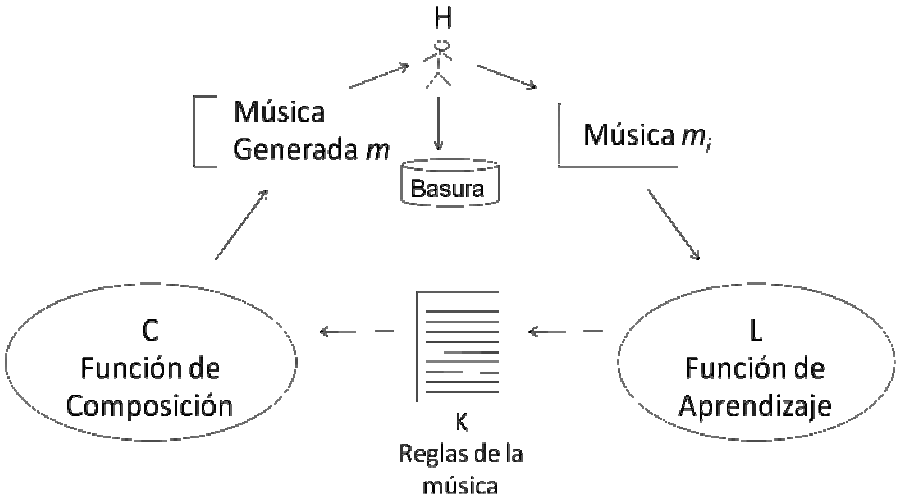

Fig. 4. Esquema del Modelo

En algunos modelos se definen las reglas de composición, lo que significa que se implementan diversas restricciones para generar la música. El lado positivo de tener restricciones se ve reflejado en la estructura de la música, es decir, las reglas permiten obtener composiciones que cumplen con las estructuras que frecuentemente se encuentran en las melodías. Sin embargo, un problema que se presenta en las composiciones musicales, es que generalmente se incluyen excepciones a las reglas, lo que permite que haya más diversidad en la música y por otro lado, definir las reglas plantea un problema en sí mismo. De esta forma, un sistema con reglas definidas modela la estructura de la música, pero genera melodías poco novedosas [Todd 1999].

En nuestra propuesta es fundamental la propiedad evolutiva del modelo, que permite que las gramáticas no estén formadas con reglas fijas, sino que se pueden ir modificando de acuerdo a la realidad [Galindo 1991] con los nuevos ejemplos de música, obteniéndose las reglas de manera automática. Así se refleja la estructura de la música y por otro lado, se obtiene la variedad que se espera de una pieza musical. Para que el modelo aprenda a componer como algún autor en particular, basta con proporcionarle ejemplos de dicho autor.

Nuestro modelo está basado en un enfoque lingüístico [Galindo 1995]. Describimos las composiciones musicales como frases formadas por secuencias de notas como elementos léxicos que representan sonidos y silencios a través del tiempo. El conjunto de todas las composiciones musicales constituye el lenguaje musical.

Definición 1: Una *nota* es una representación del tono y duración de un sonido musical.

Definición 2: El *alfabeto* es el conjunto de todas las notas: alfabeto = {notas}.

Definición 3: Una *composición musical m* es un arreglo de notas musicales: Composición musical = $a_1\ a_2\ a_3 \ldots a_n$ donde $a_i \in$ {notes}.

En nuestra investigación trabajamos con composiciones musicales *m* de música monofónica o melodías, modelamos dos variables: frecuencias y tiempos musicales. Dividimos estas variables para formar una secuencia de símbolos con cada una de ellas.

Definición 4: El Lenguaje Musical es el conjunto de todas las composiciones musicales: Lenguaje Musical = {composiciones musicales}.

Por ejemplo, la secuencia de notas (frecuencias) de la composición musical "*El cóndor pasa*":

b e $d_\#$ e $f_\#$ g $f_\#$ g a b_2 d_2 b_2 e_2 d_2 b_2 a g e g e b e $d_\#$ e $f_\#$ g $f_\#$ g a b_2 d_2 b_2 e_2 d_2 b_2 a g e g e b_2 e_2 d_2 e_2 d_2 e_2 g_2 e_2 d_2 e_2 d_2 b_2 g e_2 d_2 e_2 d_2 e_2 g_2 e_2 d_2 e_2 d_2 b_2 a g e g e

Asumimos que esta secuencia es una frase del lenguaje musical.

4.1 Sistema evolutivo musical

Los sistemas evolutivos interactúan con el medio, encontrando reglas que describen los fenómenos y utilizan funciones que les permiten aprender y adaptarse a los cambios. El esquema de nuestro sistema evolutivo se muestra en la Fig. 4.

El espacio de trabajo de las reglas del lenguaje musical está representado por K y existen muchas maneras de hacer esta representación, por ejemplo, gramáticas, matrices, redes neuronales, enjambres y otras. Cada género musical, estilo y autor tienen sus propias reglas de composición. No todas estas reglas son descritas en la teoría musical. Para hacer una composición automática utilizamos un sistema evolutivo para encontrar de manera no supervisada las reglas K.

La función L es un proceso de aprendizaje que genera las reglas a partir de cada composición musical m_i creando una representación K del conocimiento musical. El sistema evolutivo originalmente no tiene ninguna regla definida. Llamamos K_0 cuando K

está vacía. Mientras nuevos ejemplos m_0, m_1, …, m_i musicales son aprendidos K se modifica desde K_0 a K_{i+1}.

$$L(m_i, K_i) = K_{i+1}$$

La función L extrae los rasgos musicales de m_i y los integra a K_i, generando una nueva representación K_{i+1}. Esto significa que la representación del conocimiento K evoluciona de acuerdo con los ejemplos aprendidos. Estas reglas aprendidas K son utilizadas para generar una composición musical *m* automáticamente. Es posible construir una función C(K) donde C se llama compositor musical. La función C utiliza K para producir una nueva composición musical *m*.

$$C(K) = m$$

Para escuchar una nueva composición existe una función I llamada intérprete musical que genera el sonido.

$$I(m) = \text{sound}$$

La función I toma la música *m* generada por la función C y la manda al dispositivo de sonido. En este sentido las notas generadas son de acuerdo a la tabla midi presentada en el Anexo E. Se ejecutan funciones de generación de sonido de acuerdo al formato midi. De modo que, la función I es un reproductor de audio en formato midi, así que no hablaremos más sobre esta función.

4.1.1 Módulo de aprendizaje musical L

El proceso de aprendizaje L es la función que extrae los rasgos musicales y adiciona esta información a K. Existen diferentes formas de representar K. En nuestro trabajo utilizamos una representación basada en matrices y mostramos en la Sección IV que esta representación es equivalente a una gramática probabilística.

Definición 5: Frecuencia Musical = {frecuencias musicales} donde frecuencias musicales *mf* son el número de vibraciones por segundo (Hz) de las notas.

Definición 6: Tiempo Musical = {tiempos musicales} donde tiempos musicales *mt* son las duraciones de las notas.

La función L recibe composiciones musicales *m* y construye las reglas K. Composición Musical $m = a_1\ a_2\ a_3 \ldots\ a_n$ donde $a_i = \{f_i, t_i\}$, $i \in [1,n]$, $f_i \in$ Frecuencia Musical, $t_i \in$ Tiempo Musical, $[1,n] \subset \mathrm{N}$

Para representar las reglas K de las frecuencias y los tiempos musicales utilizamos matrices. Nos referimos a ellas como reglas M. Originalmente, estas matrices están vacías; se modifican con cada ejemplo musical.

Las reglas M son divididas por la función L en MF y MT, donde MF es el componente de las reglas referentes a las frecuencias musicales (m*f*) extraídas de las composiciones musicales y MT es la componente del las reglas del tiempo musical (*mt*).

Definición 7: MF es un espacio de trabajo formado por dos matrices. Una de ellas es la *matriz de distribución de frecuencias* (FDM) y la otra es la *matriz de distribución de frecuencias acumuladas* (CFM).

Cada vez que una composición musical m_i llega, L actualiza FDM. Entonces calcula CFM de la siguiente manera:

Definición 8: Sea FrecuenciaNotas un arreglo en el cual se almacenan los números correspondientes a las notas de una composición musical.

Definición 9: Sea *n* el número de notas reconocidas por el sistema, $n \in \mathrm{N}$.

Definición 10: Matriz de distribución de frecuencias (FDM) es una matriz con *n* renglones y *n* columnas.

Dada la composición musical $m = f_1\ f_2\ f_3 \ldots\ f_r$ donde $f_i \in$ FrequenciaNotas. El algoritmo de aprendizaje de L para generar la matriz de distribución de frecuencias FDM es:

$$\forall i \in [1,r],\ [1,r] \subset \mathrm{N},\ \mathrm{FDM}_{fi,fi+1} = \mathrm{FDM}_{fi,fi+1} + 1,$$

$$\text{donde } \mathrm{FDM}_{fi,fi+1} \in \mathrm{FDM}$$

Definición 11: La matriz de distribución de frecuencias acumuladas (CFM) es una matriz con *n* renglones y *n* columnas.

El algoritmo de L para generar la matriz de distribución de frecuencias CFM es:

$$\forall i \in [1,n],\ \forall j \in [1,n],\ [1,n] \subset \mathrm{N},\ \forall \mathrm{FDM}_{i,j} \neq 0$$

$$\mathrm{CFM}_{i,j} = \sum_{k=1}^{j} \mathrm{FDM}_{i,k}$$

Estos algoritmos para generar MF, el espacio formado por FDM y CFM, son utilizados por la función L con cada composición musical m_i. Esto hace que el sistema evolucione recursivamente de acuerdo con las composiciones musicales $m_0, m_1, m_2, \ldots, m_i$.

$$L(m_i, \ldots L(m_2, L(m_1, L(m_0, MF_0)))) = MF_{i+1}$$

4.1.3 Ejemplo

Tomemos la secuencia de frecuencias de la composición musical "*el cóndor pasa*". La melodía se muestra como una secuencia de letras minúsculas que equivalen a las notas musicales de acuerdo a la tabla 6.

Tabla 5. Notas musicales y su símbolo

Nombre	***Símbolo***
La	a
Si	b
Do	c
Re	d
Mi	e
Fa	f
Sol	g

El símbolo # significa que es sostenido, el subíndice se refiere a la escala y el tiempo no se toma en consideración:

b e $d_{\#}$ e $f_{\#}$ g $f_{\#}$ g a b_2 d_2 b_2 e_2 d_2 b_2 a g e g e b e $d_{\#}$ e $f_{\#}$ g $f_{\#}$ g a b_2 d_2 b_2 e_2 d_2 b_2 a g e g e b_2 e_2 d_2 e_2 d_2 e_2 g_2 e_2 d_2 e_2 d_2 b_2 g e_2 d_2 e_2 d_2 e_2 g_2 e_2 d_2 e_2 d_2 b_2 a g e g e

El módulo *aprende* se encarga de extraer, seleccionar y clasificar los rasgos pertenecientes a la música, generando las reglas que la describen. Para representar el lenguaje musical por medio de matrices se utilizan unidades semánticas que conforman el conjunto de los símbolos terminales. Cada unidad semántica está formada por la frecuencia de la nota y su duración. Con cada una de estas características se generan los valores de la matriz, cuyas probabilidades se determinan de acuerdo a la secuencia de unidades semánticas que conforman los ejemplos de música compuesta por seres humanos.

El conjunto de notas {**b**, $\mathbf{d_{\#}}$, **e**, $\mathbf{f_{\#}}$, **g**, **a**, $\mathbf{b_2}$, $\mathbf{d_2}$, $\mathbf{e_2}$, $\mathbf{g_2}$} representa los símbolos terminales o alfabeto del "condor pasa". Estos símbolos son usados para etiquetar cada columna de la

matriz de distribución de frecuencias FDM. Cada número almacenado en la FDM de la Fig. 5, representa cuantas veces la nota del renglón es seguida por la nota de la columna, en la melodía del cóndor pasa. Para almacenar la primera nota de cada composición musical se agrega el renglón S, que representa el axioma o símbolo inicial. Aplicando el algoritmo de aprendizaje de L generamos la matriz de frecuencias distribuidas FDM de la Fig. 5, que muestra como quedan los valores de las casillas después de haber aprendido la melodía.

	b	d♯	e	f♯	G	A	b♭	d♭	e♭	g♭
S	1									
b			2							
d♯			2							
e	1	2		2	3		1			
f♯					4					
g			6	2		2			1	
a					3		2			
b♭					1	3		2	3	
d♭							6		6	
e♭								10		2
g♭									2	

Fig. 5. Matriz de distribución de frecuencias FDM

En el Anexo B se muestra una gráfica en tres dimensiones de esta matriz. La función que desempeña el proceso de aprendizaje es determinar las probabilidades con las que el autor utiliza ciertas secuencias de notas con mayor regularidad. Esto es, encontrar los valores probabilísticos con los que se suceden los símbolos terminales que describen la melodía dada como ejemplo. La matriz se mantiene actualizada de manera permanente, de acuerdo a la información contenida en cada una de las melodías.

Aplicamos el algoritmo de L para calcular la matriz de distribución de frecuencias acumuladas CFM de la Fig. 6. a partir de la matriz de distribución de frecuencias FDM de la Fig. 5. Entonces, calculamos cada T_i de la columna T. En el Anexo B se muestra una gráfica en tres dimensiones de esta matriz.

	b	**d♯**	**e**	**f♯**	**g**	**A**	**b♭**	**d♭**	**e♭**	**g♭**	**T**
S	1										1
b			2								2
d♯			2								2
e	1	3		5	8		9				9
f♯					4						4
g			6	8		10			11		11
a					3		5				5
b♭					1	4		6	9		9
d♭							6		12		12
e♭								10		12	12
g♭									2		2

Fig. 6. Matriz de distribución de frecuencias acumuladas CFM

Definición 12: MT es un espacio de trabajo formado por dos matrices. Una de ellas es la *matriz de distribución de tiempos* (TDM) y la otra es la *matriz de distribución de tiempos acumulados* (CTM).

Cada vez que una composición musical m_i llega, el proceso de aprendizaje L actualiza TDM. Entonces calcula CTM de la siguiente manera:

Definición 13: Sea TiempoNotas un arreglo en el cual se almacenan los números correspondientes a los tiempos de una composición musical.

Definición 14: Matriz de distribución de tiempos (TDM) es una matriz con *n* renglones y *n* columnas.

Dada la composición musical $m = t_1\ t_2\ t_3 \ldots\ t_r$ donde $t_i \in$ TiempoNotas. El algoritmo de aprendizaje de L para generar la matriz de distribución de tiempos TDM es:

$$\forall i \in [1,r],\ [1,r] \subset \mathrm{N},\ \mathrm{TDM}_{ti,ti+1} = \mathrm{TDM}_{ti,ti+1} + 1,$$

$$\text{donde } \mathrm{TDM}_{ti,ti+1} \in \mathrm{TDM}$$

Definición 15: La matriz de distribución de tiempos acumulados (CTM) es una matriz con *n* renglones y *n* columnas.

El algoritmo de L para generar la matriz de distribución de tiempos CTM es:

$$\forall i \in [1,n],\ \forall j \in [1,n],\ [1,n] \subset \mathrm{N},\ \forall \mathrm{TDM}_{i,j} \neq 0$$

$$\mathrm{CTM}_{i,j} = \sum_{k=1}^{j} \mathrm{TDM}_{i,k}$$

Estos algoritmos para generar MT, el espacio formado por TDM y CTM, son utilizados por la función L con cada composición musical m_i. Esto hace que el sistema evolucione recursivamente de acuerdo con las composiciones musicales $m_0, m_1, m_2, \ldots, m_i$.

$$\mathrm{L}(m_i, \ldots \mathrm{L}(m_2, \mathrm{L}(m_1, \mathrm{L}(m_0, \mathrm{MT}_0)))) = \mathrm{MT}_{i+1}$$

4.1.3 Ejemplo

Tomemos la secuencia de tiempos de las notas de la composición musical "*el cóndor pasa*". Se muestra como una secuencia de letras minúsculas que equivalen a los tiempos musicales de acuerdo a la tabla 7:

Tabla 6. Notas, su símbolo y su valor en tiempos

Nombre	*Símbolo*	*Duración (tiempos)*
Redonda	r	1
Blanca	b	1/2
Negra	n	1/4
Corchea	c	1/8
Semicorchea	h	1/16
Fusa	f	1/32
Semifusa	m	1/64

El módulo *aprende* se encarga de extraer la secuencia de los tiempos pertenecientes a las notas de la melodía, quedando como se muestra a continuación:

n n n n n n n n n r b r n n r n n r b r n n n n n n n n n r b r n n r n n r b r b r n n n r n n r n n r r b r n n n r n n r n n r b r r n n r b r r

Con cada uno de estos tiempos se calculan estadísticamente los valores de la matriz TDM

El conjunto de tiempos {**r**, **b**, **n**} representa los símbolos terminales o alfabeto del "condor pasa", pues sólo se utilizan estos tres tiempos en esta melodía. Estos símbolos son usados para etiquetar cada columna de la matriz de distribución de tiempos TDM. Cada número almacenado en la matriz TDM de la Fig. 7, representa cuantas veces el tiempo del renglón t_i es seguido por el tiempo de la columna t_j, en la melodía del cóndor pasa. Para almacenar el primer tiempo de cada composición musical se agrega el renglón S, que representa el axioma o símbolo inicial. Aplicando el algoritmo de aprendizaje de L se genera la matriz de tiempos distribuidos TDM de la Fig. 7, en donde se muestra como quedan los valores de las casillas después de haber aprendido la melodía. En el Anexo B se muestra una gráfica en tres dimensiones de esta matriz.

	r	**b**	**n**
S			1
r	1	7	12
b	7		
n	13		29

Fig. 7. Matriz de distribución de tiempos TDM

A partir de la matriz de distribución de tiempos TDM de la Fig. 7 y aplicando el algoritmo L, se calcula la matriz de distribución de tiempos acumulados CTM de la Fig. 8. Al igual que con la matriz de frecuencias, se calcula cada T_i de la columna T. En el Anexo B se muestra una gráfica en tres dimensiones de esta matriz.

Una vez construida la matriz de distribución de tiempos acumulados CTM, se pueden determinar las probabilidades con las que el autor utiliza ciertas secuencias de tiempos con mayor regularidad. Esto es, encontrar los valores probabilísticos con los que se suceden los símbolos terminales que describen los tiempos de la melodía dada como ejemplo. Con la información extraída de cada nueva melodía, las matrices se mantienen actualizadas.

	r	**b**	**n**	**T**
S			1	1
r	1	8	20	20
b	7			7
n	13		42	42

Fig. 8. Matriz de distribución de tiempos acumulados CTM

4.1.2 Módulo compositor de Música C

La composición de música monofónica es el arte de crear una línea melódica sin acompañamiento. Para componer una melodía un compositor humano usa su creatividad y conocimiento musical. En nuestro modelo la función Compositor C genera una línea melódica basada en el conocimiento representado por las matrices de distribución de frecuencias acumuladas CFM y la de distribución de tiempos acumulados CTM.

Para la generación de música es necesario determinar la siguiente nota. En nuestro modelo cada renglón *i* de CFM representa una función de probabilidad para cada nota, con base en esta función se toma la decisión cual es la siguiente nota de la composición. Cada columna diferente de cero, representa las posibles notas que pueden seguir a la nota *i*. Las notas más probables forman patrones característicos.

Definición 12: T_i es un elemento en donde se almacena el total de la suma de las frecuencias acumuladas de cada renglón *i* de FDM.

$$\forall i \in [1,n], [1,n] \subset \mathrm{N}, \quad \mathrm{T}_i = \sum_{k=1}^{n} \mathrm{FDM}_{i,k}$$

T es una columna con *n* elementos donde es almacenado el total de la suma de las frecuencias acumuladas de FDM.

Algoritmo generador de notas

```
i=0
repeat
```

```
{
 p=random(0..T_i)
 while(CFM_i,j < p)
   j=j+1
    next note=j
 i=j
}
```

Para generar una composición musical se utiliza el algoritmo generador de notas. La generación de música comienza escogiendo la primer nota de la composición. El renglón S contiene todas las notas iniciales posibles. En nuestro ejemplo, sólo la nota **b** puede ser escogida. Así que **b** es la primer nota y el renglón *i* de $CFM_{i,j}$ que utilizamos para determinar la segunda nota. Sólo la nota **e** puede ser seleccionada después de la primer nota **b**.

Así, las primeras dos notas de esta nueva melodía musical son m_{i+1}={**b**, **e**}. Aplicando el algoritmo generador de notas para determinar la tercer nota: Tomamos el valor almacenado en la casilla del renglón de la nota **e** y la columna T, T_e=9. Se genera un número aleatorio *p* entre cero y 9, supongamos *p*=6. Para encontrar la siguiente nota, comparamos el número aleatorio *p* con cada valor diferente de cero del renglón **e**, hasta encontrar uno que sea mayor o igual. Entonces, **g** es la siguiente nota ya que $M_{e,g}$=8 es mayor que *p* = 6. De esta forma, la tercera nota de la nueva composición musical m_{i+1} es **g** por lo que m_{i+1} = {**b**, **e**, **g**,…}. Dado que es la columna *j* = **g** donde se encuentra almacenada la siguiente nota, entonces, para determinar la cuarta nota debemos aplicar el algoritmo generador de notas al renglón *i* = **g**.

Cada valor diferente de cero de cada renglón *i*, representa la función de probabilidad de las notas que acostumbran seguir a la nota *i*. Por lo tanto, se generan patrones de notas musicales de acuerdo a la función de probabilidad aprendida directamente de los ejemplos de composiciones musicales humanas, lo que permite generar música basada en patrones de música humana.

4.2 Generación de gramáticas

4.2.1 Generación de la gramática a partir de una matriz

Nuestro trabajo está basado sobre un enfoque lingüístico y hemos utilizado un espacio de trabajo representado por matrices para manejar la información musical. Ahora mostramos que esta representación de la información es equivalente a una gramática generativa probabilística.

Existen diferentes maneras para obtener una gramática generativa G, un caso particular no supervisado es nuestro modelo. A partir de la matriz de distribución de frecuencias FDM y la columna total T, es posible construir una gramática generativa probabilística.

Definición 13: MG es un espacio de trabajo formado por FDM y una gramática probabilística G.

Para obtener una gramática primero generamos una matriz de probabilidades PM, determinada a partir de la matriz de distribución de frecuencias FDM.

Definición 14: La matriz de probabilidades PM es una matriz con n renglones y n columnas.

El algoritmo para generar la matriz de probabilidades PM es:

$$\forall i \in [1,n],\ \forall j \in [1,n],\ \forall \mathrm{FDM}_{i,j} \neq 0 \qquad \mathrm{PM}_{i,j} = \mathrm{FDM}_{i,j}/\mathrm{T}_i$$

Existe una gramática generativa probabilística $G\{V_n, V_t, S, P, Pr\}$, tal que G puede ser generada a partir de PM. V_n es el conjunto de símbolos no terminales, V_t es es el conjunto de todos los símbolos terminales o alfabeto, que representa las notas musicales. El símbolo inicial o axioma está representado por S. P es el conjunto de reglas generadas y Pr es el conjunto de probabilidades de las reglas, representado por los valores de la matriz PM.

Para transformar la matriz PM en la gramática G, utilizamos el siguiente algoritmo:

1. Se construye la matriz auxiliar AM a partir de la matriz PM:
    a. Se substituyen las etiquetas de cada renglón i de PM.
    b. Se substituyen las etiquetas de cada columna j por su nota f_j y un no terminal X_j.
    c. Se copian todos los valores de las celdas de la matriz PM en las celdas

correspondientes de la matriz AM.

2. Para cada renglón *i* y cada columna *j* tal que $AM_{i,j} \neq 0$:
    a. El renglón *i* corresponde a la regla X_i, el lado izquierdo de las reglas de producción de la gramática.
    b. La columna *j* corresponde a un símbolo terminal f_j y un símbolo X_j no terminal con probabilidad $p_{i,j}$, lo que forma el lado derecho de las reglas de producción de la gramática.

Entonces las reglas de la gramática G son de la forma $X_i \rightarrow f_j\ X_j\ (p_{i,j})$. Esta es una representación de nuestro modelo por medio de gramáticas generativas probabilísticas. Por cada composición musical m_i un espacio de trabajo MG formado por FDM y la gramática G, puede ser generado recursivamente.

$$L(m_i,\ldots L(m_2, L(m_1, L(m_0, MG_0))))=MG_{i+1}$$

4.2.1.1 Ejemplo

Para generar una gramática que describa el lenguaje al que pertenece esta melodía, a partir de la información contenida en la matriz, se crean reglas de producción en donde los símbolos terminales se suceden de acuerdo a las probabilidades con las que se suceden las notas de la melodía.

A partir de la matriz de distribución de frecuencias FDM de la Fig. 5, es generada la matriz de probabilidades PM de la Fig. 9. En el Anexo B se muestra una gráfica en tres dimensiones de esta matriz.

	b	**d♯**	**e**	**f♯**	**g**	**A**	**b2**	**d2**	**e2**	**g2**
S	1									
b			1							
d♯			1							
e	1/9	2/9		2/9	3/9		1/9			
f♯					1					
g			6/11	2/11		2/11			1/11	
a					3/5		2/5			
b2					1/9	3/9		2/9	3/9	
d2							6/12		6/12	
e2								10/12		2/12
g2									1	

Fig. 9. Matriz de probabilidades PM

Se determinan las probabilidades de que una nota suceda a otra. Por ejemplo, con respecto al tercer renglón de la matriz, el que pertenece a la nota *e*, con los valores de las casillas y

utilizando la columna *total*, que representa la suma total de las casillas del renglón, se pueden determinar las probabilidades de cada columna diferente de cero.

Cada una de las columnas con valores diferentes de cero representa la nota *j* que se utilizó después de la nota *i* y el número de veces que se hizo. El valor contenido en la casilla representa la probabilidad de que la nota *j* suceda a la nota *i*.

A partir de la matriz PM de la Fig. 9 se genera de acuerdo al algoritmo, la matriz auxiliar AM de la Fig. 10. En el Anexo B se muestra una gráfica en tres dimensiones de esta matriz.

	bX_1	$d_{\#}X_2$	eX_3	$f_{\#}X_4$	gX_5	aX_6	b_2X_7	d_2X_8	e_2X_9	g_2X_{10}
S	1									
X_1			1							
X_2			1							
X_3	1/9	2/9		2/9	3/9		1/9			
X_4					1					
X_5			6/11	2/11		2/11			1/11	
X_6					3/5		2/5			
X_7					1/9	3/9		2/9	3/9	
X_8							6/12		6/12	
X_9								10/12		2/12
X_{10}									1	

Fig. 10. Matriz auxiliar AM

Con la información contenida en esta matriz es posible generar una gramática probabilística que describe al lenguaje al que pertenece la melodía. Esta gramática permite generar no sólo la melodía, sino todas las posibles oraciones del lenguaje al que pertenece dicha melodía. En términos musicales significa que se pueden generar diversas melodías equivalentes a estas oraciones.

Dada la matriz AM de la Fig. 10 se puede generar la gramática $G\{V_n,V_t, S, P, Pr\}$. Donde $V_n=\{S, X_1, X_2 X_3, X_4, X_5, X_6, X_7, X_8, X_9, X_{10}\}$ es el conjunto de los símbolos no terminales. $V_t=\{b, d_{\#}, e, f_{\#}, g, a, b_2, d_2, e_2, g_2\}$ es el conjunto de los símbolos terminales o alfabeto. S es el axioma o símbolo inicial. Pr es el conjunto de las probabilidades de las reglas representado por los valores de la matriz AM. Y las reglas P de la gramática, listadas en la Fig. 11.

$$S \rightarrow b\ X_1(1)$$
$$X_1 \rightarrow e\ X_3(1)$$
$$X_2 \rightarrow e\ X_3(1)$$

$X_3 \rightarrow b\ X_1(1/9) \mid d_{\#}\ X_2(2/9) \mid f_{\#}\ X_4(2/9) \mid g\ X_5(3/9) \mid b_2\ X_7(1/9)$

$X_4 \rightarrow g\ X_5(1)$

$X_5 \rightarrow e\ X_3(6/11) \mid f_{\#}\ X_4(2/11) \mid a\ X_6(2/11) \mid e_2\ X_9(1/11)$

$X_6 \rightarrow g\ X_5(3/5) \mid b_2\ X_7(2/5)$

$X_7 \rightarrow g\ X_5(1/9) \mid a\ X_6(3/9) \mid d_2\ X_8(2/9) \mid e_2\ X_9(3/9)$

$X_8 \rightarrow b_2\ X_7(6/12) \mid g_2\ X_{10}(6/12)$

$X_9 \rightarrow d_2\ X_8(10/12) \mid g_2\ X_{10}(2/12)$

$X_{10} \rightarrow e_2\ X_9(1)$

Fig. 11. Gramática generativa probabilística

La gramática de la Fig. 11 representa el lenguaje al que pertenece la melodía "el cóndor pasa". Con esta gramática es posible generar un infinito de oraciones o teoremas que pertenecen a este lenguaje. Por lo que, se pueden generar melodías similares que cumplen con las reglas extraídas del ejemplo, pero que son por siempre diferentes.

4.2.2 Generación de la gramática mediante inferencia gramatical

Un caso particular no supervisado de generar una gramática es mediante una *matriz evolutiva*. Nuestro modelo es independiente de la forma de representar la información. Otra manera de generar una gramática es mediante inferencia gramatical, directamente sobre las melodías. Es posible generar otra gramática con el fin de describir la estructura de las melodías. Para ello, la secuencia de notas de la melodía debe ser tratada como una cadena de caracteres en donde se puedan encontrar patrones que se repitan, factorizarlos y determinar los casos en los que se pueda utilizar recursividad.

Por ejemplo, la secuencia de notas del "cóndor pasa": $d_{\#}\ e\ f_{\#}\ g\ f_{\#}\ g\ a\ b_2\ d_2\ b_2\ e_2\ d_2\ b_2\ a\ g\ e\ g\ e\ d_{\#}\ e\ f_{\#}\ g\ f_{\#}\ g\ a\ b_2\ d_2\ b_2\ e_2\ d_2\ b_2\ a\ g\ e\ g\ e\ b_2\ e_2\ d_2\ e_2\ d_2\ e_2\ g_2\ e_2\ d_2\ e_2\ d_2\ b_2\ g\ e_2\ d_2\ e_2\ d_2\ e_2\ g_2\ e_2\ d_2\ e_2\ d_2\ b_2\ a\ g\ e\ g\ e$. La generación de la gramática comienza encontrando los patrones más largos que se repitan. La cadena más larga de esta secuencia de frecuencias es: $b\ e\ d_{\#}\ e\ f_{\#}\ g\ f_{\#}\ g\ a\ b_2\ d_2\ b_2\ e_2\ d_2\ b_2\ a\ g\ e\ g\ e$. La palabra (bis) indica que la cadena anterior se repite. Se crea una regla de producción para esta cadena, por ejemplo, le podemos asignar un no-terminal (X). Se sustituye el no-terminal X en la cadena original quedando la gramática de la siguiente manera:

$S \rightarrow X\ X\ b_2\ e_2\ d_2\ e_2\ d_2\ e_2\ g_2\ e_2\ d_2\ e_2\ d_2\ b_2\ g\ e_2\ d_2\ e_2\ d_2\ e_2\ g_2\ e_2\ d_2\ e_2\ d_2\ b_2\ a\ g\ e\ g\ e$

$X \rightarrow b\ e\ d_{\#}\ e\ f_{\#}\ g\ f_{\#}\ g\ a\ b_2\ d_2\ b_2\ e_2\ d_2\ b_2\ a\ g\ e\ g\ e$

En esta cadena se encuentran patrones que se repiten, estos serán factorizados. Nuevamente, se buscan primero las secuencias más largas. A cada una se les asigna un símbolo no terminal y se genera su regla de producción. Por ejemplo, a la secuencia $a\ g\ e\ g\ e$ se le asigna el no-terminal (Z), a la secuencia $e_2\ d_2$ el no-terminal (Y) y a la secuencia $f_{\#}\ g$ se le asigna el no-terminal (P). Se modifica la regla de producción X, introduciendo los no-terminales en lugar de las cadenas a las que fueron asignados. Con estos cambios la gramática construida hasta el momento queda:

$$X \rightarrow b\ e\ d_{\#}\ e\ P\ P\ a\ b_2\ d_2\ b_2\ Y\ b_2\ Z$$

$$P \rightarrow f_{\#}\ g$$

$$Y \rightarrow e_2\ d_2$$

$$Z \rightarrow a\ g\ e\ g\ e$$

De esta manera, se puede continuar buscando patrones que se repitan al menos una vez y agruparlos en una regla de producción, de forma tal que se construye la gramática que se muestra en la Fig. 12.

$N \rightarrow L\ b_2\ S \rightarrow X\ X\ b_2\ O\ g\ O\ Z$

$X \rightarrow b\ e\ d_{\#}\ e\ P\ P\ a\ b_2\ d_2\ b_2\ Y\ b_2\ Z$

$O \rightarrow L\ M\ N$

$Z \rightarrow a\ g\ e\ g\ e$

$P \rightarrow f_{\#}\ g$

$Y \rightarrow e_2\ d_2$

$L \rightarrow Y\ Y$

$M \rightarrow e_2\ g_2$

Fig. 12. Gramática que describe la estructura del Cóndor Pasa

Construyendo el árbol sintáctico de esta gramática se puede observar la estructura de la melodía “cóndor pasa”, Fig. 13.

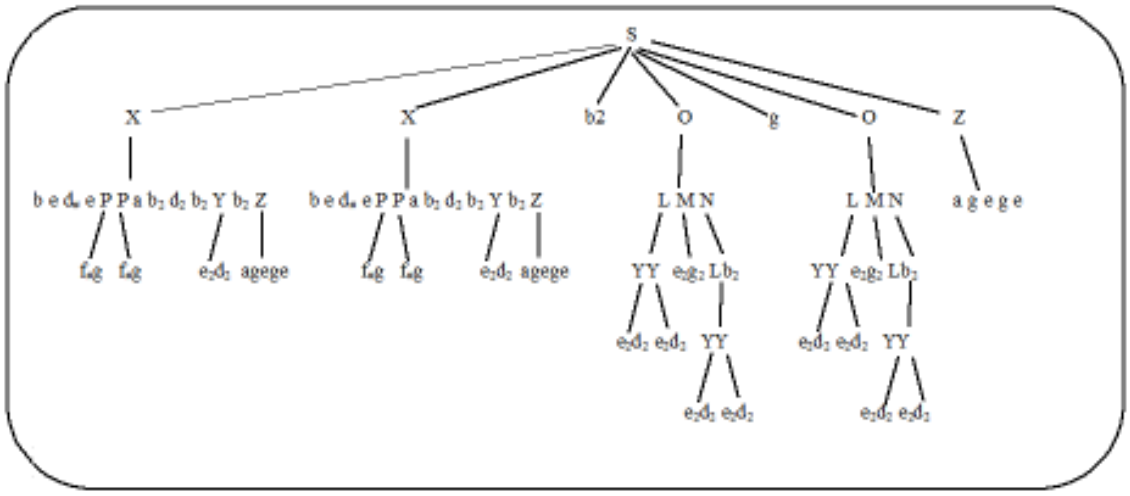

Fig. 13. Árbol sintáctico del Cóndor Pasa.

4.2.2.1 Reconocedor de melodías

Este módulo del modelo es el encargado de la generación de gramáticas y reconocimiento de melodías. El reconocimiento de melodías consiste en evaluar si una cadena de caracteres pertenece o no a cierto lenguaje descrito por una gramática. Este proceso se lleva a cabo mediante un *parser* que implementa un algoritmo del tipo *greedy*, recursivo descendente.

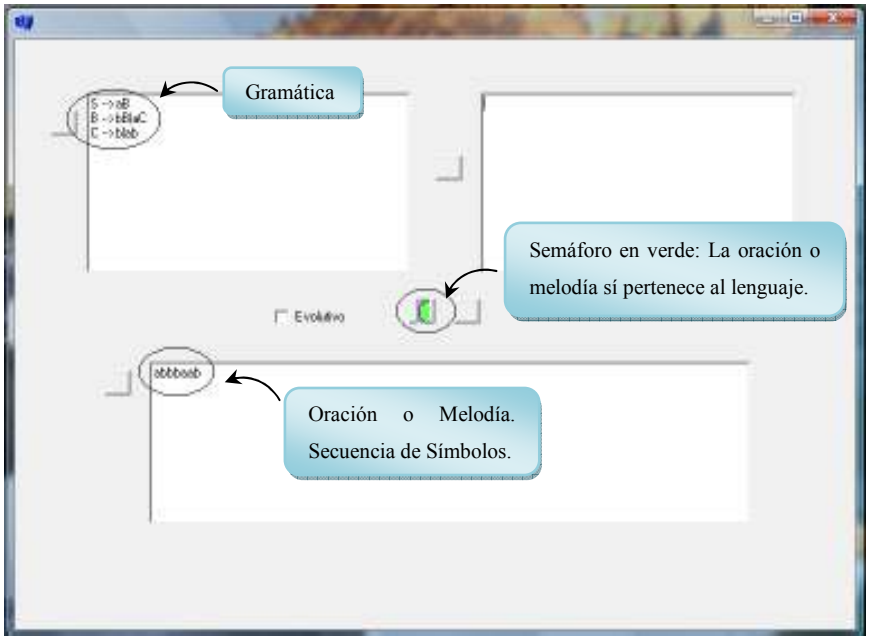

Fig. 14. Reconocedor de Melodías

En la Fig. 14 se muestra la aplicación del modelo en funcionamiento, en donde acepta o no la melodía de entrada de la ventana inferior de acuerdo a la gramática que se presenta en la

ventana superior izquierda. En el Anexo C se muestra el código del *parser* para reconocer si una melodía cumple o no con las reglas de la gramática.

4.2.2.2 Generador de gramáticas

El módulo generador de gramáticas, así como las funciones de aprendizaje de la matriz, permite que el modelo esté en contacto con la realidad y que evolucione de acuerdo a los cambios que se presenten con cada ejemplo nuevo. Esta propiedad marca una diferencia radical con algunos de los modelos existentes para la generación de música, que aunque algunos de ellos tienen la propiedad de evolucionar, se olvidan de la realidad.

Cada vez que se lee una cadena que no pertenece a la gramática se tiene la opción de generar una gramática para reconocer la nueva cadena. De esta forma, la gramática va evolucionando con cada cadena nueva que se da como entrada. En términos musicales significa que cada melodía nueva incrementa el conocimiento musical del modelo, dando posibilidades a la generación de música nueva. En la Fig. 15 se presenta cómo el modelo va generando la gramática de acuerdo a los ejemplos de oraciones que se le proporcionen.

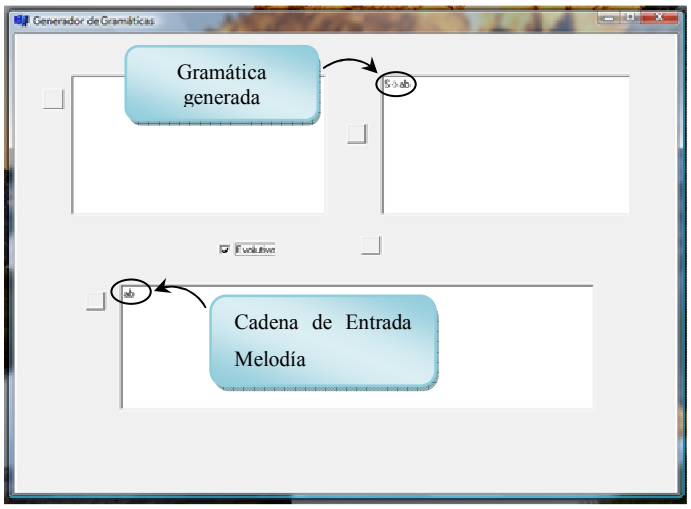

Fig. 15. Generación de una gramática a partir de cadenas de símbolos

En el generador de gramáticas se han implementado los casos en el que se agrega una cadena desconocida al final de la oración, en el que se agrega un no-terminal y otra opción a la regla de producción. La gramática se almacena en dos arreglos bidimensionales, uno de ellos se utiliza como la memoria del modelo. Los símbolos no-terminales pueden ser de

más de un caracter. En el Anexo D se presenta el código del módulo para la generación de gramáticas.

4.3 Expresividad

Con su teoría de causalidad Cochrane [2010] explica significado de la expresividad musical y la manera en la que provoca que aparezcan sentimientos y emociones.

La expresividad es un mecanismo de supervivencia que depende de la viveza con la que se transmitan y sean interpretados sentimientos y emociones. Por ejemplo, el miedo ante una amenaza. Intervienen factores físicos tales como la aceleración del ritmo cardiaco, cambios en el sistema respiratorio, endócrino, muscular, circulatorio, la secreción de neurotransmisores, etc. Como en toda comunicación, existe un emisor, un receptor y un medio de transmisión. Para que esta comunicación sea exitosa, depende de que el emisor transmita el mensaje correctamente y que el receptor lo entienda.

El significado depende de diferentes factores, como la experiencia del emisor y receptor ante las señales enviadas y recibidas, tales como las señales visuales como posturas y movimientos o sonoras, de acuerdo con la altura de las frecuencias y el tiempo con las que son emitidos los sonidos del mensaje.

Reconocer las emociones en la música tiene que ver con la manera en la que reconocemos emociones en otras personas, lo que hacemos basándonos, muchas veces, en información visual y acústica. En cuanto a la información visual y la relación que tiene con la música, se habla de una equivalencia entre el movimiento y variaciones en el volumen, ritmo, frecuencia, armonía. En cuanto a la información acústica la equivalencia es más directa. Tiene que ver con los cambios de tonalidad que se hacen al vocalizar las emociones. En el contorno de una pieza musical se pueden encontrar características que se encuentran al hablar, como por ejemplo, cuando se hace énfasis en una frase o la acentuación que se utiliza si se trata de una interrogación o una admiración.

Está más allá de nuestra investigación explicar cómo se hacen estos cambios físicos o cómo se produce la empatía entre los seres vivos. Nosotros simulamos la expresividad en la generación de música, determinando las funciones de probabilidad inherentes a la música.

4.3.1 La expresividad en nuestro modelo

Como se explicó a lo largo de nuestro trabajo, descomponemos la música en diferentes funciones que la caracterizan, tal como la frecuencia y el tiempo. Cada nota de una melodía es un símbolo acompañado de diversas características o descriptores semánticos, que le dan significado de ser un sonido largo, agudo, intenso, suave, de una guitarra o de un piano.

Con nuestro modelo es posible representar cada una de estas variables mediante matrices o gramáticas que reflejan el comportamiento probabilístico de cada una de ellas. En este artículo presentamos como modelar la frecuencia y el tiempo de las notas, que son dos de las variables de la expresividad musical, pero de la misma manera se puede construir la matriz de intensidades, por ejemplo. Mientras más variables, mayor expresividad denotará el modelo.

Una de las características de nuestro modelo es la capacidad de aprender a partir de ejemplos de música m_i. A partir de cada nuevo ejemplo m_i se generan las gramáticas probabilísticas g que describen por medio de sus reglas de producción, los patrones típicos que caracterizan los rasgos de expresividad musical de cada melodía. Estas reglas aprendidas, que contienen rasgos similares a las melodías aprendidas, se utilizan para generar música inédita m_{i+1} de manera automática.

Con nuestro modelo podemos caracterizar, de manera supervisada, diferentes tipos de música de acuerdo a sus rasgos de expresividad, por ejemplo en música alegre o triste. Además nos da la posibilidad de combinar diferentes tipos de música, por ejemplo, mezclar funciones de tiempos de música alegre con funciones de frecuencias de música triste. Incluso de diferentes géneros, por ejemplo frecuencias de música clásica con tiempos de rock. De esta manera, no sólo nos brinda la posibilidad de generar música, sino inventar nuevos géneros musicales.

CAPÍTULO 5

RESULTADOS

5.1 Metodología experimental

En la música existen diversos factores que dificultan evaluar si la música generada es por principio música y para continuar que tan buena música es. Así que para evaluar si nuestro modelo está generando música, decidimos llevar a cabo una prueba tipo Turing.

Pedimos a 26 participantes de la prueba que evaluaran la música generada de manera automática de acuerdo a su gusto, sin que ninguno de ellos supiera la forma en que había sido creada la música. Para ello, cinco melodías generadas de forma automática se mezclaron con cinco composiciones hechas por compositores humanos y se pidió a los participantes ordenar de acuerdo a su gusto las melodías escuchadas, numerándolas del 1 al 10, siendo el número 1 el correspondiente a la que más gustó y el número 10 a la que menos sin poder repetir ninguno de los números. Se reprodujeron los primeros 30 segundos de cada una de las melodías. Las 10 melodías se presentaron de acuerdo a la tabla 8:

Tabla 7. Orden en el que fueron presentadas las melodías a los participantes

ID	Melodía	Autor
A	Zanya	(generada)
B	Fell	Nathan Fake
C	Alucin	(generada)
D	Idiot	James Holden
E	Ciclos	(generada)
F	Dali	Astrix
G	Ritual Cibernetico	(generada)
H	Feelin' Electro	Rob Mooney
I	Infinito	(generada)
J	Lost Town	Kraftwerk

5.2 Resultados

En los gustos por cada melodía se encontraron calificaciones completamente diferentes, para algunas personas una misma melodía fue calificada como la que más les gustó mientras que para otras fue el décimo lugar. El resultado de esta prueba es alentador, ya que 2 de las melodías generadas automáticamente obtuvieron la tercera y cuarta posición en el gusto de los participantes, por arriba de melodías humanas de bandas famosas. La tabla 9 presenta el resultado de esta prueba.

Tabla 8. Orden de las melodías obtenido después de la prueba tipo Turing

ID	Ranking	Melodía	Autor
B	1	Fell	Nathan Fake
D	2	Idiot	James Holden
C	3	Alucín	(generada)
A	4	Zanya	(generada)
F	5	Dali	Astrix
H	6	Feelin' Electro	Rob Mooney
J	7	Lost Town	Kraftwerk
E	8	Ciclos	(generada)
G	9	Ritual Cibernético	(generada)
I	10	Infinito	(generada)

5.3 Discusión

La música generada que se utilizó para esta prueba se generó modelando únicamente la frecuencia, que constituye una de las variables de los rasgos de la música y sólo se han utilizado unos cuantos fragmentos de música como ejemplo. Hemos obtenido novedosos resultados comparables con los obtenidos por otros desarrollos que implementan algoritmos más complejos [Todd 1999] [Biles 2001].

La música resultante es por siempre diferente y de acuerdo a la opinión de músicos, las composiciones generadas son similares a los ejemplos utilizados. Sin embargo, aun estamos trabajando en el desarrollo de algoritmos que permitan modelar la estructura musical a la que está acostumbrado el oído humano y consideramos que los resultados mejorarán considerablemente si se proporciona una mayor cantidad de ejemplos y se utilizan espacios con un mayor número de dimensiones para modelar los otros rasgos musicales. En el anexo B se presenta la partitura de una melodía generada automáticamente.

En los experimentos realizados se refleja que las melodías generadas automáticamente están en el gusto de las personas por encima de composiciones humanas y que se confunden con composiciones humanas. Lo que nos permite afirmar que el modelo compositor logra el propósito de generar música automáticamente. Esta investigación está abierta para desarrollos futuros que permitan hacer que las composiciones musicales generadas automáticamente sean parte del gusto popular.

CAPÍTULO 6

CONCLUSIONES

Y TRABAJO FUTURO

6.1 Conclusiones

Con el fin de crear música automáticamente hemos desarrollado un modelo basado en matrices evolutivas. A partir de este modelo, desarrollamos una aplicación para la generación de música. Las matrices evolutivas son un conjunto de matrices que a través de ciertas operaciones se actualizan, manteniendo en constante expansión el conocimiento del sistema.

Utilizar matrices para la representación del conocimiento musical, simplifica la inferencia de las reglas de composición, en comparación con hacer inferencia gramatical directamente sobre las melodías. Las matrices evolutivas son equivalentes a gramáticas generativas.

Hemos modelado dos de las variables de la música y con ello, comprobado que nuestro modelo es eficaz para la simulación de la expresividad musical. Se puede extender para incluir más variables como la intensidad. También es importante el uso de diversos formatos musicales. El formato midi permite generar fácilmente las reglas de composición, ya que almacena información discreta. Un problema de este tipo de formato que es que en la mayoría de los archivos actuales no son utilizadas algunas variables como la intensidad, lo que provoca que la música sea menos expresiva. Esto está cambiando, pues los sintetizadores actuales ya son sensibles a la presión que se ejerza sobre sus teclas.

Las melodías generadas por nuestro sistema son innovadoras y al mismo tiempo reflejan patrones recordando a las melodías originales. El desarrollo de aplicaciones enfocadas a la composición musical, permitirá en corto tiempo que cualquier persona pueda crear música. Al mismo tiempo, ofrecerá a los músicos herramientas que los apoyen en la composición de sus obras musicales.

6.2 Aportaciones

- Hemos propuesto un modelo evolutivo para representar un lenguaje. La creación de música es nuestro objetivo principal y una aplicación de nuestro modelo.
- Hemos propuesto una forma de representación del conocimiento musical, con base en matrices evolutivas [Galindo 1998].
- Hemos desarrollado un modelo que está en constante aprendizaje, expandiendo su conocimiento para la generación de música. Con cada composición musical aprendida, el sistema se reestructura para adaptarse al conocimiento nuevo.
- Hemos desarrollado un sistema que no necesita reglas predefinidas, las genera a través de aprendizaje no supervisado y aproximación pobre en conocimiento.
- Hemos propuesto un modelo para la creación de música con la originalidad e innovación que estamos buscando y por otro lado, que refleja los patrones aprendidos de la música humana.
- Hemos desarrollado un algoritmo para transformar una matriz en una gramática generativa probabilística, con lo que enfatizamos el enfoque lingüístico sobre el que está basado nuestro trabajo.

6.3 Trabajo futuro

El trabajo futuro en el corto plazo, tiene que ver con caracterizar diferentes tipos de música, desde música triste hasta alegre, desde clásica hasta electrónica, con el fin de determinar funciones probabilísticas que nos permitan generar cualquiera de estos géneros de música. Estamos desarrollando sistemas utilizando matrices y gramáticas más de las variables que intervienen en una obra musical, con el fin de mejorar la expresividad musical de las composiciones generadas.

También tenemos en mente el desarrollo de compositores polifónicos, ampliando el significado de las unidades semánticas. Dentro del trabajo a corto plazo se prevé la utilización de otros formatos como el wav y mp3.

Las personas que han escuchado nuestras melodías las consideran música y a muchas de ellas les gustan. Ejemplos de música generada por nuestro sistema se encuentran en: www.olincuicatl.com.

6.4 Publicaciones

- García Salas, A. 1999. "Zanya Compositor Evolutivo de Música Virtual". Polibits. Año X Vol. 1 Num. 21. Centro de Innovación y Desarrollo Tecnológico en Computación. México. 1999.
- García Salas, A., and Gelbukh, A. 2008. "Music Composer Based on Detection of Typical Patterns in a Human Composer's Style." Memorias del XXIV Simposio Internacional de Computación en la Educación. Xalapa Veracruz, México.
- García Salas, A., Gelbukh, A., and Calvo, H. 2010. "Music Composition Based on Linguistic Approach." Proceedings of the 9th Mexican International Conference on Artificial Intelligence. Pachuca, México. pp. 117-128.
- García Salas, A., Gelbukh, A., Calvo, H., and Galindo Soria, F. 2011. "Automatic Music Composition with Simple Probabilistic Generative Grammars." Polibits Revista de Investigación y Desarrollo Tecnológico en Computación. Num. 44. Centro de Innovación y Desarrollo Tecnológico en Computación. México. pp. 59-65.

6.5 Conferencias

- Kit para Desarrollo de Software Musical. Concurso de Prototipos. Escuela Superior de Cómputo del Instituto Politécnico Nacional. México D.F. Julio 1997.
- Kit para Desarrollo de Software Musical. 1era Muestra de Inventiva y Creatividad. Centro de Estudios Tecnológicos Industrial y de Servicios No. 7. Dirección General de Educación Tecnológica Industrial. México D.F. Noviembre 1997.
- Aplicación de los Sistemas Evolutivos a la Composición Musical. V Semana Nacional de Ciencia y Tecnología. Centro de Estudios Tecnológicos Industrial y de

Servicios No. 7. Dirección General de Educación Tecnológica Industrial. México D.F. Octubre 1998.

- 7ª Semana de Informática. Llave Primaria del Nuevo Milenio. Instituto Tecnológico de Tlanepantla. Tlalnepantla de Baz, Estado de México. Abril 2000.
- Zanya Sistema Evolutivo para la Composición Musical. XIV Aniversario del TESE. Tecnológico de Estudios Superiores de Ecatepec. Ecatepec de Morelos, Estado de México. Septiembre 2004.
- Compositor Evolutivo de Música Virtual. 4to Congreso Internacional de Investigación. Centro de Investigación Fernando Galindo Soria A.C. Marzo 2008.
- Generador Automático de Música. Conferencia Magistral. 1er Foro de Desarrollo de Proyectos de Investigación de la MISC. Tecnológico de Estudios Superiores de Ecatepec. Ecatepec de Morelos, Estado de México. Julio 2008.
- Musical Composer Based on Detection of Typical Patterns in a Human Composer's Style. XXIV Simposio Internacional de Computación en la Educación. Sociedad Mexicana de Computación en la Educación A.C. Xalapa de Enríquez, Veracruz. México. Octubre 2008.
- Zanya Compositor Evolutivo de Musica Virtual. XXIV Simposio Internacional de Computación en la Educación. Sociedad Mexicana de Computación en la Educación A.C. Xalapa de Enríquez, Veracruz. México. Octubre 2008.
- Sistema Compositor Musical basado en Aprendizaje Supervisado. Sexto Taller de Tecnologías del Lenguaje Humano. Instituto Nacional de Astrofísica, Óptica y Electrónica. Sta. María Tonantzintla, Puebla. México. Octubre 2009.
- Music Composition based on Linguistic Approach. The 9th Mexican International Conference on Artificial Intelligence. Pachuca de Soto, Hidalgo. November 2010.

REFERENCIAS

[Aylett 2004] Ruth S. Aylett. Agents and Affect: Why Embodied Agents Need Affective Systems. Lecture Notes in Computer Science. Springer Berlin/Heidelberg. ISSN 0302-9743 (Print) 1611-3349 (Online). Volume 3025/2004. Methods and Applications of Artificial Intelligence. DOI 10.1007/b97168. Copyright 2004. ISBN 978-3-540-21937-8. Páginas 496-504. Subject Collection. Informática. Fecha de SpringerLink jueves, 01 de abril de 2004.

[Aymeric *et al.* 2004] Zils Aymeric, Pachet François. Automatic Extraction of Music Descriptors From Acoustic Signals. Proceedings of the 116th AES Convention, May 2004. Sony CSL Paris.

[Biles 2001] John A. Biles. GenJam: Evolution of a jazz improviser. Source. Creative evolutionary systems. Section: Evolutionary music. Pages: 165 – 187. Year of Publication: 2001. Publisher Morgan Kaufmann Publishers Inc. San Francisco, CA, USA.

[Birchfield 2003] David Birchfield. Generative model for the creation of musical emotion, meaning and form. Source International Multimedia Conference. Proceedings of the 2003 ACM SIGMM workshop on Experiential telepresence. Berkeley, California SESSION: Playing experience. Pages: 99 – 104. Year of Publication: 2003. ISBN:1-58113-775-3. Publisher ACM New York, NY, USA.

[Blackburm *et al*. 1998] Steven Blackburm and David DeRoure. A tool for content based navigation of music. Source International Multimedia Conference. Proceedings of the sixth ACM international conference on Multimedia. Bristol, United Kingdom. Pages: 361 – 368. Year of Publication: 1998. ISBN:0-201-30990-4

[Blackwell 2007] Tim Blackwell. Swarming and Music. Evolutionary Computer Music. Springer London. DOI 10.1007/978-1-84628-600-1. 2007. ISBN 978-1-84628-599-8 (Print) 978-1-84628-600-1 (Online) DOI 10.1007/978-1-84628-600-1_9. Páginas 194-217. Subject Collection Informática. Fecha de SpringerLink viernes, 12 de octubre de 2007.

[Brey 2003] Barry B. Brey. The Intel Microprocesors 8086/8088, 80186, 80286, 80386 and 80486. Architecture Programming and Interfacing. Ed. Prentice Hall. DeVry Institute of Technology, Columbus.

[Bui *et al*. 2002] The Duy Bui, Dirk Heylen, Mannes Poel, and Anton Nijholt. ParleE: An Adaptive Plan Based Event Appraisal Model of Emotions. Lecture Notes in Computer Science. Springer Berlin/Heidelberg. ISSN 0302-9743 (Print) 1611-3349 (Online). Volume 2479/2002. KI 2002: Advances in Artificial Intelligence. DOI 10.1007/3-540-45751-8. Copyright 2002. ISBN 978-3-540-44185-4. DOI 10.1007/3-540-45751-8_9. Páginas 1-9. Subject Collection Informática. Fecha de SpringerLink martes, 01 de enero de 2002.

[Bulmer 2000] Michael Bulmer. Music From Fractal Noise. University of Queensland. *Proceedings* of the Mathematics 2000 Festival, Melbourne, 10 – 13 January 2000.

[Cochrane 2010] Tom Cochrane. 2010. A Simulation Theory of Musical Expressivity. The Australasian Journal of Philosophy, Volume 88, Issue 2, p.191-207.

[Eck *et al*. 2002] Douglas Eck, Jüurgen Schmidhuber. A First Look at Music Composition using LSTM Recurrent Neural Networks. Source Technical Report: IDSIA-07-02. Year of Publication: 2002. Publisher Istituto Dalle Molle Di Studi Sull Intelligenza Artificiale

[Egges *et al*. 2003]. Arjan Egges, Sumedha Kshirsagar, and Nadia Magnenat-Thalmann. A Model for Personality and Emotion Simulation. Lecture Notes in Computer Science. Springer Berlin/Heidelberg. ISSN 0302-9743 (Print) 1611-3349 (Online). Volume 2773/2003. Knowledge-Based Intelligent Information and Engineering Systems. DOI 10.1007/b12002. Copyright 2003. ISBN 978-3-540-40803-1. Páginas 453-461. Subject Collection Informática. Fecha de SpringerLink viernes, 10 de octubre de 2003.

[Essa *et al*. 1997] Irfan A. Essa and Alex P. Pentland. Coding, Analysis, Interpretation, and Recognition of Facial Expressions. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, July 1997.

[Feng *et al*. 2000] G. C. Feng and Pong C. Yuen. Recognition of Head-&-Shoulder Face Image Using Virtual Frontal-View Image. IEEE Transactions on Systems, Man, and Cybernetics—Part a: Cybernetics, Vol. 30, No. 6, November 2000.

[Francisco *et al*. 2005] Virginia Francisco GilMartín, Pablo Gervás Gómez-Navarro, Raquel Hervás Ballesteros. Análisis y síntesis de expresión emocional en cuentos leídos en voz alta. Procesamiento del lenguaje natural. Nº 35 Sept. 2005, pp. 293-300. ISSN 1135-5948.

[Galindo 1991] Fernando Galindo Soria. Sistemas Evolutivos: Nuevo Paradigma de la Informática. Memorias XVII Conferencia Latinoamericana de Informática, Caracas Venezuela, julio de 1991.

[Galindo 1995] Fernando Galindo Soria. 1995. Enfoque Lingüístico. Memorias del Simposio Internacional de Computación de 1995. Cd. de México: Instituto Politécnico Nacional CENAC.

[Galindo 1998] Fernando Galindo Soria. 1998. Matrices Evolutivas. Memorias de la Cuarta Conferencia de Ingeniería Eléctrica CIE/98. Cd. de México: Instituto Politécnico Nacional CINVESTAV, 8: pp. 17-22.

[Gamow 1959] Materia Tierra y Cielo. George Gamow. Compañía Editorial Continental S.A. México 1959.

[Gardner 1978] Martin Gardner. Música blanca y música parda, curvas fractales y fluctuaciones del tipo 1/f. Investigación y Ciencia, junio 1978.

[Hild *et al*. 1992] Hermann Hild, Advances Johannes Feulner, Wolfram Menzel. Harmonet: A Neural Net for Harmonizing Chorales in the Style of J.S.Bach. in in Neural Information Processing 4 (NIPS 4), pp. 267-274, R.P. Lippmann, J.E. Moody, D.S. Touretzky (eds.), Morgan. Kaufmann.Universität Karlsruhe, Germany.

[Jackson 1987] John V. Jackson. ACM SIGART Bulletin Issue 101 (July 1987) Pages: 23–26. Year of Publication: 1987. ISSN:0163-5719.

[Järveläinen 2000] Hanna Järveläinen. Algorithmic Musical Composition. April 7, 2000. Tik-111.080 Seminar on content creation Art@Science. Helsinki University of Technology Laboratory of Acoustics and Audio Signal Processing.

[Kosina 2002] Karin Kosina. Music Genre Recognition.. Diplomarbeit. Eingereicht am Fachhochschul-Studiengang. Mediente Chnik Und Design in Hagenberg. Im Juni 2002.

[Lee 1999] Kwangyong Lee. Integration of various emotion eliciting factors for life-like agents. International Multimedia Conference. Proceedings of the seventh ACM international conference on Multimedia. Orlando, Florida, United States. Pages: 155 – 158. Year of Publication: 1999. ISBN:1-58113-239-5.

[Legaspi 2007] Roberto Legaspi Yuya Hashimoto Koichi Moriyama Satoshi Kurihara Masayuki Numao. Music Compositional Intelligence with an Affective Flavor. International Conference on Intelligent User Interfaces. Proceedings of the 12th international conference on Intelligent user interfaces. Honolulu, Hawaii, USA. SESSION: Multi-modal interfaces. Pages: 216 – 224. Year of Publication: 2007. ISBN:1-59593-481-2.

[López 2002] Rubén López Cano. Entre el giro lingüístico y el guiño hermenéutico: tópicos y competencia en la semiótica musical actual. Revista Cuicuilco Volumen 9, No. 25, mayo-agosto 2002. Número especial: Análisis del discurso y semiótica de la cultura: perspectivas analíticas para el tercer milenio. Tomo II.

[Lu *et al*.2003] Lie Lu, Hong-Jiang Zhang, Stan Z. Li. Content-based audio classification and segmentation by using support vector machines. Multimedia Systems. Springer Berlin/Heidelberg. ISSN 0942-4962 (Print) 1432-1882 (Online). Volume 8, Number 6 / abril de 2003. Regular paper. DOI 10.1007/s00530-002-0065-0. Páginas 482-492. Subject Collection Informática. Fecha de SpringerLink martes, 01 de abril de 2003.

[Mántaras *et al*. 2006] Ramon López de Mántaras, Maarten Grachten, Josep-Lluís Arcos. A Case Based Approach to Expressivity-Aware Tempo Transformation. Source Machine Learning Volume 65 , Issue 2-3 (December 2006). Pages: 411 – 437. Year of Publication: 2006. ISSN:0885-6125. Publisher Kluwer Academic Publishers Hingham, MA, USA.

[McAlpine *et al*. 1999] Kenneth McAlpine, Eduardo Miranda, Stuart Hoggar. Making Music with Algorithms: A Case-Study System. Computer Music Journal Volume 23, Issue 2 (Summer 1999) Pages: 19 – 30. Massachusetts Institute of Technology.

[McCauley 1998] Lee McCauley, Stan Franklin. An Architecture for Emotion. AAAI Technical Report FS-98-03. Compilation copyright © 1998. Institute for Intelligent Systems. The University of Memphis.

[Melucci *et al*. 1999] Massimo Melucci and Nicola Orio. Musical information retrieval using melodic surface. International Conference on Digital Libraries Proceedings of the fourth ACM conference on Digital libraries. Berkeley, California, United States. Pages: 152 – 160. Year of Publication: 1999. ISBN:1-58113-145-3. Padua University Department of Electronics and Computing Science. ACM New York, NY, USA.

[Meyer 1956] Leonard B. Meyer. Emotion and Meaning in Music. Chicago: Chicago University Press.

[Miranda *et al*. 2006] Eduardo R. Miranda, Jesus L. Alvaro, Beatriz Barros. Music Knowledge Analysis: Towards an Efficient Representation for Composition. Springer Berlin/Heidelberg ISSN 0302-9743 (Print) 1611-3349 (Online). Volume 4177/2006. Current Topics in Artificial Intelligence. DOI 10.1007/11881216. ISBN 978-3-540-45914-9. Selected Papers from the 11th Conference of the Spanish Association for Artificial Intelligence (CAEPIA 2005). DOI 10.1007/11881216_ 35. Páginas 331-341. Subject Collection Informática. Fecha de SpringerLink viernes, 13 de octubre de 2006.

[Mota *et al*. 2003]. Selene Mota and Rosalind W. Picard. Atomated Posture Analysis for detecting Learner's Interest Level. Conference on Computer Vision and Pattern Recognition Workshop. Volume 5, 2003, pp.49. Publication Date: 16-22 June 2003. ISSN: 1063-6919, ISBN: 0-7695-1900-8. Digital Object Identifier: 10.1109/CVPRW.2003.10047. Current Version Published: 2008-09-12.

[Namunu *et al*. 2004]. Namunu C Maddage, Changsheng Xu, Mohan S Kankanhalli, Xi Shao. Content-based music structure analysis with applications to music semantics understanding. Source International Multimedia Conference. Proceedings of the 12th annual ACM international conference on Multimedia. SESSION: Technical session 3: audio processing. Pages: 112 – 119. Year of Publication: 2004. ISBN:1-58113-893-8. Publisher ACM New York, NY, USA.

[Napolitano 2001] Antonio Napolitano. Universidad, Estado, Sociedad. Anales de la Universidad Metropolitana Vol. 1, N° 2, 2001: 153-169. Departamento de Humanidades Universidad Metropolitana.

[Ortega *et al*. 2002] Alfonso Ortega de la Puente, Rafael Sánchez Alfonso, Manuel Alfonseca Moreno. Automatic composition of music by means of Grammatical Evolution. ACM SIGAPL APL. Volume 32, Issue 4 (June 2002) Pages: 148 – 155. Year of Publication: 2002. ISSN:0163-6006. Publisher ACM New York, NY, USA.

[Pantic *et al*. 2000] Maja Pantic and Leon J.M. Rothkrantz. Automatic Analysis of Facial Expressions: The State of the Art. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, no. 12, December 2000.

[Papadopoulos 1999] George Papadopoulos, Geraint Wiggins. AI Methods for Algorithmic Composition: A Survey, a Critical View and Future Prospects. AISB Symposium on Musical Creativity 1999, pages [110-117]. School of Artificial Intelligence, Division of Informatics, University of Edinburgh.

[Pearce 2002] Marcus Pearce, David Meredith and Geraint Wiggins. Motivations and Methodologies for Automation of the Compositional Process. Musicæ Scientiæ Journal 2002 Volume 6. Department of Computing, City University, London.

[Peña 2005] Peña Guerrero Maximino. Captura de Múltiples Eventos MIDI en Tiempo de Ejecución. Tesis doctoral, Cinvestav IE. México D.F. abril del 2005.

[Picard *et al*. 2001] Rosalind W. Picard, Elias Vyzas, and Jennifer Healey. Toward Machine Emotional Intelligence: Analysis of Affective Physiological State. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23, No. 10, October 2001.

[Picard 2003] R.W. Picard. Affective Computing: Challenges. International Journal of Human-Computer Studies, Volume 59, Issues 1-2, July 2003, pp. 55-64 MIT Media Laboratory Cambridge, MA USA

[Salomaa 1987] Arto Salomaa. Formal Languages. Academic Press, New York, 1973. Revised edition in the series "Computer Science Classics", Academic Press, 1987.

[Tatai *et al*. 2003] Gábor Tatai, Annamária Csordás, Attila Szaló, László Laufer. The Chatbot Feeling – Towards Animated Emotional ECAs. Lecture Notes in Computer Science. Springer Berlin/Heidelberg. ISSN 0302-9743 (Print) 1611-3349 (Online). Volume 2902/2003. Progress in Artificial Intelligence. DOI 10.1007/b94425. Copyright 2003. ISBN 978-3-540-20589-0. Multi-Agents and AI for the Internet (MAAII). Páginas 336-340. Subject Collection Informática. Fecha de SpringerLink jueves, 06 de noviembre de 2003.

[Tian *et al*. 2001] Ying-li Tian, Takeo Kanade, and Jeffrey F. Cohn. Recognizing Action Units for Facial Expression Analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23, No. 2, February 2001.

[Todd 1999] Peter M. Todd & Gregory M. Werner. Frankensteinian Methods for Evolutionary Music Composition. Musical networks. Pages: 385. Year of Publication: 1999. ISBN:0-262-07181-9. Editors. Niall Griffith, Peter M. Todd. Publisher MIT Press Cambridge, MA, USA.

[Velásquez *et al*. 1997] Juan D. Velásquez, Pattie Maes. Cathexis: a computational model of emotions. International Conference on Autonomous Agents. Proceedings of the first international conference on Autonomous agents. Marina del Rey, California, United States. Pages: 518 – 519. Year of Publication: 1997. ISBN:0-89791-877-0.

[Velázquez 1999] Roberto Velázquez Cabrera. Analisis de Aerófonos Mexicanos. Conferencia para el Congreso Internacional de Computación CIC-99. IPN, México, noviembre de 1999.

[Yacoob et al. 1996] Yaser Yacoob and Larry S. Davis. Recognizing Human Facial Expressions From Long Image Sequences Using Optical Flow. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 18, No. 6, June 1996.

Sitios en Internet

[Bakshee *et al*. 1994] Igor S. Bakshee and Toshimitsu Musha. Random Series in Computer Graphics. http://www.mathematica-journal.com/issue/v4i1/graphics/09-10_galleryText.mj.pdf. Fecha de Consulta 30/V/2007

[Castro 2005] Verónica Castro. Entrevista a Robert Zatorre: La música y su relación con el cerebro. Entrevista 11-05-2005 http://portal.educ.ar/noticias/entrevistas/robert-zatorre-la-musica-y-su-1.php

[Galindo 2004] Fernando Galindo Soria. Representación de la Estructura Profunda del Ruido 1/F mediante la Ecuación de la Naturaleza.
http://www.fgalindosoria.com/ecuaciondelanaturaleza/ruido_colores_y_ec_naturaleza/ruido_colores_mediante_ecuacion_naturaleza.pdf

[López 2008] Noemí López. Escuchar, Mirar, Leer y Pensar Música. Octubre 08, 2008. Etiquetas: Antigüedad Griega, Instrumentos. http://escucharmirarleerypensarmusica.blogspot.com/2006/10/la-msica-en-la-antigedad-griega.html

[Miramontes 1999] El color del ruido. Pedro Miramontes. Facultad de Ciencias, UNAM, México. Abril-Junio 1999. http://www.ejournal.unam.mx/cns/no54/CNS05401.pdf

[MusicDef] Definition of music. http://en.wikipedia.org/wiki/Definition_of_music. Fecha de Consulta 8/XI/2007

[Nagore 2004] María Nagore. El Análisis Musical, entre el Formalismo y la Hermenéutica. Músicas al Sur - Número 1 - Enero 2004. Universidad Complutense de Madrid. http://www.eumus.edu.uy/revista/nro1/nagore.html

[Wisniewski 1996] Joseph S. Wisniewski. The Colors of Noise. http://www.ptpart.co.uk/show.php?contentid=71. Fecha de Consulta 1/VI/2007

ANEXO A

COMPOSICIÓN MUSICAL

MODELANDO DIFERENTES TIPOS DE SEÑALES

Una manera de describir el mundo sonoro es mediante señales, que son la suma de diversas ondas que dan a cada sonido una característica particular. Existen sonidos agradables y aquellos considerados como ruidos. Los ruidos se han estudiado desde diferentes puntos de vista. En las antiguas culturas mesoamericanas se construían diversos tipos de instrumentos, que se desarrollaban no con el objetivo de producir notas armoniosas utilizadas para hacer composición musical, sino para generar ruidos [Velázquez 1999]. Así que, no sólo estudiaban los ruidos sino que se utilizaban de una manera práctica. Actualmente, los ruidos también se utilizan para hacer música.

Con base en investigaciones hechas por R. F. Voss [Gardner 1978], los ruidos no sólo se manifiestan de manera desagradable al oído humano, ya que música como el jazz, el blues y el rock tiene características del ruido 1/f. Por lo que, propone un algoritmo para generar ruido 1/f con el fin de hacer composición musical.

Para saber la relación que existe entre las diferentes frecuencias que forman parte de una señal y su amplitud, se utiliza una herramienta conocida como *espectro de frecuencia*, que indica la manera en la que están distribuidas las frecuencias en la señal. Ciertas señales presentan una distribución de frecuencias en su espectro igual a $1/f^n$, por ejemplo el ruido blanco ($1/f^0$), el ruido pardo (browniano $1/f^2$) y el ruido rosa (1/f) [Miramontes 1999] [Wisniewski 1996]. Una manera de hacer composición musical de forma automática es modelar estas señales.

A.1 Ruido blanco

Se denomina ruido blanco a aquel que presenta una dimensión espectral = $1/f^0 = 1$ [Gardner 1978]. El ruido blanco tiene una distribución de frecuencias en la que todas las frecuencias que lo componen aparecen con la misma probabilidad y no existe ninguna dependencia entre ellas. Un ruido blanco se puede producir mediante una sucesión de frecuencias generadas de manera aleatoria.

En la Fig. 16, se muestra un ejemplo de música blanca graficada en dos dimensiones: Se utilizan dos variables *notas* y *tiempo,* en donde *notas* representa las ordenadas y *tiempo* las abscisas. Se obtuvo escogiendo al azar entre 9 notas y el tiempo se ha considerado constante.

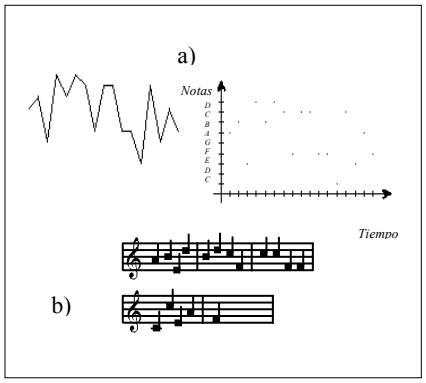

Fig. 16. (a) Gráfica de música blanca. (b) Partitura de esta melodía

En la Fig. 17 se muestra el pseudocódigo de una función con la que se puede generar música blanca, utilizando escalas en Do mayor numeradas del 1 al 60.

```
          1   2 ...                              ... 60
notas {C, D, E, F, G, A, B, C,...., E, F}
main()
{
  While(no ESC)
  {
     sound( notas[ random(60) +1]) ;
     delay(200);          //en milisegundos
  }
  nosound();
}
```

Fig. 17. Pseudocódigo para la generación de música blanca

A.2 Ruido browniano o pardo

El ruido browniano tiene la característica de ser altamente correlacionado y presenta una dimensión espectral = $1/f^2$. Un ruido browniano es una sucesión de sonidos en donde la

frecuencia siguiente depende fuertemente de la frecuencia anterior. El ruido browniano es un ejemplo de sistemas donde a partir de eventos erráticos se obtienen resultados ordenados con un alto grado de correlación, cuando la correlación es muy baja o no existe, se pueden obtener sistemas desordenados como es el caso del ruido blanco.

Una manera de obtener música parda es generando una secuencia de notas musicales, en donde la nota siguiente sea escogida entre un rango de 3 notas más altas o 3 notas más bajas a partir de la nota anterior. La Fig. 18, muestra una gráfica donde se ve el inicio de una melodía parda. El tiempo en este ejemplo se considera constante.

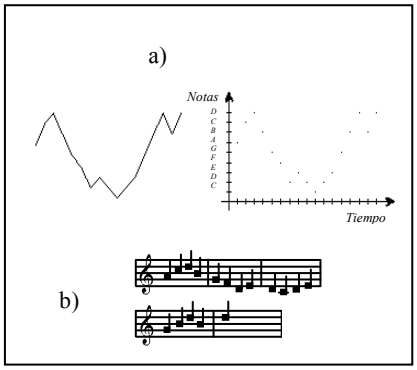

Fig. 18. (a) Gráfica de música parda. (b) Partitura de esta melodía

En la Fig. 19 se presenta el pseudocódigo de este algoritmo para generar música browniana, con la característica de que uno de los valores que se puede obtener para la siguiente nota es cero, que significa que se puede repetir la nota. La música compuesta de esta forma en algunos fragmentos es parecida a la música del vuelo del abejorro de Nikolai Rimsky Korsakov (1844-1908).

```
          1   2 ...                              ... 60
notas {C, D, E, F, G, A, B, C,...., E, F}

main()
{
  Nota=random(60+1);
  While( no ESC)
  {
    Nota= random(7)-3;
    sound( notas[Nota] );
    delay(200);          //en milisegundos
  }
  nosound();
}
```

Fig. 19. Pseudocódigo para la generación de música parda

Hicimos que se genere una gráfica de acuerdo con éste algoritmo. La Fig. 20 es la gráfica tomada de la pantalla de la computadora, que se genera al ejecutar este algoritmo. Las diferentes longitudes de las líneas, así como los ángulos sobre los que se trazan, dependen del valor de la nota.

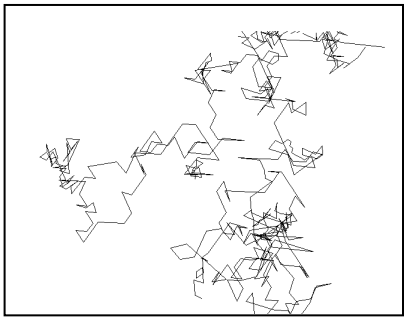

Fig. 20. Gráfica que se genera con líneas de acuerdo a las notas de la melodía

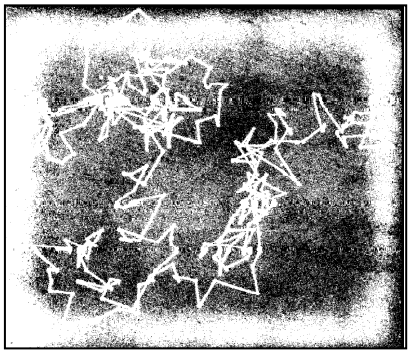

Fig. 21. Posiciones sucesivas de una partícula de humo

La Fig. 21 es una fotografía de las posiciones sucesivas de una partícula de humo registradas en intervalos de 1 minuto, en donde se puede apreciar el movimiento browniano. Fue obtenida del libro uno, dos, tres, infinito [Gamow 1959]. En las Fig. 20 y 21 se puede comparar el modelo y la realidad, en donde realmente se está comparando un sonido con una imagen que representa las distintas posiciones de una partícula.

A.3 Ruido rosa (1/f)

El ruido rosa se encuentra en una gran cantidad de fenómenos del Universo [Bakshee *et al*. 1994], como el flujo del tránsito, el crecimiento de las ciudades, el crecimiento de los árboles, la aparición de las manchas solares, el crecimiento de las montañas, el movimiento de las olas del mar, la formación de galaxias, la formación de nubes, etc. que presentan un comportamiento con densidad espectral alrededor de 1/f.

El caso de la música no es la excepción, de acuerdo a estudios realizados por Voss y comentados en [Gardner 1978], música como la clásica, el jazz y el rock presentan características de ruido 1/f. La música 1/f no es tan desordenada como la música blanca, pero tampoco es tan correlacionada como la parda, lo que provoca que no sea tan monótona. Esto quiere decir que en una melodía rosa la secuencia de notas puede contener notas con distancias pequeñas y de repente encontrar saltos entre notas más distantes. Voss desarrolló un algoritmo para generar música 1/f y en [Galindo 2004] se hace un análisis de su estructura que permite visualizarlo como un árbol sintáctico. En la Fig.22 se muestra un fragmento de una composición rosa.

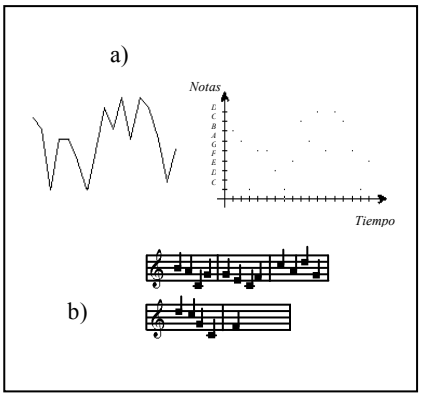

Fig. 22. (a) Gráfica de ruido 1/f y (b) Partitura de esta melodía

En la Fig. 23 se muestra el pseudocódigo para generar música 1/f sobre un arreglo de 60 notas con 10 dados que pueden tomar valores de 0 a 6.

```
          1   2  ...                                  ...60
notas {C, D, E, F, G, A, B, C, D, E,...., E, F}
avienta_dado [10]
numero_binario = 0000000000
numero_binario2 = 1111111111
main()
{
  While( No Esc )
  {
     for( nn=0; nn<=9; nn++ )
     {
        if( bit nn de numero_binario != bit nn de numero_binario2 )
           avienta_dado[nn] = random(7);
        Nota += avienta_dado[nn];
     }
     sound( notas[Nota]);
     delay(200);                        //en milisegundos
     numero_binario2 = numero_binario;
     numero_binario ++;
  }
  nosound();
}
```

Fig. 23. Pseudocódigo para la generación de ruido 1/f

ANEXO B

GRÁFICAS TRIDIMENSIONALES

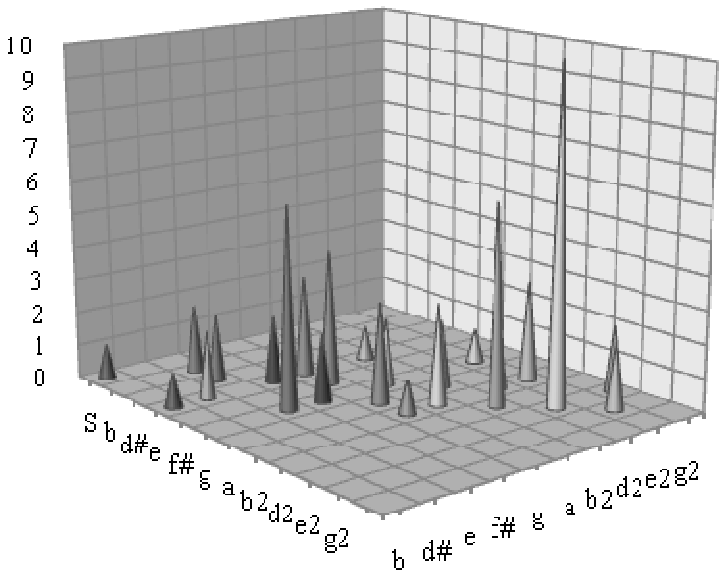

Fig. 24. Matriz de distribución de frecuencias FDM en 3D

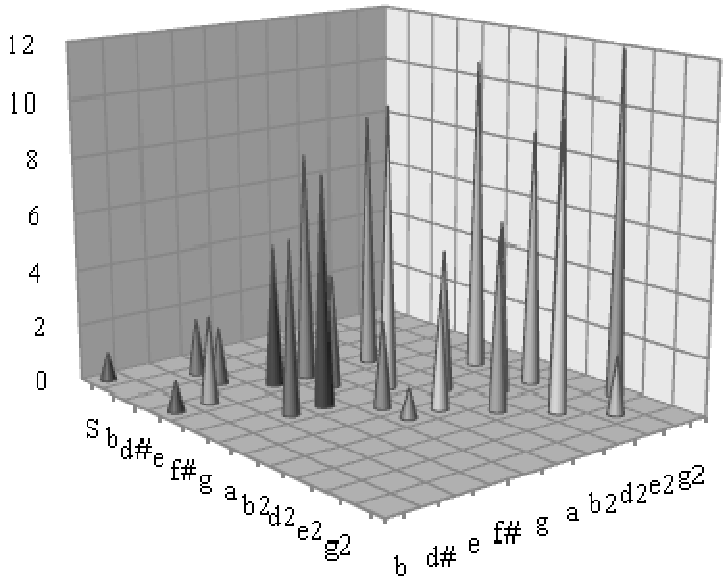

Fig. 25. Matriz de distribución de frecuencias acumuladas CFM en 3D

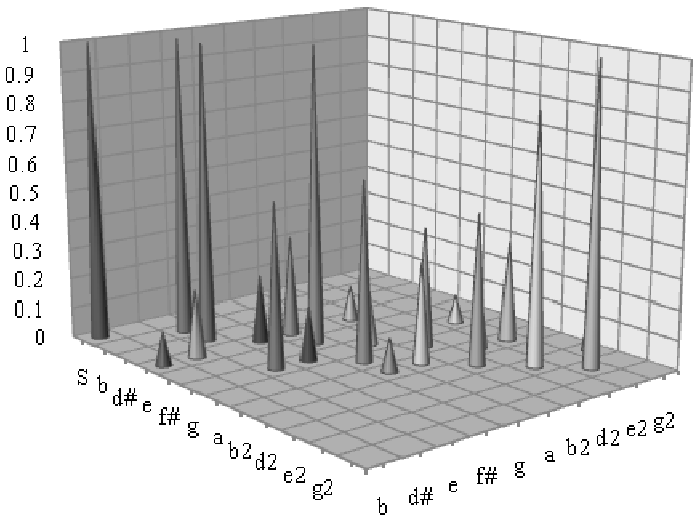

Fig. 26. Matriz de probabilidades PM en 3D

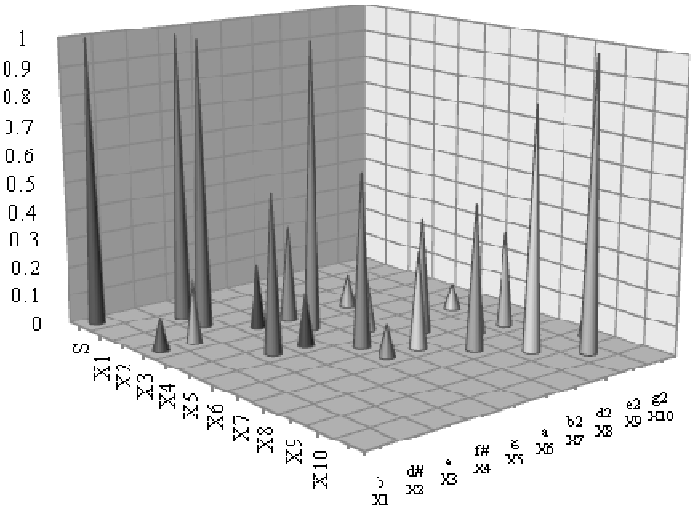

Fig. 27. Matriz Auxiliar AM en 3D

ANEXO C

CÓDIGO RECONOCEDOR DE MELODÍAS

```
void G( int yy )
{
int xx=0;
int xrule=0, xgrcpy=0;
char Rule[33]="\0";

 while( !error && Gram[yy][xx] != '\0'  && Gram[yy][xx] != '|' )
 {
  if( Melodia[xme] == Gram[yy][xx] )
  {
   xx++;                                          // Si el caracter leido de la melodía es igual
   xme++;                                         // que el terminal de la gramática lee el siguiente
   if( Gram[yy][xx] == '\0' || Gram[yy][xx] == '|' )
   {
    xg=xx;
    yg=yy;
   }
  }
  else
   if( Gram[yy][xx] == 'Æ' )                      //Si es un símbolo no-terminal ejecuta de
   {                                              //de forma recursiva esta función
    xx++;
    while( Gram[yy][xx] != '£' )
     Rule[xrule++] = Gram[yy][xx++];
    xgrcpy=xx;
    xrule=0;
    xx=atoi(Rule);
    G( xx );
    xx=xgrcpy+1;
   }
   else
    if( Gram[yy][xx-1] == '\x0' || Gram[yy][xx-1] == '|' )
    {
     if( xx = BuscaOr( yy, xx ) )                 //Si no es ni terminal, ni no-terminal
      xx++;                                       //busca alguna alternativa
     else
     {
      error=1;
      xg=xx;
      yg=yy;
     }
    }
    else
    {
     error=1;                                     //Si no es terminal, no no-terminal y no
     xg=xx;                                       //hay opciones, entonces marca error
     yg=yy;
    }
 }
}
```

ANEXO D

CÓDIGO GENERADOR DE GRAMÁTICAS

```
void EvGram()
{
int xx=0,yy=0,xx2=0,yyvn=0,xxvn=0, xxcpy=0;
char Rule[33]="\0";
int xrule=0;

 xx=xg;
 yy=yg;
 if( yg==0 && xg==0 )
 {
  yy++;
  Vn[1][0]='S';                          //Si no existe agrega el símbolo raíz
 }
 else
  if( error==1 )
  {
      while( Vn[yyvn][0] != '\0' )
        yyvn++;
      while( Gram[yy][xx] != '\0' )
        Gram[yyvn][xx2++]=Gram[yy][xx++];
      xxcpy=xx;
      Gram[yyvn][xx2++] = '|';
      xx=xg;
      Gram[yy][xx++] = 'Æ';
      Borra(Rule);
      itoa(yyvn,Rule,10);                //Agrega un símbolo no-terminal
      while( Rule[xrule] != '\x0' )      //en las reglas de producción
        Gram[yy][xx++] = Rule[xrule++];
      Gram[yy][xx++] = '£';
      xrule=0;
      while( xx < xxcpy )
        Gram[yy][xx++] = '\0';
      xx=xx2;
      yy=yyvn;
      Vn[yyvn][xxvn++] = 'Æ';  //alt 146.
      while( Rule[xrule] != '\x0' )      //Agrega un símbolo no-terminal
        Vn[yyvn][xxvn++]=Rule[xrule++];  //del lado izquierdo de las reglas
      Vn[yyvn][xxvn++] = '£';  //alt 156. //de producción
      xrule=0;
 }
 while(Melodia[xme] != '\0')             //Agrega los nuevos terminales
  Gram[yy][xx++] =  Melodia[xme++];
 while( GramCpy[yy][xg] != '\x0' )       //Copia el resto de la gramática
  Gram[yy][xx++] = GramCpy[yy][xg++];
}
```

ANEXO E

TABLA DE FRECUENCIAS MIDI

Octava -5

Nombre de Octava	Octava MIDI	Número de Nota MIDI	Nombre de Nota	Frequencia Hz
--	-5	0	C	8.1757989156
--	-5	1	$C^{\#}/D^{b}$	8.6619572180
--	-5	2	D	9.1770239974
--	-5	3	$D^{\#}/E^{b}$	10.3008611535
--	-5	4	E	10.3008611535
--	-5	5	F	10.9133822323
--	-5	6	$F^{\#}/G^{b}$	11.5623257097
--	-5	7	G	12.2498573744
--	-5	8	$G^{\#}/A^{b}$	12.9782717994
--	-5	9	A	13.7500000000
--	-5	10	$A^{\#}/B^{b}$	14.5676175474
--	-5	11	B	15.4338531643

Octava -4

Nombre de Octava	Octava MIDI	Número de Nota MIDI	Nombre de Nota	Frequencia Hz
--	-4	12	C	16.3515978313
--	-4	13	$C^{\#}/D^{b}$	17.3239144361
--	-4	14	D	18.3540479948
--	-4	15	$D^{\#}/E^{b}$	19.4454364826
--	-4	16	E	20.6017223071
--	-4	17	F	21.8267644646
--	-4	18	$F^{\#}/G^{b}$	23.1246514195
--	-4	19	G	24.4997147489
--	-4	20	$G^{\#}/A^{b}$	25.9565435987
--	-4	21	A	27.5000000000
--	-4	22	$A^{\#}/B^{b}$	29.1352350949
--	-4	23	B	30.8677063285

Octava -3

Nombre de Octava	Octava MIDI	Número de Nota MIDI	Nombre de Nota	Frequencia Hz
--	-3	24	C	32.7031956626
--	-3	25	$C^{\#}/D^{b}$	34.6478288721
--	-3	26	D	36.7080959897
--	-3	27	$D^{\#}/E^{b}$	38.8908729653
--	-3	28	E	41.2034446141
--	-3	29	F	43.6535289291
--	-3	30	$F^{\#}/G^{b}$	46.2493028390
Low	-3	31	G	48.9994294977
Low	-3	32	$G^{\#}/A^{b}$	51.9130871975
Low	-3	33	A	55.0000000000
Low	-3	34	$A^{\#}/B^{b}$	58.2704701898
Low	-3	35	B	61.7354126570

Octava -2

Nombre de Octava	Octava MIDI	Número de Nota MIDI	Nombre de Nota	Frequencia Hz
Low	-2	36	C	65.4063913251
Low	-2	37	$C^{\#}/D^{b}$	69.2956577442
Low	-2	38	D	73.4161919794
Low	-2	39	$D^{\#}/E^{b}$	77.7817459305
Low	-2	40	E	82.4068892282
Low	-2	41	F	87.3070578583
Low	-2	42	$F^{\#}/G^{b}$	92.4986056779
Bass	-2	43	G	97.9988589954
Bass	-2	44	$G^{\#}/A^{b}$	103.8261743950
Bass	-2	45	A	110.0000000000
Bass	-2	46	$A^{\#}/B^{b}$	116.5409403795
Bass	-2	47	B	123.4708253140

Octava -1

Nombre de Octava	Octava MIDI	Número de Nota MIDI	Nombre de Nota	Frequencia Hz
Bass	-1	48	C	130.8127826503
Bass	-1	49	$C^{\#}/D^{b}$	138.5913154884
Bass	-1	50	D	146.8323839587

Bass	-1	51	$D^{\#}/E^{b}$	155.5634918610
Bass	-1	52	E	164.8137784564
Bass	-1	53	F	174.6141157165
Bass	-1	54	$F^{\#}/G^{b}$	184.9972113558
Middle	-1	55	G	195.9977179909
Middle	-1	56	$G^{\#}/A^{b}$	207.6523487900
Middle	-1	57	A	220.0000000000
Middle	-1	58	$A^{\#}/B^{b}$	233.0818807590
Middle	-1	59	B	246.9416506281

Octava 0

Nombre de Octava	**Octava MIDI**	**Número de Nota MIDI**	**Nombre de Nota**	**Frequencia Hz**
Middle	0	60	C	261.6255653006
Middle	0	61	$C^{\#}/D^{b}$	277.1826309769
Middle	0	62	D	293.6647679174
Middle	0	63	$D^{\#}/E^{b}$	311.1269837221
Middle	0	64	E	329.6275569129
Middle	0	65	F	349.2282314330
Treble	0	66	$F^{\#}/G^{b}$	369.9944227116
Treble	0	67	G	391.9954359817
Treble	0	68	$G^{\#}/A^{b}$	415.3046975799
Treble	0	69	A	440.0000000000
Treble	0	70	$A^{\#}/B^{b}$	466.1637615181
Treble	0	71	B	493.8833012561

Octava 1

Nombre de Octava	**Octava MIDI**	**Número de Nota MIDI**	**Nombre de Nota**	**Frequencia Hz**
Treble	1	72	C	523.2511306012
Treble	1	73	$C^{\#}/D^{b}$	554.3652619537
Treble	1	74	D	587.3295358348
Treble	1	75	$D^{\#}/E^{b}$	622.2539674442
Treble	1	76	E	659.2551138257
Treble	1	77	F	698.4564628660
High	1	78	$F^{\#}/G^{b}$	739.9888454233
High	1	79	G	783.9908719635
High	1	80	$G^{\#}/A^{b}$	830.6093951599

High	1	81	A	880.0000000000
High	1	82	$A^{\#}/B^{b}$	932.3275230362
High	1	83	B	987.7666025122

Octava 2

Nombre de Octava	Octava MIDI	Número de Nota MIDI	Nombre de Nota	Frequencia Hz
High	2	84	C	1,046.5022612024
High	2	85	$C^{\#}/D^{b}$	1,108.7305239075
High	2	86	D	1,174.6590716696
High	2	87	$D^{\#}/E^{b}$	1,244.5079348883
High	2	88	E	1,318.5102276515
High	2	89	F	1,396.9129257320
--	2	90	$F^{\#}/G^{b}$	1,479.9776908465
--	2	91	G	1,567.9817439270
--	2	92	$G^{\#}/A^{b}$	1,661.2187903198
--	2	93	A	1,760.0000000000
--	2	94	$A^{\#}/B^{b}$	1,864.6550460724
--	2	95	B	1,975.5332050245

Octava 3

Nombre de Octava	Octava MIDI	Número de Nota MIDI	Nombre de Nota	Frequencia Hz
--	3	96	C	2,093.0045224048
--	3	97	$C^{\#}/D^{b}$	2,217.4610478150
--	3	98	D	2,349.3181433393
--	3	99	$D^{\#}/E^{b}$	2,489.0158697766
--	3	100	E	2,637.0204553030
--	3	101	F	2,793.8258514640
--	3	102	$F^{\#}/G^{b}$	2,959.9553816931
--	3	103	G	3,135.9634878540
--	3	104	$G^{\#}/A^{b}$	3,322.4375806396
--	3	105	A	3,520.0000000000
--	3	106	$A^{\#}/B^{b}$	3,729.3100921447
--	3	107	B	3,951.0664100490

Octava 4

Nombre de Octava	Octava MIDI	Número de Nota MIDI	Nombre de Nota	Frequencia Hz

--	4	108	C	4,186.0090448096
--	4	109	$C^{\#}/D^{b}$	4,434.9220956300
--	4	110	D	4,698.6362866785
--	4	111	$D^{\#}/E^{b}$	4,978.0317395533
--	4	112	E	5,274.0409106059
--	4	113	F	5,587.6517029281
--	4	114	$F^{\#}/G^{b}$	5,919.9107633862
--	4	115	G	5,919.9107633862
--	4	116	$G^{\#}/A^{b}$	6,644.8751612791
--	4	117	A	7,040.0000000000
--	4	118	$A^{\#}/B^{b}$	7,458.6201842894
--	4	119	B	7,902.1328200980

Octava 5

Nombre de Octava	Octava MIDI	Número de Nota MIDI	Nombre de Nota	Frequencia Hz
--	5	120	C	8,372.0180896192
--	5	121	$C^{\#}/D^{b}$	8,869.8441912599
--	5	122	D	9,397.2725733570
--	5	123	$D^{\#}/E^{b}$	9,956.0634791066
--	5	124	E	10,548.0818212118
--	5	125	F	11,175.3034058561
--	5	126	$F^{\#}/G^{b}$	11,839.8215267723
--	5	127	G	12,543.8539514160

ANEXO F

MELODÍA GENERADA AUTOMÁTICAMENTE

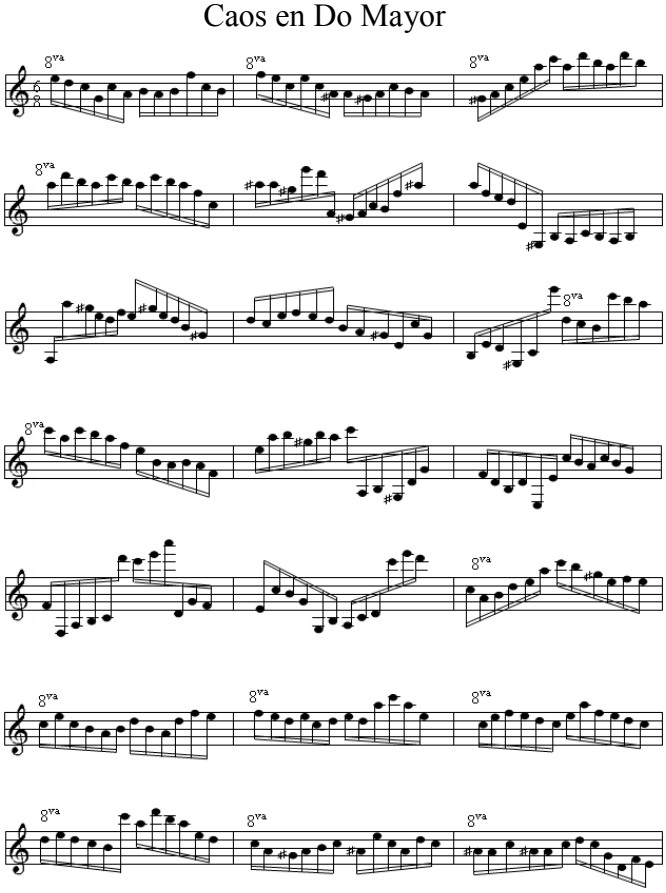

Fig. 28. Composición generada automáticamente sobre la base de 4 fragmentos de Paganini