

15806-121
FE915-00'



CINVESTAV-IPN
Biblioteca de Ingeniería Eléctrica



FB0000013929

CENTRO DE INVESTIGACION Y DE
ESTUDIOS AVANZADOS DEL
I. P. N.
BIBLIOTECA
INGENIERIA ELECTRICA



CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS AVANZADOS DEL I.P.N.

**DEPARTAMENTO DE INGENIERÍA ELÉCTRICA
SECCIÓN COMPUTACIÓN**

TEMA DE TESIS:

GENERACIÓN AUTOMÁTICA DE MÚSICA

PARA OBTENER EL GRADO DE:

MAESTRO EN CIENCIAS EN LA ESPECIALIDAD DE INGENIERÍA ELÉCTRICA

PRESENTA:

ING. SALVADOR ERNESTO ZIZUMBO CASTAÑEDA

ASESOR Y DIRECTOR DE TESIS:

DR. SERGIO VÍCTOR CHAPA VERGARA

**CENTRO DE INVESTIGACION Y DE ESTUDIOS AVANZADOS DEL I. P. N.
BIBLIOTECA INGENIERIA ELECTRICA**

XM

CLASIF.	22. #
ADQUIS.	P1-1482
FECHA	17 JUN 2002
PROCD.	17-22
\$	

Resumen: Esta tesis muestra la generación automática de una pieza musical mediante el uso de métodos formales que permiten modelar el lenguaje musical de una manera conveniente para ser tratado por la computadora. Se estudian conceptos de armonía tonal moderna con la finalidad de que las piezas musicales posean un cierto grado de belleza. Además, con el propósito enriquecer las características expresivas de las piezas, éstas pueden reescribirse utilizando un conjunto de reglas armónicas. Se desarrolló un sistema que permite realizar la composición, reescribir las piezas, mostrar la partitura y permitir su lectura y almacenamiento en un archivo con formato MIDI.

CENTRO DE INVESTIGACION Y DE
ESTUDIOS AVANZADOS DEL
I. P. N.
BIBLIOTECA
INGENIERIA ELECTRICA

Abstract: This thesis shows the automatic generation of a musical piece by means of the use of formal methods that allows to model the musical language in a convenient manner in order to be used with the computer. Concepts of modern harmony are studied with the objective that the musical pieces possess a certain degree of beauty. Also, with the purpose to enrich the expressive characteristics of the pieces, these can be rewritten using a group of harmonics rules. A system has been developed in order to perform the piece composition, rewriting and display of the score and also to allow the reading and storage in a MIDI file.

CENTRO DE INVESTIGACION Y DE
ESTUDIOS AVANZADOS DEL
I. P. N.
BIBLIOTECA
INGENIERIA ELECTRICA

Índice de Contenido

Capítulo 1. Introducción	3
1.1 Entorno de la generación automática de música	3
1.2 Inteligencia artificial en la música	5
1.3 Presentación del problema	6
Capítulo 2. Conceptos Básicos de Música	9
2.1 Signos principales	9
2.2 Signos secundarios	16
2.3 Principios generales de la ejecución musical	20
Capítulo 3. Armonía	25
3.1 Tonalidad	25
3.2 Relaciones de las escalas	28
3.3 Acordes y su clasificación	29
Capítulo 4. Lenguajes Formales y Gramáticas	37
4.1 Conceptos básicos	37
4.2 Gramáticas	39
4.3 Gramáticas y la generación de música	47
4.4 Comentarios finales	48
Capítulo 5. Conceptos de Automatas	49
5.1 Automatas finitos determinísticos y no-determinísticos	49
5.2 Automatas probabilísticos	53
5.3 Automatas para la generación de música	55
5.4 Comentarios finales	56
Capítulo 6. Unificación	57
6.1 Lógica	57
6.2 Unificación y sistemas de reescritura	62
6.3 Unificación y arreglo de piezas musicales	65
6.4 Comentarios finales	66
Capítulo 7. Generación Automática de Música	67
7.1 Presentación de la propuesta	67
7.2 Diseño de la aplicación	76
7.3 Implementación y pruebas	81

Agradezco a:

Dios, por su amor, bendiciones y fidelidad.

Mi madre, Noemi Castañeda Ocaranza, por su amor, apoyo y paciencia.

Mi familia y amigos, por estar siempre conmigo.

El CONACyT por el apoyo económico para la realización de mis estudios de maestría.

El CINVESTAV-IPN México, por los beneficios académico / económicos que me permitieron realizar mis estudios de maestría.

El Dr. Sergio V. Chapa Vergara por su asesoría para la elaboración de este trabajo de tesis.

La Sra. Sofía Reza por el apoyo secretarial durante mi estancia en el CINVESTAV-IPN.

Capítulo 1. Introducción

La generación automática de música es una de las áreas actuales de investigación en computación y en desarrollos tecnológicos.

Los musicólogos motivados fuertemente por crear nuevas formas de música se han acercado cada vez más a la tecnología. Esto se aprecia observando la evolución de la música a lo largo del tiempo.

1.1 Entorno de la generación automática de música

Los comienzos de la música se desconocen. Según los mitos de los pueblos, la música es de origen divino. La idea occidental de música se remonta a la antigua Grecia, así como a culturas avanzadas de Asia Menor y del Lejano Oriente. Su ámbito original era el culto. Posteriormente se convirtió en un arte de expresión estética. Y hasta el día de hoy, los músicos han transformado constantemente los géneros y formas de la música.

La música de nuestra era tiene su origen en la salmodia y los himnos (siglos I-IV), sobresaliendo el salmo judío y la música del ámbito cultural helénico del mediterráneo. Esta música fue evolucionando y en los siglos XV y XVI se manifiesta un desarrollo ininterrumpido en cuanto a tendencias y técnicas de composición musical. El punto central lo constituye la *música vocal polifónica*, representada por el contrapunto (del latín *punctum contra punctum, punto contra punto, nota contra nota*), donde las voces tienen una dimensión melódica *horizontal*, mientras que, al sonar simultáneamente, tienen una dimensión armónica *vertical*.

En el período comprendido aproximadamente entre 1600 y 1750 surge el *Barroco*, música representada por la ópera y la fuga, considerada confusa en lo armónico, llena de disonancias, de dificultad melódica, poco natural, desigual.

Surge después el *Clasicismo*, que en la historia de la música se conoce como la época y el estilo de los tres grandes maestros vieneses: Haydn, Mozart y Beethoven. Este período fue motivado por la perfección de la forma. *Clásico* significa en general lo ejemplar, bello, lleno de proporción y armonía, además de sencillo y comprensible. En el *Clasicismo* se suprime la compleja armonía del barroco, así como la polifonía contrapuntista. Todo se centra ahora en la melodía, donde el hombre se expresa sencilla y naturalmente. Los ritmos evitan el contraste en un espacio reducido.

El siglo XIX se considera el siglo del *Romanticismo*. El *Romanticismo* surgió sin ruptura con el *Clasicismo*, aunque introdujo un nuevo elemento poético, predominando la expresión del "yo". La armonía romántica es continuación de la clásica por medio del cromatismo, las alteraciones y las enarmonías, que llegan hasta el límite con la atonalidad (tonalidad fluctuante). El *Romanticismo* da prioridad a sonidos cercanos a la naturaleza.

El siglo XX cultiva una variedad de estilos en la música mucho mayor que la de cualquier otro período anterior. Después de la 2ª Guerra Mundial surge un gran interés por la novedad, destacando el jazz, la música aleatoria o experimental y la música electrónica.

El jazz tuvo su origen en la comunidad negra de Nueva Orleans, Louisiana, combinando la tradición afro-americana, los cantos religiosos y la música de la población blanca con las bandas musicales y las orquestas de baile. El jazz posee las siguientes características:

- *Blue notes*: Terceras y séptimas son tanto mayores como menores.
- *Off-Beat*: Son desviaciones del ritmo regular; esto se conoce como síncopas.
- *Polifonía*: Los instrumentos melódicos varían, adornan y juegan con la melodía, improvisando en función de la tesitura, estilo y temperamento.

La música aleatoria o experimental tiene como sus representantes a John Cage y Pierre Boulez, quienes hacían énfasis en las formas abiertas o acciones espontáneas donde existe una gran indeterminación; Boulez combina la tradición de Debussy, Webern y Messiaen con la modernidad de su época. Los compositores de esta corriente consideran que el azar no destruye la obra, sino que la amplía.

La música electrónica surgió hacia 1950, y consiste en sonidos y composiciones generados electrónicamente. Su principal representante es Karlheinz Stockhausen. En este tipo de música el trabajo del compositor consiste en la fabricación del material (tonos sinusoidales), su transformación y sincronización. El resultado es la cinta magnética, que hace innecesario al intérprete y a la partitura. La composición consideraba 81 tonos sinusoidales de 100-17200 Hz. En una nueva fase, la música electrónica se interpreta en el escenario, donde el músico interactúa con la música generada electrónicamente grabada en cinta. La aparición de la computadora ayudó a los compositores a generar sonidos apegados a las frecuencias de los tonos de las escalas. Xenakis trabajó con computadoras para generar música estocástica [17], aumentando el interés por hacer uso de la computadora como una herramienta para los compositores.

1.2 Inteligencia artificial en la música

En los últimos años ha crecido el interés por lograr que las computadoras realicen tareas que originalmente se pensaba podrían llevarse a cabo únicamente por humanos, éste es el caso de la composición musical.

La inteligencia artificial se organiza principalmente en dos partes [14]:

- Una parte *científica*, también conocida como ciencia cognitiva, enfocada al desarrollo de teorías sobre la inteligencia humana.
- Una parte *de aplicación*, enfocada al desarrollo de programas que exhiben un comportamiento inteligente.

En el campo de la Inteligencia Artificial la generación automática de música es uno de los principales retos por las características y variantes que la música posee.

Los proyectos sobre música abarcan tanto la parte científica como la parte de aplicación. Un sistema de generación musical puede incluir la construcción de un programa que produzca una partitura a partir de un "lenguaje" musical conocido [14]. Otros sistemas de generación musical únicamente se desarrollaron en forma teórica y todavía no se implementan, como los propuestos por Lerdaahl y Jackendoff [22].

Institutos como el MIT, han dedicado parte de su investigación a la obtención del conocimiento musical mediante sistemas basados en conocimientos [23], donde se buscan patrones en la música. Otros proyectos se enfocaron a la ejecución musical usando el enfoque de representación de restricciones [24].

1.3 Presentación del problema

Hasta hoy han sido propuestas diversas técnicas para la generación automática de música haciendo uso de fractales [10] [11] [13], de probabilidad aleatoria o de probabilidad condicional [12]. Los trabajos que se basan en el uso

de fractales proponen la investigación de la música por medio de una representación visual de señales acústicas. Los trabajos que consideran la probabilidad aleatoria ponderan el uso de cada uno de los tonos de la escala cromática. Los que se basan en probabilidad condicional, calculan la probabilidad de los tonos y su duración considerando los tonos y duraciones precedentes. En estas propuestas no se consideran modos ni grados de la tonalidad, por lo que los resultados no son satisfactorios desde el punto de vista musical.

El trabajo aquí propuesto difiere de los anteriores al considerar reglas de armonía tonal moderna. Se considera el modo de la tonalidad, los modos de cada grado de la tonalidad y la relación dinámica que existe entre ellos, llamada progresión. Existen diversos tipos de progresiones que pueden utilizarse al componer música, en este trabajo se hace uso de la progresión del círculo de quintas.

Este trabajo de tesis tiene los siguientes objetivos:

- Crear un modelo para el lenguaje musical, utilizando métodos formales para el tratamiento de lenguajes, en el que las piezas generadas se apeguen a las reglas armónicas que permitan que la pieza sea estéticamente correcta. Dado el enfoque que se dio al trabajo, el sistema se basó en gramáticas y autómatas probabilísticos.
- Adicionalmente se usan recursos armónicos que permiten enriquecer las características expresivas de los grados armónicos involucrados en la pieza generada. Se puede considerar este proceso de enriquecimiento de la pieza musical como un sistema de reescritura que pasará de una forma de representación a otra aplicando reglas de armonía.
- La implementación de un sistema capaz de utilizar el modelo creado, de generar automáticamente una pieza, de desplegar la partitura de la pieza y

almacenarla en un archivo MIDI estándar para ser reproducido o modificado por alguna otra aplicación.

Los seis capítulos restantes de esta tesis se han estructurado de la siguiente forma:

El segundo capítulo presenta conceptos básicos de la teoría musical para entender los elementos que forman una pieza musical y cómo expresarla de forma escrita.

En el capítulo 3 se analizan los conceptos relacionados con la armonía: la tonalidad, los principales tipos de acordes y las relaciones de estructura y de función que poseen.

El cuarto capítulo trata conceptos generales de lenguajes formales, distintos tipos de gramáticas generativas determinísticas y estocásticas, así como su aplicación para la generación automática de música.

El capítulo 5 resume los conceptos básicos de autómatas, citando las definiciones y características de los tipos de autómatas que se utilizaron para realizar este trabajo.

El sexto capítulo menciona algunos temas de unificación, sistemas de reescritura y su uso para el arreglo de piezas musicales.

En el capítulo 7 se describe la aplicación desarrollada. Se incluye además el código que describe las clases y métodos principales.

Capítulo 2. Conceptos Básicos de Música

Para leer notación musical es necesario conocer su sistema de representación: los signos con que se escribe y la interpretación de éstos indicada en las leyes que los rigen. El estudio de estos signos y leyes son el objeto de la Teoría de la Música. Debido a que la notación musical es muy amplia, se tratarán sólo los elementos más comunes: signos principales, signos secundarios, la tonalidad, el compás o medida y algunos principios generales de la ejecución musical. Los adornos y abreviaturas no serán considerados.

2.1 Signos principales

Pentagrama (pauta). El pentagrama es la renglonadura de cinco líneas rectas, paralelas y equidistantes utilizada para escribir música. Se compone de 5 líneas y 4 espacios. Las líneas y los espacios se numeran de abajo hacia arriba (Figura 1). Esta organización expresa un orden en la tonalidad que va de tonos graves a tonos agudos.

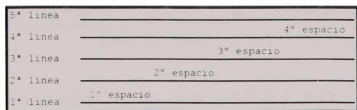


Figura 1 El pentagrama

Una pieza musical puede escribirse en uno o más pentagramas. Dos o más pentagramas forman un sistema y las notas de cada pentagrama se tocan simultáneamente; lo cuál se indica con unas llaves llamadas *acoladas* que unen los pentagramas del sistema.

Las Notas. Las notas expresan, según su figura, diferente duración. Las figuras están formadas por el cuerpo de la nota, opcionalmente una plica y ésta a su vez opcionalmente corchetes (Figura 2). Las figuras de nota y su duración relativa se muestran en la Tabla 1. Según su colocación en el pentagrama las notas indican un tipo específico de sonido.

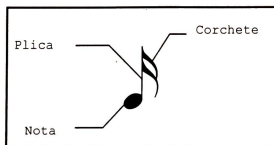


Figura 2 Partes de la nota

Redonda o unidad	Blanca o mitad	Negra o cuarto	Corchea u octava	Doble corchea o dieciseisavo	Triple corchea o treintaidosavo	Cuádruple corchea o sesentaicuatrosavo
○	◐	◑	♩	♪	♫	♬
	♪	♩	♫	♬	♯	♮
4/4	1/2	1/4	1/8	1/16	1/32	1/64

Tabla 1 Valor de duración de las notas musicales

La redonda representa la más larga duración y cada una de las figuras restantes representará la mitad de la figura que antecede y, por consiguiente, el doble de aquella que la sigue. La Tabla 2 muestra algunas equivalencias entre las notas.

Notas menores que las cuádruples corcheas (quíntuple corchea, etc.), generalmente no se tocan debido a su brevedad.













Nota	Equivale a ...		o equivale a...		o equivale a...	
		2 blancas		4 negras		8 corcheas
		2 negras		4 corcheas		8 dobles corcheas
		2 corcheas		4 dobles corcheas		8 triples corcheas

Tabla 2 Equivalencias entre notas

Cuando varias corcheas, dobles corcheas, triples corcheas, o cuádruples corcheas se colocan yuxtapuestas, se pueden reemplazar los corchetes por barras que unirán dichas figuras, esto dependerá de la métrica de la pieza musical (posteriormente se explicará la relación entre la métrica y las notas). El número de barras debe ser igual al número de corchetes que éstas reemplacen en cada figura, de manera que para escribir corcheas se pondrá una barra, para dobles corcheas dos barras y así sucesivamente.

Las notas, sea cual fuere su figura, se colocan en el pentagrama ya sea en las líneas o en los espacios, debajo de la primera línea o arriba de la quinta línea. Se pueden escribir también otras notas arriba o debajo del pentagrama empleando unas líneas pequeñas llamadas *líneas auxiliares* o *suplementarias* (Figura 4), el número de estas pequeñas líneas no está limitado. Las notas colocadas en el pentagrama, de abajo para arriba, indican los sonidos que van de lo grave a lo agudo.

Son siete los nombres de las notas que expresan todos los sonidos, estas notas forman una serie de sonidos que se utilizan para ir de lo grave a lo agudo (léase de izquierda a derecha) y se llama *serie ascendente*, invirtiendo el orden de estas notas se obtendrá una serie que va de lo agudo a lo grave y se llamará *serie descendente*.

Do Re Mi Fa Sol La Si

Se puede agregar una segunda serie a la primera, después una tercera etc. Se llama *octava* a la distancia que separa dos notas del mismo nombre, cuyas dos notas pertenecen a dos series vecinas (Figura 3).

El total de sonidos puede ser descrito como una sucesión infinita de series (ascendentes o descendentes). Teóricamente una pieza musical puede incluir sonidos de cualquier serie, es decir, hacer uso de todo el universo de sonidos.



Figura 3 Series y Octavas

Varias notas pueden ocupar la misma posición horizontal en el pentagrama pero diferente posición vertical, formando un *acorde*. Todas las notas en un acorde, se tocan simultáneamente (Figura 4).

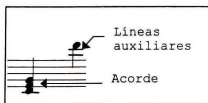


Figura 4 Acordes y líneas auxiliares

Llaves. Las llaves son símbolos que se colocan al principio del pentagrama. Su objeto es fijar el nombre de las notas y al mismo tiempo el lugar que ellas mismas ocupan en la escala musical.

Hay tres clases de llaves, sus figuras se muestran en la Tabla 3:

- Llave de Fa, se coloca en la 3ª y en la 4ª línea del pentagrama.
- Llave de Do, se coloca en la 1ª, en la 2ª, en la 3ª y en la 4ª línea.
- Llave de Sol, se coloca en la 1ª y en la 2ª línea.

<i>Llave de Fa</i>	<i>Llave de Do</i>	<i>Llave de Sol</i>
		

Tabla 3 Clases de llaves ¹

Cada una de estas llaves da su nombre a la nota colocada en la misma línea que ella ocupa. Conocido el nombre de una nota, es fácil encontrar el nombre de las otras notas puesto que siempre se suceden en el orden indicado en la serie.

Escala musical (diapasón musical). Es la reunión de todos los sonidos que puede apreciar el oído, desde el más grave hasta el más agudo, que pueden ser emitidos por voces o por instrumentos.

Se divide esta escala o diapasón en tres partes principales, cada una de estas partes toma el nombre de *registro*.

- El *registro grave* que comprende los más graves (1^{er} tercio de la escala).
- El *registro agudo* que comprende los más agudos (3^{er} tercio de la escala).
- El *registro medio* que comprende los intermedios, más agudos que los registros graves y más graves que los registros agudos (2^o tercio de la escala).

¹ La llave de Fa que se coloca en la 3ª línea, la llave de Do que se coloca en la 2ª línea y la llave de Sol que se coloca en la 1ª línea no se usan actualmente, sólo sirven para el cambio de tonalidad (transporte).

La escala musical, tiene una gran extensión y no puede escribirse en un pentagrama si no es con la ayuda de un gran número de líneas suplementarias. Para facilitar este inconveniente se inventaron las llaves por medio de las cuales se pueden colocar en el pentagrama los diferentes registros de la escala musical.

Para asignar a cada sonido un rango determinado en la escala musical, está convenido escoger un sonido que sirviendo de norma o punto de partida, permita fijar la relación de todos los otros sonidos entre sí. Este sonido es el tono *La*.

Aplicación de las llaves a los instrumentos. Existe una relación entre las llaves. La llave de Fa 4ª línea produce los sonidos más graves en la escala, sus registros producen sonidos cada vez más agudos, las diferentes llaves se suceden en el orden siguiente: la llave de Fa 3ª línea, la llave de Do 4ª línea, la de Do 3ª línea, la de Do 2ª línea, la de Do 1ª línea, la de Sol 2ª línea y por último la de Sol 1ª línea.

La música para piano, órgano y arpa se escribe sobre 2 pentagramas. El pentagrama inferior, sobre la cual se escribe la llave de Fa en 4ª línea, sirve para los sonidos graves (se tocan generalmente con la mano izquierda). El pentagrama superior, sobre la cual se escribe la llave de Sol 2ª línea, sirve para los sonidos agudos (que se tocan generalmente con la mano derecha). Se unen estos dos pentagramas por medio de un signo que se llama *acolada* y se coloca al principio de las dos pautas.

Existe además otra clase de *acolada* que se usa solamente en la partición (descrita a continuación), y que sirve para unir dos o más pentagramas que pertenecen a instrumentos de una misma especie o familia, o también para varias partes corales.

Se llama *partición* a la reunión de todas las partes de una pieza de música, cada una escrita para cada uno de los instrumentos o voces involucradas.

Silencios o pausas. Los silencios son signos que indican la interrupción del sonido. Hay siete figuras de silencio (Tabla 4) que expresan la duración más o menos larga de la interrupción del sonido.








<i>Pausa</i>	<i>Media pausa</i>	<i>Aspiración</i>	<i>Media aspiración</i>	<i>Cuarto de aspiración</i>	<i>Octavo de aspiración</i>	<i>Dieciseisavo de aspiración</i>
						
4/4	1/2	1/4	1/8	1/16	1/32	1/64

Tabla 4 Figuras y valor relativo de los Silencios²

Alteración. La alteración es un signo que modifica el sonido de la nota a la cual afecta. La alteración se coloca antes de la nota que modifica y en la misma línea o espacio en que la nota está colocada, su efecto continúa sobre todas las notas del mismo nombre que se encuentren en el límite de un compás cualquiera que sea la octava en que estén colocadas. También se colocan al principio del pentagrama e inmediatamente después de la llave, siempre en la misma línea o espacio de la nota que debe modificar, mientras esta alteración exista en la llave su efecto se hace extensivo a todas las notas de su mismo nombre, sea cual fuere la octava en que se encuentren escritas, puede ser alguna de las siguientes (Tabla 5):

- El sostenido, sube el sonido de la nota medio tono.
- El bemol, baja el sonido de la nota medio tono.
- El becuadro, destruye el efecto del sostenido o del bemol.
- El doble sostenido, eleva el sonido dos veces lo que el sostenido simple.
- El doble bemol, disminuye el sonido dos veces lo que el bemol simple.

² La pausa se coloca debajo de la cuarta línea, la media pausa sobre la tercera línea.

<i>Sostenido</i>	<i>bemol</i>	<i>becuadro</i>	<i>doble sostenido</i>	<i>doble bemol</i>
#	b	q	x	bb

Tabla 5 Las alteraciones

2.2 Signos secundarios

Con las figuras anteriores no es posible expresar cualquier valor de duración, para hacerlo posible se han inventado con este objeto otros signos llamados *signos secundarios*.

Punto. El punto se coloca después de una nota para aumentar su valor en una mitad más. Con la ayuda del punto, o puntillo, se pueden obtener duraciones iguales a tres cuartas partes del valor de las diferentes figuras de las notas. Se coloca igualmente después de las figuras de los silencios teniendo el mismo efecto que sobre las notas. No se usa un punto después de la pausa ni de la media pausa, se emplea solamente a partir de la aspiración.

Doble Punto. Se puede también colocar un segundo punto después de una nota o de un silencio, aumentando en una mitad la duración del primer punto; es decir, aumenta todavía una cuarta parte más de su duración primaria. Se pueden obtener entonces duraciones iguales a siete octavas partes del valor de las diferentes figuras de las notas.

Tresillo. Con el fin de no multiplicar los signos de escritura musical, para representar el tresillo se emplean las figuras que se conocen. Un tresillo indica que tres notas deben tocarse en el tiempo que normalmente se tocarían únicamente dos. Se llama doble tresillo o seisillo a la unión de dos tresillos en un sólo grupo. En lugar de poner un tres a cada uno de los tresillos separados se indica con un 6 que se coloca sobre todo el grupo.

Divisiones irregulares. Algunas veces se forman grupos de valores irregulares dividiendo una figura de nota. Estos grupos compuestos de notas en número impar (5, 7, 9, 11, etc.), se representan por la especie de nota par que combine la división más análoga, debiendo llevar encima este grupo una cifra que indique el número de notas que contiene.

Ligadura. La ligadura es una línea curva que une dos notas de un mismo sonido, sea cual fuere su duración. Indica el aumento del valor de la segunda nota al valor o duración de la primera, se dice entonces que estas notas son ligadas.

Tonalidad. La tonalidad es el conjunto de leyes que rigen la constitución de las escalas. Los sostenidos o bemoles que forman parte de una escala no se colocan delante de cada una de las notas que alteran, porque esto recargaría la escritura musical. Se les coloca, por su orden de sucesión, inmediatamente después de la llave, al principio del pentagrama y sobre las mismas líneas y espacios de las notas que alteran.

Los sostenidos o bemoles colocados así forman lo que se llama la *armadura de la llave* y su efecto se hace extensivo a toda la duración de la pieza musical a no ser que la armadura de la llave se modifique al enunciar una nueva tonalidad.

Compás o medida. El compás es la división de una pieza musical en partes iguales, esta división se indica por medio de barras que atraviesan el pentagrama perpendicularmente y se les da el nombre de *barras de compás*. El conjunto de valores, notas o silencios que están comprendidos entre dos barras de compás forman un compás. La suma de estos valores debe ser igual (*cuadratura*) en todos los compases de una misma pieza musical y por consiguiente todos estos compases tendrán una duración igual.

El fin de una pieza de música se indica siempre por medio de una *doble barra* de compás encima de la cuál se escribe generalmente la palabra *Fin* o *Fine*. La

doble barra también se utilizan para separar dos partes de una pieza, hacer un cambio en la armadura de la llave o hacer un cambio en el compás.

Un compás se divide en dos, tres o cuatro partes llamadas *tiempos*. Cuando los tiempos de un compás son divisibles por dos, se les llama *tiempos binarios* y constituyen el compás simple o medida simple. Cuando los tiempos de un compás son divisibles en tres se les llama *tiempos ternarios* y constituyen el compás compuesto o medida compuesta.

Los diferentes compases se indican con dos cifras dispuestas en forma de fracciones que siempre se refieren a la redonda, que es la unidad. Estas dos cifras se colocan al principio de la pieza musical, inmediatamente después de la armadura de la llave. El numerador (indicando número) expresa la cantidad de valores de tiempo que forman un compás, el denominador (indicando la denominación) expresa la duración de esos valores.

La métrica también sirve para saber cuáles y cuántas notas deben agruparse con una barra. Un compás se divide en tantas partes como indique el numerador, sólo se pueden agrupar las notas que estén dentro de una de estas divisiones de compás y que su duración sea menor o igual al valor indicado por el denominador. Esto es para dar una mejor idea del ritmo. La Figura 5 muestra como se agrupan las notas en un compás de 4/4.



Figura 5 Agrupación de notas de acuerdo a la métrica

Movimiento. El movimiento es el grado de lentitud o violencia con que debe ser ejecutada una pieza musical. Ya que los signos expresan duración, sean notas o silencios, tienen entre sí un valor relativo, el movimiento es el que determina la duración absoluta de estos signos diferentes. Existe una gran variedad en los movimientos, desde el más lento hasta el más vivo, se indican con términos italianos que se colocan al principio de la pieza musical sobre el pentagrama. Los términos de la Tabla 6 indican los movimientos principales.

<i>Términos</i>	<i>Abreviaturas</i>	<i>Significado</i>
<i>Largo</i>		Largo, muy lento
<i>Larghetto</i>		Un poco menos lento que el Largo
<i>Lento</i>		Lento
<i>Adagio</i>		Menos lento que el anterior
<i>Andante</i>	<i>And^{te}</i>	Espacio, moderado (andando)
<i>Andantino</i>	<i>And^{ti}</i>	Menos espacio que el andante
<i>Allegretto</i>	<i>All^{ro}</i>	Menos vivo que el allegro
<i>Allegro</i>	<i>All^{ro}</i>	Alegre, aprisa, con viveza
<i>Presto</i>		Aprisa, más que el anterior
<i>Prestissimo</i>	<i>Prest^{is}</i>	Muy vivo, lo más aprisa posible

Tabla 6 Términos que se usan para los movimientos

A estos términos pueden añadirse otros para modificarlos (Tabla 7), en cuanto al carácter o expresión e interpretación de la pieza, y con ayuda de adverbios se pueden obtener nuevas modificaciones (Tabla 8).

<i>Término</i>	<i>Significado</i>	<i>Término</i>	<i>Significado</i>
<i>Affettuoso</i>	Con afecto	<i>Maestoso</i>	Majestuoso
<i>Agitato</i>	Con agitación	<i>Moderato</i>	Moderato
<i>Brioso, con Brio</i>	Energico, con brio	<i>Mosso</i>	Animado, movido
<i>Cantabile</i>	Cantable, cantando	<i>Risoluto</i>	Resuelto
<i>Con anima</i>	Con alma	<i>Scherzando (Scherz.)</i>	Jugueteando
<i>Con espressione</i>	Con expresión	<i>Sostenuto</i>	Sostenido el movimiento
<i>Con Fuoco</i>	Con Fuego	<i>Tempo Giusto</i>	Justo, preciso el tiempo
<i>Con moto</i>	Movido, con vida	<i>Vivace</i>	Vivamente
<i>Con spirito</i>	Con espíritu	<i>Vivo</i>	Vivo con vivacidad
<i>Grazioso</i>	Con gracia		

Tabla 7 Términos auxiliares

<i>Adverbio</i>	<i>Significado</i>
<i>Poco</i>	Poco
<i>Poco a poco</i>	Poco a poco
<i>Piú</i>	Más
<i>Molto piú</i>	Mucho más
<i>Non molto</i>	No mucho
<i>Un poco piú</i>	Un poco más

<i>Adverbio</i>	<i>Significado</i>
<i>Non tanto</i>	No tanto
<i>Non Troppo</i>	No mucho
<i>Assai</i>	Más
<i>Molto</i>	Mucho
<i>Quasi</i>	Casi

Tabla 8 Adverbios para los movimientos

Sin embargo, aún con todas estas indicaciones, no es posible designar el movimiento con exactitud, puesto que la misma indicación se aplica muchas veces a diversos movimientos. Esta dificultad quedó allanada con la invención de un instrumento, llamado *metrónomo*, que indica con precisión las más pequeñas diferencias en el movimiento. La *indicación metronómica* se coloca después del término que indica el movimiento; se expresa por medio de una figura de nota (con o sin punto) seguida de un número del cual se separa por medio de un signo igual (=). El valor de la figura de nota debe durar el tiempo que tenga cada oscilación del metrónomo.

Punto de órgano o punto de espera (calderón). Se puede suspender el movimiento de una manera momentánea. Esta suspensión, cuya duración no está determinada, se indica con el signo '∧'. Colocado encima o debajo de una nota, toma el nombre de punto de órgano; colocado encima o debajo de un silencio toma el nombre de punto de espera.

2.3 Principios generales de la ejecución musical

Una vez conocidos los signos que se emplean en la notación musical, su entonación y duración, se conocen los elementos necesarios para la lectura musical. Sin embargo, la ejecución de una obra musical sería fría e incolora si no estuviese, por decirlo así, animada, vivificada, coloreada de alguna manera por el intérprete que, identificado con el pensamiento del autor, da a esa obra la

expresión que le es propia. La expresión comprende: el fraseo, la acentuación, el matiz y el carácter.

Fraseo. El fraseo es la observancia exacta de la puntuación musical. Toda composición musical, lo mismo que el discurso en el lenguaje, se divide en periodos o frases, y en miembros de periodos o miembros de frases. Un miembro de periodo, está compuesto de una o varias pequeñas ideas melódicas que toman el nombre de diseño o dibujo melódico.

Acentuación. Ciertas notas en una frase musical, así como algunas sílabas de una palabra, deben ser acentuadas con más o menos fuerza, llevando una inflexión particular. Esta acentuación se indica por medio de signos o con términos italianos que se muestran en la Tabla 9.

Ligadura de fraseo. La ligadura de fraseo se coloca sobre varias notas de diferente sonido e indica que deben enlazarse los sonidos uno con otro, sosteniendo el sonido hasta unirlo con el siguiente y así sucesivamente, sin que haya ninguna interrupción o silencio entre uno y otro.

Punto. El punto se coloca encima o debajo de las notas e indica que los sonidos deben producirse picados, es decir separando perfectamente un sonido del otro, cuyo efecto es enteramente contrario al que produce la ligadura. Indica que la nota puede tocarse con una duración un poco menor.

Guión. El guión se coloca encima o debajo de las notas e indica que las notas pueden tocarse con una duración un poco mayor.

Punto alargado ' v '. El punto alargado se coloca también sobre o debajo de las notas (exceptuando las de larga duración), y determina la ejecución destacada o *staccato*, es decir, que los sonidos deben destacarse con vivacidad y a la vez deberán ser articulados con energía.

Términos	Abreviaturas	Significado
<i>Forte-piano</i>	<i>fp</i>	Fuerte la primera nota y débil la siguiente
<i>Piano-forte</i>	<i>pf</i>	Débil la primera nota y fuerte la siguiente
<i>Legato</i>	<i>Leg</i>	Ligado
<i>Legatissimo</i>	<i>Leg^{ssimo}</i>	Lo más ligado posible
<i>Liggiero</i>	<i>Legg</i>	Ligera la articulación (no pesada)
<i>Marcato</i>	<i>Marc</i>	Marcado, marcando
<i>Pesante</i>	<i>Pes</i>	Pesado, pesadamente
<i>Rinforzando</i>	<i>Rinf</i> ó <i>Rfz</i>	Reforzando el sonido
<i>Sforzando</i>	<i>Sfz</i>	Dando mayor fuerza al sonido, de pronto
<i>Sostenuto</i>	<i>Sost</i>	Sostenido el sonido
<i>Staccato</i>	<i>Stacc</i>	Destacado
<i>Tenuto</i>	<i>Ten</i>	Teniendo el sonido

Tabla 9 Términos de Acentuación

Cuando una o varias notas deben acentuarse más fuertemente que otras, se indica esta acentuación con el signo \vee , que se coloca encima o debajo de las notas que deban llevar dicha acentuación.

Matices. Los matices son los diferentes grados de fuerza por los cuales pueden pasar uno o varios sonidos. El signo '<' indica que se debe aumentar gradualmente la fuerza del sonido, '>' indica que se debe disminuir gradualmente la fuerza del sonido, '< >' indica que primero se debe aumentar y después disminuir gradualmente la fuerza del sonido. Se pueden utilizar términos italianos para los matices con cambios graduales (Tabla 10) o para matices constantes (Tabla 11).

Términos	Abreviatura	Significado
<i>Crescendo</i>	<i>Cres</i>	Cresciendo, aumentando
<i>Decrescendo</i>	<i>Decres</i>	Decresciendo, disminuyendo
<i>Diminuendo</i>	<i>Dim</i>	Disminuyendo
<i>Calando</i>	<i>Cal</i>	
<i>Morendo</i>	<i>Mor</i>	Muriendo
<i>Perdendosi</i>	<i>Perd</i>	Disminuyendo el sonido hasta perderse
<i>Smorzando</i>	<i>Smorz</i>	Alejando el sonido hasta extinguirlo

Tabla 10 Términos para matices con cambios graduales

<i>Términos</i>	<i>Abreviaturas</i>	<i>Significado</i>
<i>Pianissimo</i>	<i>pp</i>	Muy débil
<i>Piano</i>	<i>p</i>	Débil
<i>Mezzo piano</i>	<i>mp</i>	Medio débil
<i>Un poco piano</i>	<i>poco p</i>	Un poco débil
<i>Sotto voce</i>	<i>sost v</i>	
<i>Mezza voce</i>	<i>mez v</i>	A media voz
<i>Un poco forte</i>	<i>poco f</i>	Un poco fuerte
<i>Mezzoforte</i>	<i>mf</i>	Medio fuerte
<i>Forte</i>	<i>f</i>	Fuerte
<i>Fortissimo</i>	<i>ff</i>	Muy fuerte

Tabla 11 Términos para matices constantes

Carácter. El carácter constituye el color, el tinte general que se da a una pieza musical en su expresión. El intérprete debe saber expresar los sentimientos más variados: la calma, la tristeza, el dolor, la desesperación, la alegría. El carácter, al igual que la acentuación y los matices, se pueden indicar con términos italianos.

La Figura 6 presenta un fragmento de la Sonata No. 8 Op.13 *Pathétique*, segundo movimiento de Ludwig van Beethoven, en donde se muestra la aplicación de los elementos notacionales que se tratan en este capítulo.

The image displays a musical score for guitar, consisting of four systems of two staves each. The top staff of each system is in the treble clef, and the bottom staff is in the bass clef. The time signature is 4/4, and the key signature has one sharp (F#). The melody in the treble clef features a mix of eighth and sixteenth notes, often beamed together, with some rests. The bass line provides a harmonic accompaniment with chords and moving lines. A triplet of eighth notes is clearly marked in the second system of the second staff.

Figura 6 Fragmento de partitura

Capítulo 3. Armonía

Hay que notar que no se puede considerar como música cualquier sucesión o combinación de sonidos. Como se sabe, la música es una de las bellas artes y su belleza radica en la combinación adecuada de los elementos que la forman. La armonía puede definirse como el arte de formar y enlazar los acordes musicales de una pieza de una manera agradable para el oído, es decir, la armonía proporciona las bases para producir obras musicales que tengan sentido, que posean un grado de belleza aceptable.

3.1 Tonalidad

Antes de hablar de tonalidad, es necesario definir qué es un tono. Esta palabra proviene del griego *tonos* que se traduce como *tensión*. Un tono está compuesto por cinco características: *afinación*, *sonoridad*, *duración*, *color*, y *volumen* [5]. La afinación depende de la frecuencia de las vibraciones presentes en el aire que estimulan los órganos auditivos. Es la característica más importante de un tono, y dicha frecuencia depende de la tensión que se ejerza sobre el cuerpo que produce el sonido. La sonoridad es la voluminosidad de un sonido, éste es, un sonido grave se escucha con más volumen que un sonido agudo. El color o timbre de un sonido es lo que permite diferenciar entre un tono tocado por una trompeta y ese mismo tono tocado en un piano. El volumen es la intensidad con que un sonido llena el espacio.

La música es un concepto que depende de las civilizaciones. La música china difiere de la hindú, y éstas a su vez de la que se conoce comúnmente como música occidental. La diferencia radica tanto en la selección del material tonal, como en la relación que existe entre los tonos que lo forman.

La música que se conoce comúnmente como occidental se forma a partir de una colección de doce tonos, ésto se conoce como *orden cromático*; visto en el teclado de un piano, involucra las teclas blancas y las teclas negras (Figura 7). Existe un orden que considera *siete* de estos doce tonos, y recibe el nombre de *orden diatónico*. Aunque existe un único orden diatónico, es posible obtener diferentes escalas diatónicas; esto es, se pueden obtener siete tonos del orden cromático cambiando el inicio de la distribución (Figura 8). Esto se conoce con el nombre de *modo*. Generalmente se usan dos modos: *mayor* y *menor* (originalmente llamados Jónico y Aeólico), aunque existen otros (dórico, lidio, mixolidio). La distribución interválica del modo mayor y menor es la que se muestra en la Figura 9:

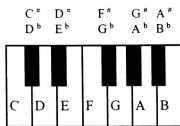


Figura 7 Octava de un piano

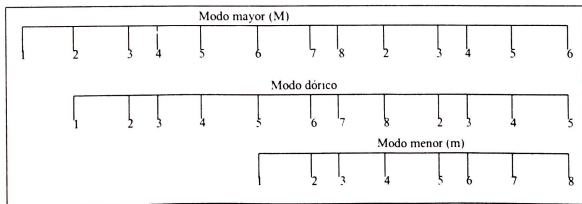


Figura 8 Diferentes modos del orden diatónico

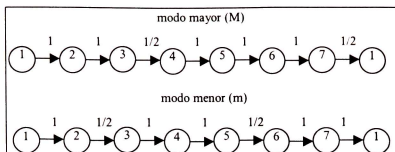


Figura 9 Intervalos en los modos mayor y menor¹

Las escalas de un modo pueden considerarse como *centro* ó *fundamental* (etiquetado como nodo 1) cualquier tono del orden cromático. Esto provoca que la relación que sostienen los tonos entre sí sea dinámica, esto es, lo que importa es el contexto o relación que existe entre un tono cualquiera y el que se elija como fundamental. Ejemplos de escalas en modo mayor o menor se muestran en la Tabla 12.

Nombre de la escala	Tonos que la forman							
Do Mayor (C M)	C	D	E	F	G	A	B	C
Re Mayor (D M)	D	E	F [#]	G	A	B	C [#]	D
La menor (A m)	A	B	C	D	E	F	G	A
Re menor (D m)	D	E	F	G	A	B ^b	C	D

Tabla 12 Ejemplos de escalas

La llave representa la organización de los doce tonos con referencia al tono fundamental y al modo de la escala; es decir, clasifica los doce tonos en dos tipos: los siete tonos que pertenecen a la escala o tonos diatónicos, y los que no pertenecen o tonos cromáticos [5].

La tonalidad representa las relaciones existentes entre los tonos con los centros, de los centros con las llaves y de las llaves con otras llaves.

¹ Los números de los nodos representan los grados de la tonalidad y los números de las aristas representan las distancias en tonos que existe entre ellos

3.3 Acordes y su clasificación

El acorde es el elemento básico de la armonía y consiste en la producción simultánea de varios sonidos diferentes. El estudio de la estructura y de las funciones que desempeña el acorde en la música constituye el objetivo principal de la *teoría armónica* [7].

El acorde representa el aspecto vertical de la música en contraste con el aspecto horizontal representado por la melodía. Es importante notar que el acorde es una forma de coexistencia de los tonos que lo forman, no una suma de tonos, ni una mezcla, ya no se tiene el concepto aislado de *tono*. La característica esencial del acorde es su *unidad*, es decir, las diferencias acústicas de los tonos que lo forman se minimizan para expresar una nueva sensación. La segunda característica es su *organización interna*, ésta es la que presenta la relación dinámica existente entre los distintos tonos que forman el acorde y lo que hace que su sonido sea agradable al escucharse.

Existen varios tipos de acordes, el tipo básico es el que recibe el nombre de *acorde perfecto*, este acorde se forma al tocar de manera simultánea el tono que funciona como *centro dinámico* dentro del acorde, que recibe el nombre de *tono raíz* y es el que da nombre al acorde, después el tono que está en el tercer grado, conocido como *tercera*, y por último el tono que se encuentra en el quinto grado o *quinta*; esto es, el acorde perfecto se forma utilizando 1, 3 y 5. El acorde que se conoce como *de séptima* se forma similarmente al perfecto, agregando el tono que esté en el séptimo grado (1, 3, 5, y 7). Es importante notar que la producción simultánea de dos tonos forma un *intervalo* y no un acorde. Otro tipo de acorde, es el que se conoce como acorde de sexta, que se caracteriza por el uso del sexto grado en lugar del séptimo.

La tercer característica del acorde es su *indiferencia a la distribución* de los tonos que lo conforman, aunque el acorde se considera completamente balanceado solo cuando su tono raíz es el más grave. Cuando el acorde perfecto

aparece en su estructura normal (1, 3, 5), se dice que el acorde está en su *posición raíz*, ahora bien, si la tercera o la quinta están en la parte más baja (3, 5, 1 y 5, 1, 3 respectivamente), se percibe cierta tensión, dichas posiciones se conocen como *primera* y *segunda inversión*. El acorde de cada tono que pertenece a la escala se conoce como *grado armónico*⁴.

Hasta este punto se ha mencionado que los acordes están balanceados, ¿cómo puede una sucesión de estados balanceados o estables expresar movimiento? Esto es posible gracias a que cada acorde posee una *cualidad dinámica* dentro de la tonalidad. La desigualdad de las distancias en la escala da lugar a cambios en la estructura de los acordes de cada grado, produciendo un cambio en sus cualidades. Existen cinco tipos de cualidades básicas:

1. Mayor (M)
2. Dominante (x)
3. Menor (m)
4. Sensible (ϕ)
5. Disminuido (o)

Las cualidades de los grados armónicos dentro de la escala mayor, considerando acordes de séptimas, son las siguientes:

- ◆ I y IV son mayores
- ◆ V es dominante
- ◆ II, III y VI son menores
- ◆ VII es sensible

Estas cualidades se clasifican, de acuerdo a su importancia en el contexto armónico, en dos: primarias (mayor, dominante, menor) y secundarias (sensible y disminuido). Estas cualidades se forman como se muestra en la Figura 11.

⁴ Los grados armónicos se representan con números romanos, los tonos que forman los grados con números arábigos.

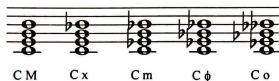


Figura 11 Cualidades

El estudio de las cualidades dinámicas de los grados de la tonalidad permite establecer relaciones entre ellos, llamadas *progresiones* [7]. Uno de los objetivos de este trabajo es generar música que tenga sentido, es decir, que los elementos que la forman estén relacionados. Debido a esto, se analizan algunos tipos de progresiones que se utilizan en la aplicación desarrollada. En este trabajo se estudian tres tipos de progresiones:

a) *Progresión del Círculo de Quintas:*

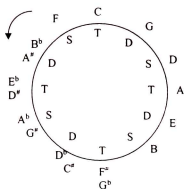
Al considerar la relación que tienen los tonos entre sí y que los acordes están en balance basados en su tono raíz, es posible notar que la progresión del círculo de quintas muestra que la relación existe también en los acordes. Cuando se escucha el sonido de IV seguido del I, el oído percibe que el acorde IV no está en un estado de balance, como si tuviera que *moverse* a otro acorde, aunque no define la dirección hacia la cuál tiene que ser dicho movimiento. Al analizar el círculo de quintas, puede notarse que, en general, los acordes tienden a desplazarse al acorde que está en la posición situada en sentido opuesto al movimiento de las manecillas del reloj. El círculo de quintas transforma a la tonalidad en un *sistema dinámico*. Los acordes que lo componen, distintos entre sí por su estructura, se diferencian ahora también por sus características dinámicas al operar en ellos este principio. El acorde principal de una tonalidad es el I, llamado "*centro tonal*", debido a que representa el centro de estabilidad de

todo el sistema. Su contraparte está representada por el acorde V (conocido como *acorde dominante*), que expresa una necesidad de resolución en el acorde I. Esta relación se conoce como *cadencia dominante* o *auténtica*. Como primera impresión pareciera que los términos no se seleccionaron apegados a la función de los acordes, ya que el acorde I es el que controla o domina la función de los demás acordes, pero el nombre se debe a que el acorde V, a diferencia de los demás grados de la tonalidad está directamente dirigido hacia I, por esto se dice que el acorde V es el que *domina* el acceso a I.

Con base en lo anterior, es posible clasificar los acordes considerando su función, identificando tres tipos [6] [7] [17]:

- *Tónicos* o *tonales* (T): Que expresan balance, como son, el acorde I y su relativo VI.
- *Dominantes* (D): que crean una tensión debido a que contienen al séptimo grado de la escala, que está a medio tono de la tónica, provocando que expresen también la necesidad de resolver en el acorde tonal, por ejemplo V y III contienen en su acorde a 7.
- *Subdominantes* (S): que simplemente expresan tensión.

En armonía, una *serie funcional*, es decir, una serie que respeta el orden y a la vez usa la cualidad dinámica de la escala, es aquella formada por T-S-D-T... [6] [17]. La música puede modelarse utilizando la serie funcional, como una secuencia de grados armónicos con cualidades tonal, dominante y subdominante. Al concatenar varias secuencias de este tipo, se tiene una pieza musical que es armónicamente correcta y resulta musicalmente agradable. Visto esto en el círculo de quintas, se obtiene la Figura 12:



T = Tonal D = Dominante S = Subdominante

Figura 12 Círculo de quintas con cualidades funcionales

b) Progresión Diatónica:

La progresión diatónica es la que se forma al enlazar los acordes cuando estos se mueven siguiendo el orden natural de la escala. Este movimiento puede ser ascendente o descendente y usualmente comprende fragmentos cortos.

Ejemplos: I → II → III ó V → IV → III → II → I

c) Progresión Cromática:

Este tipo de progresión representa el enlace entre acordes relacionados por semitonos, es decir, acordes que se mueven siguiendo el orden de la escala cromática.

Ejemplos III → bIII → II → bII → I ó V → bVI → VI

Existen otros tipos de recursos armónicos que tienen una extensa aplicación en la armonización tonal moderna, estos recursos permiten embellecer una pieza musical realzando las cualidades de los grados armónicos que la forman. Este enriquecimiento de piezas musicales es otro de los objetivos de este trabajo. Estos recursos se analizan a continuación.

➤ *Sustitución del acorde dominante*

Al analizar la estructura de las tres cualidades primarias de la armonía (mayor, dominante y menor), se observa que sus diferencias se establecen por cambios en sus notas tercera y séptima (ver Figura 11). En base a ésto, las notas tercera y séptima se consideran *notas de color*, en tanto que la fundamental y la quinta son *notas de soporte*. En la sustitución del acorde dominante, el que lo sustituye es otro dominante que tiene en común con él las notas tercera y séptima, pero situadas en sentido inverso, como séptima y tercera. La sustitución se basa en estas dos notas porque, como se mencionó, son las que identifican la cualidad. Para encontrar el acorde sustituto de cualquier dominante, se toma la quinta del acorde que se desea sustituir y se baja medio tono, esto se muestra en la Figura 13.

Sol x	Re x
f	b
d	A ^b
b	f
g	d ^b

Figura 13 Sustitución del acorde dominante

Un resultado importante de la sustitución del acorde dominante es la transformación de la progresión del círculo de quintas en una progresión cromática. Por ejemplo, suponiendo que se tienen los siguientes tres acordes relacionados según el círculo de quintas: II → V → I

En la tonalidad de CM, se trata de los acordes Dm Gx y CM. Si se sustituye Gx por bDx el resultado es: II → bIIx → I

La afinidad sonora entre los acordes se expresa en la armonía con los *acordes paralelos* y los *acordes sinónimos*.

➤ *Acordes paralelos*

Un acorde es paralelo de otro cuando tienen tres notas en común. La relación se presenta entre acordes cuyas notas fundamentales están situadas a distancia de terceras. Todos los acordes tienen cuando menos dos acordes paralelos, situados una tercera arriba y una tercera debajo de su nota fundamental. El primero comparte con el acorde original sus tres primeras notas y con el segundo sus tres últimas. Las posibles relaciones paralelas se muestran en la Figura 14.

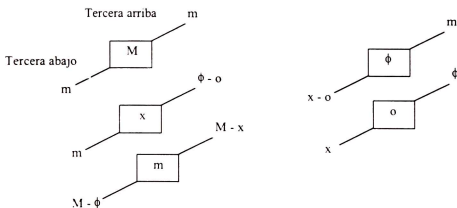


Figura 14 Acordes paralelos

➤ *Acordes sinónimos*

Los acordes sinónimos son los que tienen todas las notas en común. Su característica fundamental es la identidad absoluta de su sonoridad. Existen tres clases de acordes sinónimos:

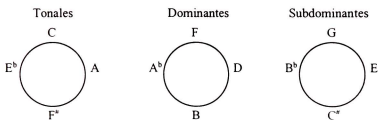
1. Todo acorde mayor con sexta (M+6) es sinónimo de un menor situado a una tercera menor abajo.

Ejemplos: C M+6 = A m, F M+6 = D m, b A M+6 = F m.

2. Todo acorde menor con sexta (m+6) es sinónimo de un sensible situado una tercera menor abajo.

Ejemplos D m+6 = B ϕ , F m+6 = D ϕ , b B m+6 = G ϕ .

3. Todo acorde disminuido es sinónimo de los otros tres acordes disminuidos que tienen su misma cualidad funcional.



Capítulo 4. Lenguajes Formales y Gramáticas

En el capítulo 3 se menciona que el uso de progresiones garantiza que la relación entre los grados armónicos involucrados en éstas sea agradable. La progresión del círculo de quintas y el uso de la serie funcional permite modelar la composición musical como la generación de cadenas formadas por grados tonales, dominantes y subdominantes. Teóricamente una pieza musical podría estar formada por un número infinito de cadenas de la forma T-S-D [6] [7] [17]. Es necesario entonces la especificación finita de lenguajes infinitos. En este capítulo se estudian mecanismos que permiten realizar esta tarea.

4.1 Conceptos básicos

Un lenguaje puede definirse como un conjunto de sentencias, esto es, palabras formadas por símbolos de un vocabulario. La especificación sintáctica de un lenguaje con un número finito de palabras puede hacerse listando todas las palabras que le pertenecen. Esto no es posible cuando se trata de lenguajes con un número infinito de palabras. El problema principal de la teoría de lenguajes formales es desarrollar representaciones finitas de lenguajes infinitos. Esta representación puede realizarse utilizando dispositivos generadores llamados gramáticas.

4.1.1 Alfabetos y lenguajes

Un símbolo es cualquier marca o carácter distinguible, tal como a , b , $+$ o $*$ [15]. Una *cadena* o *palabra* es una secuencia finita de símbolos yuxtapuestos. La longitud de una cadena indica el número de símbolos que forman la cadena y se denota por $|w|$. La *cadena vacía* o *nula* (ϵ), carece de símbolos, por lo tanto $|\epsilon|=0$. La *concatenación* de dos cadenas es una cadena que resulta de escribir la

primera cadena seguida de la segunda, sin espacios intermedios. La cadena vacía es la identidad en la operación de concatenación. Ejemplo: Sea w una palabra cualquiera, $w \varepsilon = \varepsilon w = w$.

Un *alfabeto* o *vocabulario* es un conjunto finito de símbolos. Si V es un alfabeto, entonces V^* denota el conjunto de todas las cadenas formadas con los símbolos de V , incluyendo la cadena vacía y V^+ representa el conjunto $V^* - \{\varepsilon\}$ [3] [4] [15].

Un *lenguaje* es un conjunto cualquiera de cadenas sobre algún alfabeto. Por ejemplo, los siguientes conjuntos son lenguajes sobre el alfabeto $V = \{a, b\}$:

$$L_1 = \{\varepsilon\}, \quad L_2 = \{a, ba, aaba, bbbbbb\}, \quad L_3 = \{a^p \mid p \text{ es primo}\}$$

4.1.2 Grafos y árboles

Un *grafo dirigido*, denotado por $G = (V, E)$, consta de un conjunto finito de *vértices* o *nodos* V y un conjunto de vértices E llamados *aristas* o *arcos* que conectan a dos nodos en V . Una arista que va de los nodos v a w se denota por $v \rightarrow w$. Un ejemplo se muestra en la Figura 15.

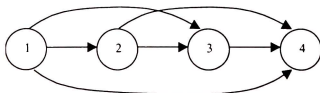


Figura 15 Grafo dirigido $\{1,2,3,4\}, \{i \rightarrow j \mid i < j\}$

Una *ruta* en un grafo dirigido es una secuencia de vértices $v_1, v_2, \dots, v_k, k \geq 1$, tal que $v_i \rightarrow v_{i+1}$ es una arista para cada $i, 1 \leq i < k$. Si $v \rightarrow w$ es una arista, se dice que v es un *predecesor* o *padre* de w y que w es un *sucesor* o *hijo* de v .

Un *árbol dirigido ordenado* es un grafo dirigido que cumple las siguientes propiedades:

- Existe un único vértice, conocido como *raíz*, que no tiene predecesores y del cuál es posible formar una ruta a cualquier vértice.
- Cada vértice, excepto el raíz, tiene exactamente un predecesor.
- Los sucesores de cada vértice están ordenados a partir de la izquierda.

Los vértices que no tienen sucesores reciben el nombre de *hoja*, y los vértices que no son ni hojas ni el nodo raíz, se les conoce como *vértices intermedios*.

Generalmente los árboles se dibujan colocando el nodo raíz en la parte superior, con todas las aristas apuntando hacia abajo, por esto, se omite su dirección.

4.2 Gramáticas

La *cardinalidad* (número de elementos) de un lenguaje puede ser infinita. Esto llevó a la creación de mecanismos que permiten la especificación finita de los lenguajes: gramáticas y autómatas. En esta sección se estudian las gramáticas y en el capítulo 5 los autómatas.

4.2.1 Gramáticas generativas

Una gramática generativa G se representa por la cuadrupla (V_N, V_T, P, S) donde: V_N representa el conjunto de símbolos variables o no-terminales, V_T el conjunto de símbolos terminales, P el conjunto de producciones y $S \in V$ se llama *símbolo inicial*. V_N , V_T y P son conjuntos finitos. Se asume que V_N y V_T son

conjuntos *disjuntos*, es decir, cumplen con la siguiente propiedad: $V_N \cap V_T = \emptyset$ ⁵
Por convención se considera que $V_N \cup V_T \cup \{\epsilon\} = V$. El conjunto de producciones de P son de la forma $A \rightarrow \alpha$, donde A es una variable y α es una cadena de símbolos de V^* .

Se usarán letras mayúsculas para indicar variables y las primeras letras minúsculas del alfabeto para representar terminales. Las cadenas de terminales se indican con letras minúsculas cercanas al final del alfabeto, y cadenas de variables y terminales se denotan con minúsculas del alfabeto griego.

Para definir el lenguaje que genera la gramática $G = (V_N, V_T, P, S)$, es necesario mencionar las relaciones \Rightarrow y \Rightarrow^* entre cadenas en V^* .

Si $\alpha \rightarrow \beta$ es una producción de P , γ y δ son cadenas en V^* , entonces $\gamma \alpha \delta \Rightarrow \gamma \beta \delta$ indica que $\alpha \rightarrow \beta$ se aplica a la cadena $\gamma \alpha \delta$ para obtener $\gamma \beta \delta$; es decir, \Rightarrow relaciona dos cadenas cuando la segunda se obtiene (deriva directamente) de la primera por medio de la aplicación de una sola producción.

Partiendo del supuesto de que $\alpha_1, \alpha_2, \dots, \alpha_m$ son cadenas en V^* , y $\alpha_1 \Rightarrow \alpha_2, \alpha_2 \Rightarrow \alpha_3, \dots, \alpha_{m-1} \Rightarrow \alpha_m$, se dice que $\alpha_1 \Rightarrow^* \alpha_m$, esto es, que dadas dos cadenas α y β , y que $\alpha \Rightarrow^* \beta$, es posible obtener β a partir de α mediante la aplicación de un determinado número de producciones de P .

El lenguaje generado por G se denota $L(G)$, es $\{w \mid w \text{ está en } V_T^* \text{ y } S \Rightarrow^* w\}$. Esto indica que una cadena está en $L(G)$ si:

- La cadena consta solo de terminales.
- La cadena se deriva a partir de S .

○

⁵ \emptyset representa el conjunto vacío.

4.2.2 Tipos de gramáticas

Las gramáticas que se definieron son *gramáticas de tipo 0*. Al aplicar restricciones en la naturaleza de las producciones se obtienen otros tres tipos de gramáticas que se llaman tipo 1, 2 y 3.

Si en una gramática $G = (V_N, V_T, P, S)$ en todas las producciones $\alpha \rightarrow \beta$ en P cumplen $|\beta| \geq |\alpha|$, la gramática G es de *tipo 1* o *sensible al contexto*. Algunos autores señalan que las producciones de las gramáticas sensibles al contexto son de la forma $\alpha_1 A \alpha_2 \rightarrow \alpha_1 \beta \alpha_2$, en donde α_1, α_2 y $\beta \in V^*$, $\beta \neq \epsilon$ y $A \in V_N$. Es posible mostrar que esta restricción no genera otra clase de lenguaje [4], y facilita ver que el reemplazo de A por β es posible cuando A aparece en el contexto de $\alpha_1 \alpha_2$. A continuación se muestra un ejemplo de este tipo de gramáticas.

Sea $G = (V_N, V_T, P, S)$, $V_N = \{S, B, C\}$, $V_T = \{a, b, c\}$, P consta de las siguientes producciones:

1. $S \rightarrow aSBC$
2. $S \rightarrow aBC$
3. $CB \rightarrow BC$
4. $aB \rightarrow ab$
5. $bB \rightarrow bb$
6. $bC \rightarrow bc$
7. $cC \rightarrow cc$

El lenguaje $L(G)$ contiene las cadenas $a^n b^n c^n$ para $n \geq 1$. En este ejemplo cada una de las producciones tiene por lo menos el mismo número de elementos del lado derecho que del izquierdo.

Las gramáticas de *tipo 2* o *libres de contexto* deben satisfacer lo siguiente. Sea $G = (V_N, V_T, P, S)$, en donde cada producción $\alpha \rightarrow \beta$ en P :

1. α consta de solo una variable.
2. β es una cadena diferente de ϵ .

Es importante notar que el lado izquierdo de la producción consta de un solo elemento, por lo que no se requiere un contexto para realizar el reemplazo, de ahí el nombre de libre de contexto. A continuación se muestra un ejemplo de este tipo de gramática.

Sea $G = (V_N, V_T, P, S)$, $V_N = \{S, A, B\}$, $V_T = \{a, b\}$, P consta de las siguientes producciones:

- | | |
|-----------------------|------------------------|
| 1. $S \rightarrow aB$ | 5. $A \rightarrow bAA$ |
| 2. $S \rightarrow bA$ | 6. $B \rightarrow b$ |
| 3. $A \rightarrow a$ | 7. $B \rightarrow bS$ |
| 4. $A \rightarrow aS$ | 8. $B \rightarrow aBB$ |

El lenguaje $L(G)$ consta de palabras con el mismo número de a's que de b's.

Si una gramática $G = (V_N, V_T, P, S)$, en donde cada producción en P es de la forma $A \rightarrow aB$ o $A \rightarrow a$ donde A y B son variable y a es un terminal, se trata de una *gramática regular* o de *tipo 3*. El siguiente ejemplo muestra una gramática de este tipo.

Sea $G = (\{S, A, B\}, \{0, 1\}, P, S)$, donde P consta de las siguientes producciones:

- | | |
|-----------------------|-----------------------|
| 1. $S \rightarrow 0A$ | 6. $B \rightarrow 1B$ |
| 2. $S \rightarrow 1B$ | 7. $B \rightarrow 1$ |
| 3. $A \rightarrow 0A$ | 8. $B \rightarrow 0$ |
| 4. $A \rightarrow 0S$ | 9. $S \rightarrow 0$ |
| 5. $A \rightarrow 1B$ | |

4.2.3 Árboles de derivación

Para representar gráficamente las derivaciones o palabras que pertenecen a una gramática libre de contexto se utilizan los *árboles de derivación*. Los vértices

de un árbol de derivación se etiquetan con símbolos variables o terminales de la gramática. Si un vértice intermedio n tiene la etiqueta A y los hijos de n se etiquetan como X_1, X_2, \dots, X_k comenzando por la izquierda, entonces $A \rightarrow X_1 X_2 \dots X_k$ debe ser una producción.

Formalmente, sea $G = (V_N, V_T, P, S)$ una gramática libre de contexto, un árbol es un árbol de derivación para G si:

- 1) Cada vértice tiene una etiqueta, que es un símbolo de V .
- 2) La etiqueta del vértice *raíz* es S .
- 3) Si un vértice es intermedio y tiene la etiqueta A , entonces $A \in V_N$.
- 4) Si n tiene la etiqueta A y los vértices n_1, n_2, \dots, n_k son hijos del vértice n , ordenados a partir de la izquierda, con etiquetas X_1, X_2, \dots, X_k respectivamente, entonces $A \rightarrow X_1 X_2 \dots X_k$ debe ser una producción en P .

Si se considera la gramática $G = (\{S,A\}, \{a,b\}, P, S)$ donde P consta de las producciones:

- | | |
|------------------------|-----------------------|
| 1. $S \rightarrow aAS$ | 4. $S \rightarrow a$ |
| 2. $A \rightarrow SbA$ | 5. $A \rightarrow ba$ |
| 3. $A \rightarrow SS$ | |

El árbol de derivación relacionado con esta gramática se muestra en la Figura 16.

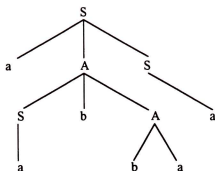


Figura 16 Árbol de derivación

4.2.4 Gramáticas generativas estocásticas.

Existen problemas en los que no sólo es importante la estructura del lenguaje, sino que también es de interés el uso que se le dé al lenguaje. Tal es el caso de la generación automática de música. Para realizar esto, debe existir una forma de ponderar el uso de las producciones a utilizar al momento de generar cadenas. Hasta este punto se han tratado las gramáticas generativas determinísticas, por lo que se tratarán a continuación las gramáticas estocásticas.

Definición: Una gramática estocástica es una cuadrupla $G_s = (V_N, V_T, P_s, S)$ donde V_N y V_T son conjuntos finitos de no-terminales y terminales; $S \in V_N$, y es el símbolo inicial; P_s es un conjunto finito de producciones estocásticas [16].

Definición: Una gramática estocástica G_s es un par ordenado $\langle G, P \rangle$ donde G es una gramática generativa y P es un conjunto ordenado de probabilidades $\{p_{ij}\}$. El conjunto P debe satisfacer lo siguiente [15]:

1. Para cada producción $P_{ij} \in P$, existe una probabilidad única $p_{ij} \in P$.
2. Para cada p_{ij} , $0 < p_{ij} \leq 1$. Si p_{ij} es cero, la producción P_{ij} puede eliminarse de G .
3. Para cada i entre 1 y $|V_N|$,

$$\sum_{1 \leq j \leq m} p_{ij} = 1$$

donde n_i es el número de producciones con el i -ésimo no-terminal A_i en el lado izquierdo de la producción.

Ejemplo: Considere la gramática $G_S = (\{E, T\}, \{a, b, c\}, P_S, E)$ donde P_S consta de las producciones:

$$P_{1,1} \quad E \rightarrow aEET \quad p_{1,1} = 3/4$$

$$P_{1,2} \quad E \rightarrow T \quad p_{1,2} = 1/4$$

$$P_{2,1} \quad E \rightarrow bE \quad p_{2,1} = 5/6$$

$$P_{2,2} \quad E \rightarrow c \quad p_{2,2} = 1/6$$

La probabilidad de una cadena generada por una gramática estocástica es el producto de las probabilidades de las producciones utilizadas en la derivación de la cadena.

Definición: Sea Δ una derivación en una gramática estocástica G_S , la probabilidad de la derivación $p(\Delta)$ está dada por:

$$p(\Delta) = \prod_{1 \leq k \leq |\Delta|} p_k$$

donde p_k es la probabilidad de la producción P_{ik} , la k -ésima producción en Δ [15].

El lenguaje estocástico $L(G_S)$ se describe por (L, p) , donde L es un lenguaje y p es la distribución de probabilidad definida sobre L . El lenguaje $L(G_S)$ se conoce como lenguaje característico de $L(G_S)$ [15].

Una gramática estocástica es *consistente* si y solo si:

$$\sum_{w \in L(G_S)} p(w) = 1$$

Si la gramática es inconsistente, alguna de las probabilidades de las producciones no se está distribuyendo a las cadenas terminales.

4.2.5 Sistemas de Lindenmayer

Los sistemas de Lindenmayer (*L-systems*) son gramáticas que poseen la característica de que no particionan el alfabeto. Cada letra se asocia con una regla de reescritura. A diferencia de las gramáticas citadas en los puntos anteriores, en los *L-systems* el proceso inicia al tomar la *cadena* inicial en lugar de la *letra* inicial. El proceso de reescritura se ejecuta simultáneamente en todas las letras de la palabra que se considera. En los *L-systems* no se especifican las letras terminales, todas las palabras pertenecen al lenguaje del sistema, el hecho de considerar una letra no-terminal indicaría una *célula* muerta [2].

Definición: Un OL-system es una tripleta $OLS=(V, P_0, F)$ donde V es el alfabeto, P_0 es una palabra no vacía sobre V (llamada el axioma o palabra inicial), y F es un conjunto finito de pares ordenados (a, P) donde $a \in V$ y $P \in W(V)$, donde $W(V)$ indica palabras formadas por símbolos de V . Un OL-system es determinístico (DOL-system) si y solo si para cada $a \in V$ existe exactamente un $P \in V^*$ [2] [13].

Ejemplo: Para DOL-system = $(\{a,b\}, a, \{a \rightarrow b, b \rightarrow ab\})$ las palabras del lenguaje son:

a, b, ab, bab, abbab, bababbab,...

donde la longitud de las palabras corresponde a los números de Fibonacci.

4.3 Gramáticas y la generación de música

Desde el punto de vista de los lenguajes formales, puede decirse que se tienen que generar cadenas formadas por símbolos $T^n S^o D^p$ para $n, o, p \geq 1$. Al unir varias de estas cadenas se tiene una pieza musical.

La composición musical (para efectos de generarse automáticamente), puede describirse con la siguiente gramática: Sea $G = (V_N, V_T, P, O)$, donde $V_N = \{O, T, S, D, H, I, J\}$, $V_T = \{\text{do, re, mi, fa, sol, la, si}\}$, y P consta de las siguientes producciones:

$O \rightarrow TSDO$

$O \rightarrow TSdT$

$T \rightarrow HT$

$T \rightarrow H$

$S \rightarrow LS$

$S \rightarrow L$

$D \rightarrow JD$

$D \rightarrow J$

$H \rightarrow \text{do}$

$H \rightarrow \text{la}$

$L \rightarrow \text{fa}$

$L \rightarrow \text{re}$

$L \rightarrow \text{si}$

$J \rightarrow \text{sol}$

$J \rightarrow \text{mi}$

Se puede ver que el uso del enfoque de las gramáticas estocásticas permitiría la ventaja de especificar cuáles producciones se usarán con mayor frecuencia.

Por ejemplo, las producciones que tienen en el lado izquierdo un no-terminal O son las que indican cuando se va a estar iterando en la producción de series funcionales. Si se le asigna a la primer producción una probabilidad alta se generarían cadenas grandes, ya que es la segunda producción la que termina con la generación de la pieza musical. De manera similar las primeras producciones que tienen del lado izquierdo T, S ó D , indican si se desea generar más de un grado tonal, dominante o subdominante, respectivamente, en forma consecutiva. Las

producciones que contienen H, L y J sirven para ponderar el uso de cada uno de los grados armónicos dentro de la pieza musical.

4.4 Comentarios finales

Los lenguajes formales y los distintos tipos de gramáticas generativas son mecanismos que permiten la especificación finita de lenguajes infinitos, como el de la música, en donde una pieza no está acotada a una longitud o duración específica.

En algunas aplicaciones prácticas, es necesario el uso de un cierto grado de incertidumbre. Por ejemplo, al modelar el ruido o distorsión en las comunicaciones, es posible que haya pérdida de datos en la transmisión. Otro caso es el de reconocimiento de patrones, en donde un patrón puede generarse por dos o más gramáticas de patrones [16].

Al utilizar un lenguaje especificado con gramáticas estocásticas en un sistema, se permite que el usuario interactúe con el sistema indicando las producciones que éste prefiere que se utilicen.

A futuro, puede experimentarse el uso de gramáticas de Lindenmayer, ya que en la música, cualquier sucesión de sonidos puede considerarse una pieza musical, formada por terminales.

Capítulo 5. Conceptos de Autómatas

Con la finalidad de estudiar otros mecanismos de especificación finita de lenguajes infinitos, este capítulo trata conceptos básicos de autómatas, así como el uso del diagrama de transición asociado al autómata, teniendo como ventaja una representación gráfica que visualice las posibles transiciones o futuros estados del sistema dependiendo de una entrada específica.

Debido a que esta tesis tiene como uno de sus objetivos la creación de una aplicación que use una gramática para generar piezas musicales, el uso de autómatas facilita al usuario entender el sistema.

5.1 Autómatas finitos determinísticos y no-determinísticos

5.1.1 Autómatas finitos determinísticos

Los autómatas finitos son modelos matemáticos que representan sistemas con entradas y salidas discretas. El sistema puede estar en cualquiera de las configuraciones internas llamadas *estados*.

Los autómatas finitos son otro método de especificación finita de lenguajes infinitos. Un autómata finito no puede definir todos los lenguajes que se representan con las gramáticas, únicamente los lenguajes de tipo 3.

Un autómata finito tiene un conjunto finito de estados y un conjunto de transiciones de un estado a otro que ocurren cuando se toma un símbolo del alfabeto de entrada Σ . Un estado, generalmente representado como q_0 , es el estado

inicial, que es donde el autómata inicia. Algunos estados se designan como finales o estados de aceptación.

Formalmente, un *autómata finito* M sobre el alfabeto Σ es un sistema $(K, \Sigma, \delta, q_0, F)$, donde K es un conjunto de estados finito no vacío, Σ es un alfabeto finito de entrada, δ es un mapeo de $K \times \Sigma$ en K , $q_0 \in K$ es el estado inicial y $F \subseteq K$ es el conjunto de estados finales [4].

La interpretación de $\delta(q, a) = p$ para q y $p \in K$ y $a \in \Sigma$, es que M , en el estado q al tomar el símbolo de entrada a , mueve su cabeza de lectura y pasa al estado p .

Es conveniente extender δ al dominio $K \times \Sigma^*$, definiendo δ' de la siguiente forma:

$$\delta'(q, \epsilon) = q.$$

$$\delta'(q, xa) = \delta(\delta'(q, x), a) \text{ para cada } x \in \Sigma^* \text{ y } a \in \Sigma.$$

La interpretación de $\delta'(q, x) = p$ es que M , estando en q con la cadena x como entrada estará en el estado p cuando termine de leer la cadena x . Por conveniencia en el resto del texto se usará δ para indicar tanto δ como δ' .

M acepta una cadena x si $\delta(q_0, x) = p$ para algún $p \in F$. El conjunto de todas las cadenas x que acepta M se denotan como $T(M)$; formalmente:

$$T(M) = \{x \mid \delta(q_0, x) \in F\}$$

Un grafo dirigido, o *diagrama de transición*, se asocia con un autómata finito de la siguiente manera. Los vértices del grafo corresponden a los estados del autómata. Si existe una transición del estado q al estado p con la entrada a , entonces se coloca una arista en el diagrama de transición del estado q al p con la etiqueta a .

La Figura 17 muestra la especificación de un autómata finito y su correspondiente diagrama de transición. El estado inicial, q_0 , se indica con una

flecha etiquetada como "inicio". Existe un estado final que se indica con un doble círculo. El autómata acepta todas las cadenas que tienen un número par de 0's y de 1's. El control inicia en q_0 y debe terminar en q_0 para que la cadena se acepte.

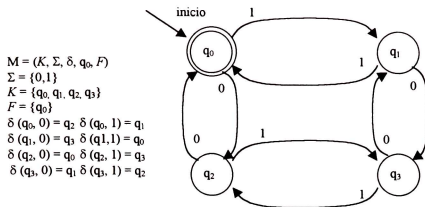


Figura 17 Daigramas de transición

5.1.2 Autómatas finitos no-determinísticos

En la teoría de lenguajes existe el concepto de no-determinismo, por lo que se creó un modelo que representa este tipo de lenguajes.

Un autómata finito no-determinístico (AFN) M , es un sistema $(K, \Sigma, \delta, q_0, F)$, donde K es un conjunto de estados finito no vacío, Σ es un alfabeto finito de entrada, δ es un mapeo de $K \times \Sigma$ en subconjuntos de K , $q_0 \in K$ es el estado inicial y $F \subseteq K$ es el conjunto de estados finales [4].

La diferencia entre autómatas determinísticos y no-determinísticos radica en que $\delta(q, a)$ es un conjunto (que puede ser vacío) de estados, en vez de que sea un único estado. La interpretación de $\delta(q, a) = \{p_1, p_2, \dots, p_k\}$ es que M , en el estado q , al tomar a como entrada puede tomar como siguiente estado p_1 ó p_2 ó... ó p_k .

La Figura 18 muestra un ejemplo de un AFN. Es importante notar que el estado q_0 puede pasar al estado q_3 o quedarse en q_0 al tomar como entrada un 0. Si M está en q_3 y toma como entrada un 1, no se tiene definida una transición.

$$M = (\{q_0, q_1, q_2, q_3, q_4\}, \{0,1\}, \delta, q_0, \{q_2, q_4\})$$

$$\delta(q_0, 0) = \{q_0, q_3\}$$

$$\delta(q_1, 0) = \phi$$

$$\delta(q_2, 0) = \{q_2\}$$

$$\delta(q_3, 0) = \{q_4\}$$

$$\delta(q_4, 0) = \{q_4\}$$

$$\delta(q_0, 1) = \{q_0, q_1\}$$

$$\delta(q_1, 1) = \{q_2\}$$

$$\delta(q_2, 1) = \{q_2\}$$

$$\delta(q_3, 1) = \phi$$

$$\delta(q_4, 1) = \{q_4\}$$

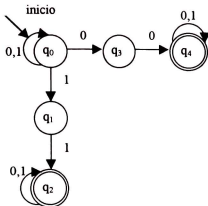


Figura 18 Automata finito no-determinístico

5.1.3 Equivalencia entre un AFD y un AFN

Los autómatas finitos determinísticos (AFD) son un caso especial de los autómatas finitos no determinísticos (AFN), en donde cada estado tiene una sola transición para cada símbolo. Por lo tanto, todos los lenguajes que reconoce un AFN los aceptan un AFD equivalente. Para realizar esto, se hace corresponder a los estados de un AFD con un conjunto de estados de un AFN.

Sea $M = (K, \Sigma, \delta, q_0, F)$ un AFN que acepta un lenguaje L . Se define un AFD $M' = (K', \Sigma, \delta', q'_0, F')$ donde M' representa todos los subconjuntos del conjunto de estados M . $K' = 2^K$. M' verifica el estado de todos los estados de M en cualquier instante. F' es el conjunto de todos los estados de $K' \in F$. Cada elemento de K' se denota como $[q_1, q_2, \dots, q_i]$ donde $q_1, q_2, \dots, q_i \in K$. Inicialmente se considera $q'_0 = [q_0]$.

Se define $\delta'([q_1, q_2, \dots, q_i], a) = [p_1, p_2, \dots, p_i]$ si y solo si $\delta(\{q_1, q_2, \dots, q_i\}, a) = \{p_1, p_2, \dots, p_i\}$

Al aplicar δ a cada elemento de $\{q_1, q_2, \dots, q_i\}$ mediante la unión, se obtiene un nuevo conjunto de estados $\{p_1, p_2, \dots, p_j\}$ ⁶. Este nuevo conjunto de estados se representa con un elemento en K' , y tiene el valor de $\delta'(\{q_1, q_2, \dots, q_i\}, a)$.

5.2 Autómatas probabilísticos

Los autómatas probabilísticos se comenzaron a desarrollar a principios de los sesentas, al presentárseles como una generalización natural de los diferentes tipos de autómatas determinísticos.

Los autómatas probabilísticos, tienen medidas de probabilidad en las transiciones entre los estados, por lo que pueden generar o aceptar lenguajes estocásticos. En general, un *lenguaje estocástico* $L(G_s)$ se caracteriza por $\langle L, p \rangle$ donde L es el lenguaje y p es la distribución de probabilidad definida sobre L .

Un autómata probabilístico es aquél en el que $\delta(q, a) = r p$, donde r es un factor de probabilidad para que M pase del estado q al estado p tomando como entrada a [8].

El conjunto de valores que puede tomar una variable X en un determinado tiempo t , denotado como $X(t)$, se conoce como *espacio muestral*, denotado por S . El conjunto de valores que puede tomar el índice t se llama *espacio paramétrico* y se denota por T . Cuando S y T son discretas, tales sistemas se conocen con el nombre de *cadena de Markov* [1].

5.2.1 Cadenas de Markov

Las cadenas de Markov se representan siguiendo el siguiente modelo: El espacio muestral se representa por un número finito de vértices, la variable aleatoria

⁶ Hopcroft y Ullman dan una demostración completa [3, pág. 22] [4, pág. 31].

x_t representa la posición de un punto en movimiento en el tiempo $t = i$; $p(x_t = j)$ es la probabilidad de que el punto pase al vértice v_j en el tiempo $t = i$ y $p(x_m = j | x_{m-1} = i)$ es la probabilidad de que el punto esté en el vértice j en el tiempo $t = m$ dado que estuvo en el vértice i en el tiempo $t = m - 1$.

Si ${}_m p_{ij} = {}_n p_{ij}$ (pasa del vértice i al vértice j en el tiempo m y n respectivamente), para cualquier número natural m y n , entonces se dice que la cadena de Markov es *homogénea*, y se llama *no-homogénea* en cualquier otro caso.

Las probabilidades ${}_m p_{ij}$ pueden representarse en una *matriz de transición* que tiene las siguientes propiedades:

- Todos los elementos p_{ij} son probabilidades, por lo tanto $0 \leq p_{ij} \leq 1 \forall i, j \in S$.
- La suma por renglones siempre es 1 (la suma de las probabilidades que salen de un nodo es 1). [1]

En la Figura 19 se muestra un autómata probabilístico de dos estados y la matriz de transición en la Tabla 13.

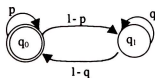


Figura 19 Autómata probabilístico de dos estados

	q_0	q_1
q_0	p	$1 - p$
q_1	$1 - q$	q

Tabla 13 Matriz de transición de un autómata de dos estados

5.3 Autómatas para la generación de música

Para generar piezas musicales que tengan variedad en los grados armónicos que las forman, puede usarse un autómata probabilístico donde las probabilidades indiquen las preferencias en los tonos que han de emplearse.

Si al igual que en la sección 4.3 se considera que se desea generar cadenas formadas por símbolos $T^n-S^o-D^p$ para $n,o,p \geq 1$, aplicando los conceptos de autómatas descritos en este capítulo se obtiene el autómata de la Figura 20:

Sea $M = (K, \Sigma, \delta, q_0, F)$ donde:

$$K = \{q_0, q_1, q_2, q_3, q_4\}$$

$$\Sigma = \{T, S, D\}$$

$$F = \{q_4\}$$

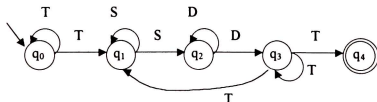


Figura 20 AFN para la generación de música

Con el uso de autómatas se tiene la ventaja de establecer una diferencia en las ponderaciones asociadas a la producción de un tipo de grado tonal. Por ejemplo, en los estados q_0 y q_3 se representa la repetición de más de un grado armónico tonal. Si se empleara una gramática como la empleada en la sección 4.3 no se podría establecer una diferencia que indicara cuando se desean repeticiones de grados armónicos tonales.

En el capítulo 7 se trabajará más con este autómata para generar piezas musicales (incluyendo el manejo de probabilidades) transformándolo en un autómata probabilístico.

5.4 Comentarios finales

La representación de un lenguaje formal utilizando un autómata finito facilita la comprensión del comportamiento de un sistema, ya que el autómata representa la información de cualquier estado del sistema independientemente del tiempo, indicando además los posibles estados siguientes.

Los autómatas finitos pueden emplearse en diferentes aplicaciones, como el diseño de circuitos, donde el circuito se compone de un número finito de compuertas y cada compuerta tiene alguno de dos estados, 0 ó 1, y dependiendo de estos valores de entrada, genera una salida [3].

En la generación automática de música, el manejo de autómatas probabilísticos representa una ventaja, ya que es posible indicar la ponderación de las producciones dependiendo del estado del sistema, lo cual no es posible con el uso de gramáticas.

En resumen puede decirse que los autómatas finitos pueden aplicarse en cualquier sistema que tenga un número finito de estados y un número finito de entradas.

Capítulo 6. Unificación

Existen recursos que permiten embellecer armónicamente una pieza musical resaltando las cualidades de los elementos que la forman. El embellecimiento de una pieza musical, visto como un proceso en la generación automática de música, puede modelarse como un *sistema de reescritura*, con reglas que se aplican en la pieza musical. La representación y formalización del tratamiento de reglas tiene su origen en la lógica, en sistemas de representación que permiten expresar la implicación. En este capítulo se estudian algunos sistemas de representación.

6.1 Lógica

La lógica juega un papel importante dentro de la Inteligencia Artificial y de manera más general en el campo de las aplicaciones computacionales. El punto de principal importancia es que la lógica se puede usar para representar conocimiento concerniente a diferentes contextos.

Un sistema lógico se define desde dos puntos de vista diferentes que son equivalentes:

La *Teoría de los Modelos* examina la relación entre sentencias de lógica al dárseles una interpretación mediante la asignación de valores de verdad, es decir, se basa en la semántica de las sentencias. El vocabulario básico en la teoría de los modelos comprende términos como: verdadero, falso, interpretación, modelo y consecuencia semántica.

La *Teoría de la Demostración* examina la relación entre sentencias en términos de su derivabilidad, es decir, aplicar reglas sobre un conjunto de sentencias con la finalidad de obtener nuevas sentencias. Esta teoría no hace referencia a ninguna interpretación, se basa únicamente en la sintaxis de las sentencias. El vocabulario básico de esta teoría usa palabras como: axioma, regla de inferencia, teorema y consecuencia sintáctica.

Los dos puntos de vista se fundamentan en la definición de un lenguaje. Un lenguaje es una colección de símbolos y de reglas para la construcción de fórmulas bien formadas (*fbf*). Este lenguaje se conoce como lenguaje de la lógica de primer orden.

6.1.1 Lógica de primer orden

El lenguaje de la lógica de primer orden consta de los siguientes símbolos primitivos:

- Delimitadores: , ()
- Símbolos de constantes: a, b, c, \dots a_1, a_2, \dots, a_n
- Símbolos de variables: x, y, z, \dots x_1, x_2, \dots, x_n
- Símbolos de función: f, g, h, \dots f_1, f_2, \dots, f_n
- Símbolos de predicado: p, q, r, \dots p_1, p_2, \dots, p_n
- Conectivos lógicos: $\sim, \wedge, \vee, \rightarrow, \leftrightarrow$.
- Cuantificadores: \forall, \exists .

El lenguaje de la lógica de primer orden se define en tres etapas:

1. Definición del concepto de término.
2. Definición del concepto de predicado o fórmula atómica.
3. Definición del concepto de fórmula bien formada.

Definición: Un término se define recursivamente como sigue:

- (i) Una constante es un término.
- (ii) Una variable es un término.
- (iii) Si f es una función de n argumentos y t_1, \dots, t_n son términos, entonces $f(t_1, \dots, t_n)$ es un término.
- (iv) Cualquier cosa que se genere mediante la aplicación de alguna de las reglas anteriores se denomina término [20].

Cualquier función o símbolo de predicado toma un número específico de argumentos, considerando que si tiene n argumentos es una función n -aria o de aridad n .

Definición: Si P es un símbolo de predicado de grado n , y t_1, \dots, t_n son términos, entonces $P(t_1, \dots, t_n)$ es un átomo o fórmula atómica.

Definición: Una fórmula bien formada (fbf o wff), puede ser una fórmula atómica o tomar alguna de las siguientes siete formas, donde W , W_1 , W_2 son cualquier fórmula bien formada y $Q(x)$ es cualquier cuantificador:

- | | |
|----------------------------------|---|
| 1. W | 5. $W_1 \rightarrow W_2$ (implicación) |
| 2. $\neg W$ (negación) | 6. $W_1 \leftrightarrow W_2$ (sí y solo si) |
| 3. $W_1 \wedge W_2$ (conjunción) | 7. $(Qx) W$ |
| 4. $W_1 \vee W_2$ (disyunción) | |

En la forma de implicación, W_1 se conoce como *antecedente* de la regla y W_2 como el *consecuente*.

Definición: La *ocurrencia* de una variable x en una fórmula es *ligada* si y sólo si la ocurrencia está en el alcance de un cuantificador empleado con la variable, o

bien, es la ocurrencia en ese cuantificador. En caso contrario se dice que el símbolo de variable x es *libre*.

Definición: Una *literal* puede ser una fórmula atómica o bien la negación de una fórmula atómica. Estos tipos de literales se conocen como literal positiva y literal negativa respectivamente.

La *forma clausular* es una de los diferentes tipos de formas normales de la lógica de primer orden. Las diversas formas normales tienen en común la característica de estructurar de una manera uniforme las expresiones.

Definición: Una cláusula se define como una disyunción finita de cero o más literales.

6.1.2 Elementos de la teoría de la demostración

Como se mencionó anteriormente, la teoría de la demostración establece relaciones en el lenguaje de la lógica de primer orden considerando la generación de sentencias a partir de otras, es decir, considera la derivabilidad de una sentencia en el contexto de un conjunto de reglas de reescritura o reglas de transformación [21].

Para explicar esto, imagine un lenguaje L cualquiera, por ejemplo el conjunto de todas las palabras formadas con los símbolos a, b . Ahora considere un conjunto R de reglas que toman la siguiente forma:

“de un conjunto de palabras de *aquel* tipo, derive una palabra de *este* tipo”, es decir,
“del conjunto de cadenas $\{aX, abX\}$, donde X es cualquier cadena, derive una cadena $ababX$ ”

Si se toma cualquier conjunto inicial S de palabras en L , al aplicar varias reglas en R , eventualmente derivaremos una nueva palabra s de S . La totalidad de la aplicación de las reglas en L constituye la relación de derivabilidad caracterizada por R :

$$\vdash = \{ \langle S, s \rangle \mid S \subseteq L, s \in L \text{ y } s \text{ es derivable de } S \text{ usando } R \}$$

El conjunto de sentencias que se tienen inicialmente recibe el nombre de *axiomas*, las sentencias derivadas se conocen como *teoremas* (consecuencias sintácticas), y las reglas de derivación en R se llaman *reglas de inferencia*, las principales reglas de inferencia se describen en la sección 6.1.3. La unión de los axiomas y R constituye el *sistema de inferencia*.

Sistema de inferencia

Un sistema de inferencia consiste en un conjunto de axiomas los cuales son estatutos (sentencias) considerados como válidos, y un conjunto de reglas de inferencia R que es la maquinaria que permite la producción de nuevos estatutos válidos generados a partir de los estatutos previamente obtenidos. En esta estructura, los estatutos válidos producidos por el sistema se llaman *teoremas*. Una *derivación* ó *demostración* de un teorema es la lista ordenada de axiomas, reglas de inferencia y teoremas previamente obtenidos que fueron necesarios para producir aquel teorema. Para un sistema axiomático dado, la *derivabilidad* está dada por la relación binaria \vdash , la cual es reflexiva y transitiva, mas no simétrica. Para una fórmula A la derivabilidad se escribe como $\vdash A$ [20] [21].

Si E es un conjunto de fórmulas, la expresión $E \vdash A$ significa que A es derivable de E ; esto es, A es derivable del sistema si los elementos de E son considerados como axiomas adicionales, en cuyo caso los elementos de E son vistos como hipótesis.

6.1.3 Reglas de inferencia.

Las reglas de inferencia pueden aplicarse a ciertas *fbf*s y un conjunto de *fbf*s para producir nuevas *fbf*s. Una importante regla de inferencia es *modus ponens*.

Modus Ponens

Es la operación que produce la *fbf* W_2 , de la *fbf* W_1 y de la *fbf* $W_1 \rightarrow W_2$. Esto es, si se cumplen dos premisas p_1 y p_2 , se puede concluir c .

$$\begin{array}{l} p_1 : p \rightarrow q \\ p_2 : p \\ \hline c : q \end{array}$$

Especialización Universal.

Es la operación que produce la *fbf* $W(A)$ de la de *fbf* $\forall x W(X)$, cuando A es cualquier símbolo constante.

Usando Modus Ponens y Especialización Universal producimos la *fbf* $W_2(A)$ de las *fbf*s $(\forall x) [W_1(x) \rightarrow W_2(x)]$ y $W(A)$.

Las reglas de inferencia, entonces, producen nuevas *fbf*s derivadas de otras *fbf*s y se llaman en el cálculo de predicados *teoremas*. Una secuencia de aplicaciones de reglas de inferencia constituyen una *demostración del teorema*.

6.2 Unificación y sistemas de reescritura

En diversas aplicaciones es de interés determinar si dos términos tienen alguna correspondencia o instancia en común. Los sistemas de reescritura se basan en la correspondencia (matching) y juegan un papel importante en los sistemas de deducción. En el contexto de la música, el proceso de hacer arreglos a una pieza musical necesita un conjunto de reglas armónicas que se deseen aplicar sobre la

pieza, para esto es necesario buscar la correspondencia entre el antecedente de la regla en alguna parte de la pieza musical y posteriormente sustituir el consecuente de la regla por la parte que correspondió con el antecedente en la pieza musical.

Una sustitución que hace que dos cláusulas se puedan resolver se conoce como *unificador* y el proceso de hallar tales sustituciones se denomina *unificación* [20].

La unificación fue introducida por Robinson como el paso principal de la regla de inferencia llamada *resolución* [20] [18]. Sin embargo la unificación no es sólo empleada en la resolución, su naturaleza de comparación de estructura puede explotarse en muchos casos donde se trata con expresiones simbólicas, por ejemplo en interpretes para lenguajes de ecuaciones, en sistemas que usan una base de datos organizada en términos de producciones, en procesamiento de lenguaje natural donde se usan reglas de inferencia para realizar una derivación semántica a partir de una sentencia de entrada [18]. En este trabajo, se utiliza para enriquecer una pieza musical aplicando reglas de armonía.

A continuación se mencionan conceptos y definiciones básicas sobre términos, sistemas de reescritura y la unificación de estatutos o E-unificación.

Definición: Dado un conjunto de variables V y un conjunto F de símbolos de función, $V \cap F = \emptyset$. El *álgebra libre* sobre V , denotada por $T(F, V)$ contiene el conjunto de todos los términos construidos a partir de los símbolos de función F y las variables en V [18] [19].

Definición: Una *sustitución* es un mapeo σ de V a $T(F, V)$, extendido a un endomorfismo de $T(F, V)$. Se denota por S , el conjunto de todas las sustituciones. Si $\sigma \in S$ y $A \in T(F, V)$, se escribe σA la aplicación de σ en A [19].

Definición: Un *sistema de reescritura de términos* R es un conjunto de pares de términos $\alpha_k \rightarrow \beta_k$ tal que $V(\beta_k) \subseteq V(\alpha_k)$. Se dice que un término A reduce una

ocurrencia u a un término B utilizando la regla $\alpha_k \rightarrow \beta_k$ y reescribe $A \rightarrow_{[u,k]} B$ si existe una correspondencia η de α_k en $A|u$ y $B = A[u \leftarrow \eta(\beta_k)]$. En ocasiones se denota como $A \rightarrow B$ ó $A \rightarrow_u B$. ' \rightarrow ' se conoce como la *relación de reducción* en $T(F, V)$. Se dice que un término A está en forma normal si no existe un término B tal que $A \rightarrow B$ [18] [19].

Como se ve, al tener una ocurrencia u del antecedente α de la regla k en una expresión A , se genera una expresión B sustituyendo la ocurrencia u en A por el consecuente β de la regla k .

A continuación se muestra un pequeño ejemplo de un sistema de reescritura de términos:

$$R_1 = \{g(a) \rightarrow a \wedge f(b)\}$$

Si se tiene el término $A = g(x)$ y se tiene la correspondencia $\eta = g(a) g(x)$, al aplicar la sustitución $\sigma = \{x \leftarrow a\}$ se obtiene que $B = x \wedge f(b)$.

Para determinar si los términos del antecedente de la regla unifican con términos de la expresión, deben cumplir con el siguiente criterio:

Sea ta el término actual del antecedente de la regla y ea el término actual de la expresión.

ta es constante y ea es constante y son iguales

ta es operador y ea es operador e iguales

ta es variable y ea es variable e iguales

ta es variable y ea es variable y diferentes, se agregan a la lista de sustitución

ta es variable y ea es constante, se agrega en la lista de sustitución

Si existen elementos en la lista de sustitución, la sustitución de ea por ta debe realizarse en el resto de la expresión.

A continuación se muestra un ejemplo de este proceso: Sea R el conjunto de reglas que contiene a r_1 y r_2 y e la expresión sobre la que se aplicarán las reglas.

$$r_1: (x * 1) \rightarrow x \qquad r_2: (x + 0) \rightarrow x \qquad e = ((3 * 1) + (5 + 0)).$$

Al aplicar r_1 sobre e , se obtiene la siguiente lista de sustitución $\{(x / 3)\}$, cambiando $e' = (3 + (5 + 0))$.

Aplicando r_2 sobre e' , la lista de sustitución es la siguiente $\{(x / 5)\}$ teniendo finalmente e'' la siguiente forma: $e'' = (3 + 5)$.

6.3 Unificación y arreglo de piezas musicales

Los conceptos de unificación pueden aplicarse en la generación automática de música con la finalidad de obtener variaciones en la pieza, haciendo arreglos a la pieza generada por los mecanismos mencionados en los capítulos 4 y 5.

Para realizar esto, es necesario contar con un de reglas. Estas reglas se tomaron de los conceptos del capítulo 2 de este trabajo y considerando la opinión de músicos Villanueva [25] y Zúñiga [26].

Si por ejemplo se tiene la siguiente regla:

$$CM FM \rightarrow CM CX FM$$

Esta regla indica que un primer grado mayor, seguido de un cuarto grado mayor, puede reemplazarse por una secuencia de grados armónicos en donde se agrega en la parte intermedia un primer grado dominante, que "suaviza" la transición. Se entiende con esto que no se trata de un proceso de generación de una pieza, sino de reescritura de una ya existente.

En el siguiente capítulo se verá a detalle la aplicación de la unificación en el arreglo de piezas musicales.

6.4 Comentarios finales

Existe una gran variedad de áreas en Inteligencia Artificial en donde se aplica la unificación:

- *Procesamiento de Lenguaje Natural*: Este procesamiento se basa generalmente en reglas de transformación, las cuales, por ejemplo, interpretan una sentencia de entrada en una sentencia más apropiada para su representación interna [18].
- *Sistemas Expertos*: Un sistema experto es un programa cuyo desempeño depende de su habilidad para representar y manipular el conocimiento. Frecuentemente el conocimiento se representa en forma de reglas de producción, donde al satisfacerse las precondiciones de una regla, se ejecuta una acción [14] [18].

La unificación es un mecanismo poderoso que puede aplicarse en distintas áreas en donde se realice una comparación de estructura, como es el caso del arreglo de piezas musicales.

Una de las principales características del proceso de unificación es que, al emplear una regla, las expresiones generadas no tienen pérdida de la verdad; aplicada al contexto musical, no hay pérdida de belleza.

Capítulo 7. Generación Automática de Música

7.1 Presentación de la propuesta

La computadora ha demostrado ser una herramienta poderosa que auxilia al hombre a realizar algunas tareas. En las últimas décadas ha crecido el interés por hacer que la computadora sea capaz de tomar decisiones y realizar procesos que hasta hoy solo han sido posibles realizar con asistencia humana.

Con este interés, se han desarrollado técnicas que permitan a la computadora generar música de forma automática haciendo uso de fractales [10] [11] [13], de probabilidad aleatoria o de probabilidad condicional [12]. El inconveniente de estas propuestas es que no consideran tópicos de armonía que garanticen que los resultados tengan un nivel mínimo de calidad.

En el trabajo que aquí se propone, primeramente se consideran los aspectos de armonía que se trataron en el capítulo 3, como son el uso de progresiones, con la finalidad de que la música tenga una dirección. También se usan recursos armónicos que permiten enriquecer una pieza musical y dar mayor variedad a las piezas que se generan.

Como se mencionó en la sección 3.3, la serie funcional respeta el orden tonal y a la vez usa la cualidad dinámica de la escala para formar progresiones agradables al oído. Este trabajo presenta la composición basada en series funcionales.

7.1.1 Representación formal del lenguaje musical

Visto desde el punto de vista de los lenguajes formales, puede decirse que se tienen que generar cadenas formadas por símbolos $T^n \cdot S^o \cdot D^p$ para $n, o, p \geq 1$. Al unir varias de estas cadenas se tiene una pieza musical.

La composición musical (para efectos de generarse automáticamente), puede describirse con la siguiente gramática: Sea $G = (V_N, V_T, P, O)$, $V_N = \{O, T, S, D, H, I, J\}$, $V_T = \{\text{do, re, mi, fa, sol, la, si}\}$, donde P consta de las siguientes producciones:

$O \rightarrow$	TSDO	TSDT
$T \rightarrow$	HT	H
$S \rightarrow$	LS	L
$D \rightarrow$	JD	J
$H \rightarrow$	do	la
$L \rightarrow$	fa	re
$J \rightarrow$	sol	mi

Esta serie se representa en forma gráfica por el autómata de la Figura 21:

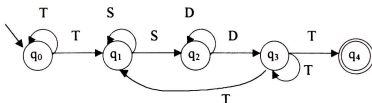


Figura 21 Primer AFN para composición

Esta figura muestra un autómata de estado finito no-determinístico (AFN), pero como se menciona en la sección 5.1.3 es posible obtener un AFD equivalente.

Sea $M = (K, \Sigma, \delta, q_0, F)$ donde:

$$K = \{q_0, q_1, q_2, q_3, q_4\}$$

$$\Sigma = \{T, S, D\}$$

$$F = \{q_4\}$$

Las transiciones de δ se describen en la Tabla 15.

Estados \ Entradas	T	S	D
q ₀	{ q ₀ , q ₁ }	ϕ	ϕ
q ₁	ϕ	{ q ₁ , q ₂ }	ϕ
q ₂	ϕ	ϕ	{ q ₂ , q ₃ }
q ₃	{ q ₁ , q ₃ , q ₄ }	ϕ	ϕ
q ₄	ϕ	ϕ	ϕ

Tabla 14 Transiciones del primer AFN para composición

Para obtener el AFD equivalente, se aplica el proceso descrito en la sección 5.1.3.

Sea $\{q_0\} = q_0'$

$\delta(q_0', T) = \delta'(\{q_0\}, T) = \{q_0, q_1\} = q_1'$

$\delta(q_0', S) = \delta'(\{q_0\}, S) = \phi$

$\delta(q_0', D) = \delta'(\{q_0\}, D) = \phi$

$\delta(q_1', T) = \delta'(\{q_0, q_1\}, T) = \{q_0, q_1\} \cup \phi = q_1'$

$\delta(q_1', S) = \delta'(\{q_0, q_1\}, S) = \phi \cup \{q_1, q_2\} = q_2'$

$\delta(q_1', D) = \delta'(\{q_0, q_1\}, D) = \phi \cup \phi = \phi$

$\delta(q_2', T) = \delta'(\{q_1, q_2\}, T) = \phi \cup \phi = \phi$

$\delta(q_2', S) = \delta'(\{q_1, q_2\}, S) = \{q_1, q_2\} \cup \phi = q_2'$

$\delta(q_2', D) = \delta'(\{q_1, q_2\}, D) = \phi \cup \{q_2, q_3\} = q_3'$

$\delta(q_3', T) = \delta'(\{q_2, q_3\}, T) = \phi \cup \{q_1, q_3, q_4\} = q_4'$

$\delta(q_3', S) = \delta'(\{q_2, q_3\}, S) = \phi \cup \phi = \phi$

$\delta(q_3', D) = \delta'(\{q_2, q_3\}, D) = \{q_2, q_3\} \cup \phi = q_3'$

$\delta(q_4', T) = \delta'(\{q_1, q_3, q_4\}, T) = \phi \cup \{q_1, q_3, q_4\} \cup \phi = q_4'$

$\delta(q_4', S) = \delta'(\{q_1, q_3, q_4\}, S) = \{q_1, q_2\} \cup \phi = q_2'$

$\delta(q_4', D) = \delta'(\{q_1, q_3, q_4\}, D) = \phi \cup \phi \cup \phi = \phi$

Y se obtiene el autómata de la Figura 22:

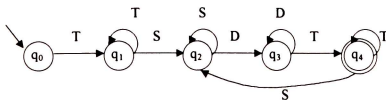


Figura 22 Primer AFD para la composición

Hasta este punto no se ha tratado la duración de cada acorde (terminales generados por las no-terminales H, L, J), que se indican en la siguiente gramática con el no-terminal K. Tampoco se ha considerado la octava en la que se ejecutan los acordes. Aparte de esto, es necesario considerar también la presencia de silencios dentro de la pieza, pero aunque estos pueden estar presentes en cualquier parte de la pieza, se usarán únicamente al terminar una "frase", esto es una cadena formada por T-S-D, y se indica con el no-terminal M, resultando la siguiente gramática⁷:

Sea $G = (V_N, V_T, P, O)$, $V_N = \{O, T, S, X, H, L, J, M, A, B, C, D, E, F, G\}$, $V_T = \{I_i, II_i, III_i, IV_i, V_i, VI_i, VII_i\}$ para $2 \leq i \leq 6$, donde P consta de las siguientes producciones:

O	→	TSXO		TSXMO		TSXT				
T	→	HT		H						
S	→	LS		L						
X	→	JX		J						
H	→	CK		AK						
L	→	FK		DK		BK				
J	→	GK		EK						
K	→	redonda		blanca		negra		corchea		dcorchea
M	→	sredonda		sblanca		snegra		scorchea		sdcorchea
C	→	I ₂		I ₃		I ₄		I ₅		I ₆
D	→	II ₂		II ₃		II ₄		II ₅		II ₆
E	→	III ₂		III ₃		III ₄		III ₅		III ₆
F	→	IV ₂		IV ₃		IV ₄		IV ₅		IV ₆
G	→	V ₂		V ₃		V ₄		V ₅		V ₆
A	→	VI ₂		VI ₃		VI ₄		VI ₅		VI ₆
B	→	VII ₂		VII ₃		VII ₄		VII ₅		VII ₆

Esta gramática es independiente de una escala en particular, es decir, que se expresa en grados de la tonalidad. El detalle que hay que considerar es que las piezas deben terminar con el primer grado de la tonalidad; por ejemplo en la tonalidad de C, los grados tonales son C y A, pero las piezas deben terminar con C, por lo que la tercer producción de P (TSXT), debe reemplazarse por TSXC. El autómatas que resulta de tomar estas consideraciones se muestra en la Figura 23:

⁷ En esta gramática los grados dominantes se identifican con una X en lugar de la D que se usó en la primer gramática, con la finalidad de evitar que se confunda con la D del no-terminal que identifica el segundo grado de la tonalidad.

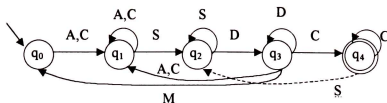


Figura 23 Segundo AFN para la composición

En este autómata se reemplazó la variable T por los no terminales relacionados al primer y sexto grado, que son los tonales. Este autómata es no-determinístico, por lo que se aplicó un proceso similar al caso anterior, obteniendo el autómata que se muestra en la Figura 24:

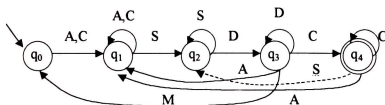


Figura 24 Segundo AFD para la composición

Los símbolos S y D se refieren a los grados de la tonalidad que tienen cualidad subdominante y dominante respectivamente. Estos símbolos se reemplazaron por los no terminales asociados a los grados que tienen estas cualidades. El autómata resultante se muestra en la Figura 25:

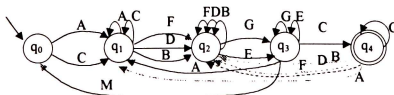


Figura 25 AFD con el reemplazo de cualidad funcional

Si se consideran 5 octavas para la ejecución de cada uno de los grados de la tonalidad, en el autómata anterior faltaría reemplazar cada uno de los no-terminales por los grados armónicos en cada una de las 5 octavas. Este autómata tendría que ser más complejo si se considera que cada uno de estos grados armónicos puede tener 5 duraciones diferentes (redonda, blanca, negra, corchea y semicorchea), lo cual da una probabilidad de $1/25$, restando por considerar la cantidad de nodos que salen de dicho nodo, lo que para asignar la probabilidad de un grado armónico se usaría la siguiente fórmula:

$$p(g_armón) = 1/Noctavas * 1/Nfiguras * 1/Nsalidas$$

Por ejemplo, para q_0 que tiene dos grados como salientes, A y C, cada uno tiene la siguiente probabilidad = $1/5 * 1/5 * 1/2 = 1/50$.

El programa de aplicación debe permitir al usuario elegir la ponderación que tendrán los grados armónicos para determinar la frecuencia de su aparición en la pieza. Pero asignar probabilidades a cada tipo de figura de cada grado que sale de un nodo puede resultar una tarea desgastante, por lo que se tomó la opción de que el usuario indique únicamente la distribución de probabilidad a la que prefiere que se ajusten las figuras. Las distribuciones que pueden elegirse en cada caso son: uniforme, normal y exponencial.

Desde el punto de vista musical, tomar en forma aleatoria la octava en la que se ejecuta un grado armónico no es la mejor opción. Por ejemplo, considere que se está en la tonalidad de *C Mayor* y que el autómata genera la cadena C-A-B-G-C, y cada grado se ejecuta en cualquier octava. El sonido no es agradable, incluso si todos se ejecutaran en la misma octava. Según algunos músicos [25] [26], es conveniente que la distancia que separa las notas más graves de los acordes que se ejecutan de forma consecutiva sea mínima. Por ejemplo, si se toma el acorde del primer grado seguido del sexto y el sexto se ejecuta en su posición raíz en la misma escala que el primer grado, considerando la escala cromática, la distancia

que los separa es de 9 semitonos. Si se considera la propiedad de los acordes de ser indiferentes a la distribución (sección 3.3), y el acorde del sexto grado (que tiene como tonos a-c-e) se ejecuta en su primer inversión, es decir c-e-a, la distancia en semitonos que separa al primer y sexto grado es igual a 0.

Ahora considere que el autómata genera la cadena C-B-F-G-C. La menor distancia entre C-B se obtiene al tomar ambos acordes en su posición raíz, pero ejecutando B una octava más abajo que C. Si se toman los elementos B-F, donde el acorde de B = b-d-f y F = f-a-c, la distancia más corta entre estos dos acordes se obtiene al ejecutar F en su segunda inversión y una octava más arriba de la que se usó para ejecutar B. Si después se ejecuta G = g,b,d, como F está en su segunda inversión, la distancia más corta se obtiene al tomar G en su primera inversión y una octava más abajo que la de F. Esto indica que la ejecución de un acorde depende de la inversión y la octava del acorde inmediato anterior.

Sean $A_{x-1, i, o}$ y $A_{x, j, p}$, donde x representa su posición en la cadena, i, j para $0 \leq i, j \leq 2$ representa la inversión de cada acorde y o, p indican la octava en la que se ejecuta cada uno de los acordes. Para determinar j y p, se toma la inversión que presente la menor distancia entre el tono raíz de i y el tono que funcionó como raíz considerando la inversión j. Esta comparación debe hacerse para $o-1 \leq p \leq o+1$.

Esto produce 9 comparaciones en total, pero puede hacerse únicamente con 3 al considerar lo siguiente:

La escala cromática se forma por 12 semitonos. Sean q_1 y q_2 la distancia en semitonos entre dos tonos con respecto del tono de do, y sea d_1 la distancia entre estos dos tonos, es decir, $d_1 = q_2 - q_1$, si $|d_1| > 6$, implica que hay una distancia $d_2 < 6$ si se toma alguno de los dos tonos en alguna de las octavas vecinas, es decir, $d_2 = 12 - |d_1|$. Por ejemplo, si se considera que los tonos C-A se ejecutan en la misma octava k, $q_1 = 0$ y $q_2 = 9$, la distancia que los separa es $d_1 = 9$, si se considera que A se ejecuta en la octava k-1, la distancia que los separa es $d_2 = 3$ ó $d_2 = 12 - |d_1|$. Si

por el contrario se consideran los tonos en orden inverso A-C, $q_1 = 9$ y $q_2 = 0$, siendo $d_1 = -9$, $d_2 = 3$ si se ejecuta C en la octava $k+1$.

Como se ve en el ejemplo, para el caso en el que $d_1 > 6$ o $d_1 < -6$, el signo de la distancia d_1 determina cuál de las octavas vecinas se considera, si d_1 es positivo se toma la octava $k-1$ de lo contrario se toma la octava $k+1$.

A continuación se muestra el algoritmo para determinar la inversión j y la octava p en que se ejecutará el acorde Ax dado que el acorde $Ax-1$ tiene la inversión i y se ejecutó en la octava o .

Si $Ax-1$ no existe

$p = \text{octava_por_omision}$

$j = 0$; es decir sin inversión

salir

$\text{raizAnt} = \text{tono que funcionó como raíz en } Ax-1 \text{ con inversión } i$

para $z=0$ hasta $z < 3$; esto es, para cada una de las inversiones

$\text{raizAct} = \text{tono raíz de } Ax \text{ con la inversión } z$

$q_1 = \text{distancia entre } do \text{ y } \text{raizAnt}$

$q_2 = \text{distancia entre } do \text{ y } \text{raizAct}$

$\text{distancia} = q_2 - q_1$

Si $\text{distancia} = 0$ entonces

$j = i$

$p = o$

salir

Si $\text{distancia} > 6$ ó $\text{distancia} < -6$ entonces

Si $\text{distancia} > 0$ entonces

$\text{distancia} = 12 - (q_2 - q_1)$

Si existe do entre raizAct y raizAnt entonces, hay cambio de octava

$\text{octava} = -1$

sino

```

                                octava = 0                                (continuación...)
sino
    distancia = 12 - (q1 - q2)
    Si existe do entre raizAnt y raizAct entonces
        octava = 1
    sino
        octava = 0
sino
    si distancia < 0
        distancia = distancia*-1
    octava = 0
si distancia < min_distancia entonces
    min_distancia = distancia
    j = z
    octava_final = octava
fin para
p = o + octava_final

```

En la sección 5.2.1 se mencionó que las probabilidades de los autómatas probabilísticos se pueden representar con una matriz de transición, en donde cada renglón representa las probabilidades asociadas a cada arista que sale de cada nodo. En el diseño de la aplicación se consideran objetos *nodo*, y cada nodo contiene una colección de objetos *arista* que salen del nodo. Los objetos *arista* contienen la probabilidad que el usuario le asigna y el tipo de distribución de probabilidad al que se ajustan las duraciones que se generan.

Para la generación automática de cadenas, un acorde se describe de la siguiente forma: $Ac = (n, m, o, i, f, x, T, s)$ donde para el acorde Ac , n es un identificador de su nombre, m indica el modo del acorde, o la octava en la que se ejecuta, i la inversión, f la figura del acorde (r = redonda, b = blanca, n = negra, c = corchea, s = semicorchea), x si el acorde es bemol, sostenido o natural, T es el

conjunto de tonos que forman el acorde de acuerdo al modo y s indica si el acorde es de sexta en lugar de séptima.

El algoritmo que se utiliza para la generación de cadenas es el siguiente:

```
Inicializar la semilla para generar número aleatorios
```

```
nodo_actual = 0
```

```
ciclo infinito
```

```
    aleatorio1 = número aleatorio
```

```
    se busca la arista que se identifica por aleatorio1 según la ponderación del usuario
```

```
    se asigna a  $m$  el modo correspondiente al grado que representa la arista
```

```
    se obtiene el tipo de distribución de las figuras de ese grado en ese nodo
```

```
    aleatorio2 = número aleatorio
```

```
    se asigna a  $f$  la figura según aleatorio2
```

```
    se busca la inversión de  $Ac$  utilizando el algoritmo que me describió anteriormente
```

```
    se agrega  $Ac$  a una lista de Acordes
```

```
    se toma como nodo_actual el destino de la arista que se seleccionó
```

```
    si el nodo_actual es igual al estado final ( $q_4$  de la Figura 25) entonces
```

```
        preguntar si se desea terminar de generar acordes
```

```
        si desea terminar
```

```
            salir de la función
```

```
fin ciclo infinito
```

7.2 Diseño de la aplicación

La aplicación que se desarrolló se divide principalmente en dos partes: una máquina para la composición de piezas musicales usando un autómata

probabilístico, y un sistema de basado en reglas, en este caso reglas de la teoría armónica, que permiten reescribir o hacer arreglos a la pieza musical.

El funcionamiento del sistema se representa en forma gráfica en la Figura 26. El proceso (2) ha sido formalizado en la sección anterior.

Como se mencionó en la sección 6.2, la unificación puede usarse en aplicaciones que manipulan la estructura de expresiones. En la resolución se producen nuevas expresiones a partir de otras ya existentes de forma que se garantiza que las nuevas expresiones sean válidas. Se describe en esta sección los procesos (4) y (5) del esquema funcional de la aplicación desarrollada.

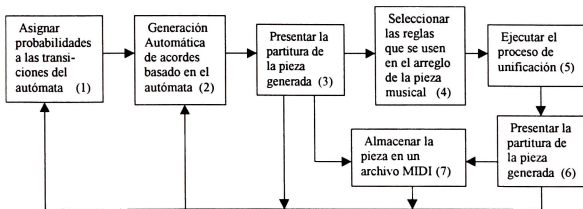


Figura 26 Esquema operativo de la aplicación

Si se considera la música como una secuencia de palabras en donde cada palabra es de la forma $Ac = (n, m, o, i, f, x, T, s)$, es posible reemplazar conjuntos de palabras por otras que se consideren más agradables para quien escuche la música.

Para esto es necesario definir un conjunto axiomático o conjunto de reglas que se apliquen en el proceso de unificación. De los conceptos de armonización del capítulo 3, se consideraron recomendaciones de los músicos Villanueva [25] y

Zúñiga [26], para tomar instancias de dichos conceptos. Las reglas que se consideraron se muestran en la Tabla 15:

<i>Antecedente</i>	<i>Consecuente</i>
CM FM	CM Cx FM
Gx CM	Gx Em bEx Dm bDx CM
FM Gx CM	FM G+6 Gx CM
FM Gx Am Dm	FM Em Ax Dm
FM Em Am	FM Ex Am
CM Bo Em Am	CM Bo Ex Am
Em Am	Ex Ax
Am Dm	Ax Dm
Dm Gx CM	Dm bDx CM

Tabla 15 Reglas de la aplicación

Para realizar la unificación fue necesario establecer una notación que contemplara los parámetros a considerar en cada una de las reglas. La notación que se utiliza es la siguiente:

Sea R una regla, $R = \{Ant, Cons\}$ donde Ant y Cons representan el antecedente y el consecuente de la regla. Tanto Ant como Cons se forman por un conjunto de acordes descritos de la siguiente forma: Acord = {nr, mr, ar, sr} en donde nr representa el grado armónico del acorde, mr el modo del acorde, ar el accidente del acorde (natural, bemol, sostenido) y sr indica si se trata de un acorde de sexta (1 de sexta, 0 de séptima).

Considerando esta notación, las reglas de la Tabla 15 se expresan de la siguiente forma:

1. $CMn0 FMn0 \rightarrow CMn0 Cxn0 FMn0$
2. $Gxn0 CMn0 \rightarrow GXn0 Emn0 EXb0 Dmn0 Dxb0 CMn0$
3. $FMn0 Gxn0 CMn0 \rightarrow FMn0 GMn1 Gxn0 CMn0$
4. $FMn0 Gxn0 Amn0 Dmn0 \rightarrow FMn0 Emn0 Axn0 Dmn0$
5. $FMn0 Emn0 Amn0 \rightarrow FMn0 Exn0 Amn0$

6. CMn0 Bon0 Emn0 Amn0 → CMn0 Bon0 Exn0 Amn0
7. Emn0 Amn0 → Exn0 Amn0
8. Amn0 Dmn0 → Axn0 Dmn0
9. Dmn0 Gxn0 CMn0 → Dmn0 Dxb0 CMn0

La aplicación permite al usuario la posibilidad de seleccionar las reglas que desea utilizar para el arreglo de la pieza musical.

El proceso de unificación se muestra en el siguiente algoritmo:

Sea LAnt el antecedente de la regla en turno

LCons el consecuente de la regla

LAcYa la lista de acorde que ya analizaron

LAcorde lista de acordes de la pieza generada por la aplicación

|LAcordes| representa el número de acordes o cardinalidad de LAcordes

Para cada una de las reglas seleccionadas

Para cada acorde de la pieza generada por la aplicación

Si Unifica(LAcorde con el LAnt) y la duración de lo que unifica con

LAnt alcanza para asignar duración a los elementos de LCons entonces

Para cada Acorde de LCons

Buscar la inversión y la octava en que se ejecutará

Si |LAnt| = |LCons| entonces

Duración(Ai)=Duración(Bi) Ai∈LCons Bi ∈ LAcorde

Sino

Distribuir la duración en el consecuente

Agregar al final de LAcYa LCons

Si falta duración por asignar en LAcYa entonces

Crear silencio con la duración faltante

Remove el acorde de LAcorde

Sino

(continuación...)

Agregar al final de LAcYa el primer elemento de LAcorde

La función *Unifica* se describe con el siguiente algoritmo:

LAcordes = Lista de acordes que faltan por analizar

LAnt = Antecedente de la regla que se desea verificar

mientras existan elementos en LAcordes y existan elementos en LAnt

 Ac = i-ésimo elemento de LAcordes

 Ant = i-ésimo elemento de LAnt

 Si el nombre de Ac es diferente al de Ant o el modo de Ac es diferente al de Ant

 Regresar Falso

fin mientras

regresar Verdadero

Hay que notar que en el sistema que se desarrolló, la unificación no solo se determina con el empatamiento de los elementos de la lista de acordes generados por la misma aplicación y los acordes del antecedente de las reglas seleccionadas, también es necesario considerar que la duración total de los elementos que se unificaron con el antecedente de la regla sea suficiente para asignar duración a los elementos del consecuente de la regla.

Ejemplo: Considere que la aplicación genera la siguiente cadena:

CM50sn7 FM52sn7 Em42cn7 CM50nn7

A primera vista parece que los dos primeros elementos de esta cadena resolverían con la primera regla, pero esto no es posible, ya que la duración de los términos que unifican es de semicorchea, y como el sistema no maneja duraciones

menores a semicorchea, no se le asignaría duración al tercer elemento del consecuente de la primera regla.

Una vez que se valida que la duración es suficiente para asignar valores de duración a cada elemento del consecuente de la regla, el sistema asigna la duración a los elementos del consecuente de la regla haciendo la siguiente consideración:

Si la cardinalidad del antecedente de la regla es la misma que la del consecuente (visto en el algoritmo como Si $|L_{Ant}| = |L_{Cons}|$ entonces), se asigna a cada elemento del consecuente de la regla la duración de los elementos que resolvieron en la cadena generada por el autómata. Esto se aplica en las reglas 4, 5, 6, 7 y 8.

Si la cardinalidad difiere, entonces se distribuye de manera uniforme la duración acumulada por el resolvente entre la cardinalidad del consecuente de la regla; este valor acumulado se maneja en número de semicorcheas. Esta consideración se aplica en las reglas: 1, 2, 3 y 9.

7.3 Implementación y pruebas

El sistema se desarrolló utilizando el paradigma orientado a objetos y la implementación se realizó en el ambiente Windows utilizando Visual C++ versión 5.0.

7.3.1 Clases

Para implementar el autómata se definieron las siguientes clases:

CArista: Contiene información relacionada con las aristas del autómata, como es la probabilidad de tomar dicha arista en la transición, la distribución de probabilidad para determinar el tipo de figura, y el nodo al que se dirige la arista.

CNodo: Objeto que describe un estado del autómata y contiene un conjunto de aristas.

CRegla: Objeto que modela una regla de producción. Contiene dos listas de descriptores del antecedente y consecuente de la regla.

El código de los prototipos de las principales clases se muestra a continuación:

```
class CEtiqueta : public CObject
{protected:
    CString m_nombre;
public:
    CEtiqueta();
    char *RegresaNombre();
    CString RegresaNombreCS();
};

class CArista : public CEtiqueta
{ protected:
    BYTE m_bProba;
    BYTE m_bDFig;
    BYTE m_bSiguiente;
public:
    CArista();
    CArista(char *);
    CArista(char *, BYTE, BYTE);
    virtual ~CArista();
    BOOL AsignaValores(int, int);
    BYTE RegresaProba();
    BYTE RegresaFig();
    BYTE RegresaSig();
};

class CNodo : public CEtiqueta
{ protected:
    int m_nAristas;
    CArista *m_Aristas[5];
public:
    CNodo();
    CNodo(char *, int);
    int RegresaNAristas();
    CArista *RegresaArista(int);
    virtual ~CNodo();
    BOOL AsignaArista(int, CArista *);
};
```

```

class CRegla : public CObject
{ protected:
    COBList m_LAntecedente, m_LConsecuente;
public:
    CRegla();
    CRegla(CString, CString);
    virtual ~CRegla();
    COBList *RegresaAntecedente();
    COBList *RegresaConsecuente();
};

```

(continuación...)

7.3.2 Métodos importantes

El código del método que genera la cadena de grados armónicos se muestra a continuación:

```

void CInter5View::OnAutmataGeneracadenas()
{
    int producciones, padre_actual, aleatorio, n_aristas, suma, arista;
    int aleatdura, tipoDist, figura;
    CString Acorde, AcordeAux;
    CACorde *cac_ant=NULL, *cac_act=NULL, *cac_aux=NULL;
    COBList LAcorde;

    char nombres[5]={'r','b','n','c','d'};
    char datos_cac[7];
    char Grado;
    if (!m_TreeDoc)
        m_TreeDoc = GetDocument();
    m_TreeDoc->LimpiaListaAcorde();

    srand((unsigned)time( NULL ));
    for (padre_actual = 0, producciones = 0, ; producciones++)
    {
        aleatorio = rand()%100;
        n_aristas = m_TreeDoc->RegresaPadres(padre_actual)->RegresaNAristas();
        // ve a cuál de las aristas le corresponde la proba generada
        for (suma = 0, arista = 0, arista < n_aristas, arista++)
        {
            suma+=m_TreeDoc->RegresaPadres(padre_actual)->RegresaArista(arista)->RegresaProba();
            if (aleatorio < suma)
                break; // encontró la arista
        }
    }
}

```

(continuación...)

```
AcordeAux=m_TreeDoc->RegresaPadres(padre_actual)->RegresaArista(arista)
->RegresaNombreCS();

//MODO
Grado = AcordeAux{AcordeAux.GetLength()-1}; // ve si es C,D,...
datos_cac[0] = Grado;

switch (Grado)
{
case 'M':datos_cac[1]='n';break; // es silencio
case 'C':
case 'F': datos_cac[1] = 'M';
          break;
case 'G': datos_cac[1] = 'X';
          break;
case 'B': datos_cac[1] = 'o';
          break;
default: datos_cac[1] = 'm';
          break;
}

// DURACION
// obtiene el tipo de dist. de duración para buscarla en la matriz de duraciones
tipoDist = m_TreeDoc->RegresaPadres(padre_actual)->RegresaArista(arista)->RegresaFig();
aleatdura = rand()%100; // obtiene otro aleatorio para la duración

// busca la duración correspondiente la distribución
for (suma = 0, figura = 0; figura < NUM_FIGURAS; figura++)
{
    suma += FunProba[tipoDist][figura];
    if (aleatdura < suma)
        break; // encontro la figura segun el renglon tipoFig
}

// Código para cambiar la octava y la inversion
cac_act=new CACorde(datos_cac[0],datos_cac[1]);
cac_act->BuscaInversion(cac_ant);
sprintf(datos_cac,"%d%d",cac_act->RegOct(),datos_cac[1]=cac_act->RegInv());

cac_act->AsignaFig(nombres[figura]);
Acordes+=cac_act->RegDatos();

LAcorde.AddTail(cac_act);
if (cac_act->m_Nombre != 'M') // si no es silencio, lo guardo
    // para usarlo después en las inversiones
    cac_ant = cac_act;

// AVANZA EN EL GRAFO
padre_actual=m_TreeDoc->RegresaPadres(padre_actual)->RegresaArista(arista)
->RegresaSig();
```

```

        if (padre_actual == (NUM_PADRES-1)) // Si está en q4 puede terminar o seguir
        {
            char cadena[60];
            sprintf(cadena,"se tienen %d acordes, ¿Desea terminar ahora?",producciones+1);
            if (MessageBox(cadena,NULL,MB_ICONINFORMATION | MB_YESNO) == IDYES)
                break;
        }
    }
    m_TreeDoc->CopiaLAcordes(LAcorde);
    AfxMessageBox(Acorde.GetBuffer(MAXPROD));
    m_TreeDoc->MuestraLAcordes();
    Invalidate();
}

```

El código de la función que busca la inversión de un acorde es el siguiente:

```

/*-----
Método :      BuscaInversion()
Objetivo:     Encontrar la inversion y la octava en que se ejecutara este
               acorde de acuerdo a la inversión y octava de Ant,
               considera bemol y sostenidos de acuerdo al campo accidente
Parámetros:  Ant: Acorde que se toco anteriormente
Salida:      Ninguna
-----*/

void CACorde::BuscaInversion(CACorde *Ant)
{
    int Modos[4][7]={ {2,2,1,2,2,2,1},{2,2,1,2,2,1,2},{2,1,2,2,1,2,2},{2,1,2,1,2,2,2}};
    // distancias mayor, dominante, menor, disminuido
    char Cromat[]="CcDdEeFfGgAaBbcDdEeFfGgAaB";
    char CromatC[25];
    char raizAnt=0,raizAct=0;
    int desplaza=0,n_tonos=0,n_modos,tono,octava,i,j;
    int min_dist=100,inversion,distancia,octava_final;
    if (m_Nombre == 'M') // si es silencio no se le puede calcular inversion
        return;

    if (Ant == NULL || Ant->m_Nombre == 'M') // si no hay anterior o es silencio
    { //entonces se queda sin inversion en octava 3
        m_Octava = OCATAVA_DEFAULT;
        m_Inversion=0;
        return;
    }

    if (Ant->m_Nombre == m_Nombre && Ant->m_Modo == m_Modo)
    {
        m_Octava=Ant->m_Octava;
        (Ant->m_Inversion)? m_Inversion=Ant->m_Inversion-1 : m_Inversion=1;
        return;
    }
    n_tonos = Ant->m_Inversion*2;
}

```

```

switch(Ant->m_Modo)
{
    case 'M': n_modos = 0; break;
    case 'X': n_modos = 1; break;
    case 'm': n_modos = 2; break;
    case 'o': n_modos = 3; break;
}

// Sacar el tono raiz de Ant segun su inversion y considerando modo
for (desplaza=0,tono=0;tono < n_tonos; tono++)
    desplaza+=Modos[n_modos][tono];
raizAnt=Cromat[posicion_cadena(Cromat,Ant->m_Nombre)+desplaza+Ant->m_Accidente];

//Obtener modo del actual
switch(Act->m_Modo)
{
    case 'M': n_modos = 0; break;
    case 'X': n_modos = 1; break;
    case 'm': n_modos = 2; break;
    case 'o': n_modos = 3; break;
}

// ya que sé la raiz de anterior, busca cual inversion le corresponde a este acorde
// tiene menor diferencia con raizAnt

i= posicion_cadena(Cromat, raizAnt);
for(n_tonos=0,inversion=0,inversion<3;n_tonos+=2,inversion++) //a cada octava busca sus 3 inversiones
{
    for (desplaza=0,tono=0;tono < n_tonos; tono++)
        desplaza+=Modos[n_modos][tono];
    raizAct=Cromat[posicion_cadena(Cromat, m_Nombre)+desplaza+ m_Accidente];
    //sacar distancia entre raizAnt y esta raizAct con esta inversion
    j= posicion_cadena(Cromat, raizAct);
    distancia=j-i;
    if (!distancia)
        { // sale porque es la distancia minima
            m_Octava = Ant->m_Octava;
            m_Inversion = inversion;
            return;
        }
    if (distancia > 6 || distancia < -6) // si es mayor a la mitad del teclado veo de atras de Ant
    {
        memset(CromatC,0,sizeof(CromatC));
        if (distancia > 0)
            { // este acorde estaba despues de Ant, pero como se volteo, ahora esta antes de Ant
                distancia=12-(j-i); //obtengo el complemento de la cromatica
                strcpy(CromatC,Cromat+1+posicion_cadena(Cromat,raizAct),distancia);
                (strchr(CromatC,'C')) ? octava = -1 : octava = 0;
            }
    }
}

```

```

else
    // este acorde estaba antes que Ant, pero ahora esta despues de Ant
    distancia=12-(i-j);
    strncpy(CromatC,Cromat+1+posicion_cadena(Cromat,raizAnt),distancia);
    (strchr(CromatC,'C')) ? octava = 1: octava = 0;
}
else
{
    if (distancia < 0) distancia*=-1;
    octava = 0;
}
if (distancia < min_dist)
{
    min_dist = distancia;
    m_inversion = inversion;
    octava_final=octava; // ve si hay que modificar oct
}
}
if (Ant->m_Octava+octava_final < 2)
    m_Octava = 2; // por si llegara a estar mas abajo de la segunda octava
else if (Ant->m_Octava+octava_final > 5)
    m_Octava = 5; // por si fuera mas agudo que la 5a octava
m_Octava = Ant->m_Octava+octava_final;
}

```

7.3.3 Operatividad

El sistema tiene los siguientes menús de opciones:

Archivo	Ver	Autómata	Ayuda
Nuevo	Barra de Herramientas	Modificar	Acerca de...
Abrir	Barra de estado	Probabilidades	
Guardar		Generar cadenas	
Guardar como...		Seleccionar reglas	
Lista de archivos recientes		Unificar	
Salir		Visualizar	

Archivo

Nuevo: Si existe una pieza activa, libera su memoria e inicializa las variables para crear una nueva pieza musical.

Abrir. Esta opción permite leer un archivo MIDI en formato 0 y despliega su partitura. Cuando se lee un archivo no es posible aplicar la unificación.

Guardar y Guardar como...: Guarda la pieza musical en un archivo MIDI con formato 0, esto permite que se ejecute o se edite en otras aplicaciones.

Lista de archivos recientes: Mantiene una lista de la ruta de acceso de los cuatro últimos archivos manejados por la aplicación.

Salir. Termina la aplicación.

Ver

Barra de Herramientas: Muestra u oculta la barra de herramientas.

Barra de Estado: Muestra u oculta la barra de estado, esta indica el estado de la aplicación o una explicación de los menús que se pueden seleccionar al pasar el ratón sobre estas opciones del menú,

Autómata

Modificar probabilidades: Esta opción permite asignar la probabilidad a cada una de las transiciones del autómata que se indica en la sección 7.1 (Figura 25). Inicialmente la probabilidad de cada nodo se divide de manera uniforme de acuerdo con el número de aristas que salen del nodo.

Generar cadenas: Ejecuta el autómata con las probabilidades que tiene el autómata. Cada vez que el autómata llega al estado final, muestra los acordes que se formaron y pregunta si desea terminar la generación o si desea que se generen más acordes.

Seleccionar Reglas: Despliega las reglas que se pueden usar agrupándolas según la característica de la teoría armónica de donde se tomaron.

Unificar. Realiza el proceso de unificación considerando las reglas que seleccionó el usuario.

Visualizar. Redibuja la partitura de la pieza musical.

Ayuda

Acerca de...: Contiene la versión y año de realización del sistema.

Los elementos de la barra de herramientas se muestran en la Figura 27.

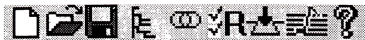


Figura 27 Barra de herramientas de la aplicación

Los elementos se asocian con las opciones de menú en el siguiente orden:

- | | |
|-----------------------------|-----------------------|
| 1. Nuevo | 6. Seleccionar_Reglas |
| 2. Abrir | 7. Unificar |
| 3. Guardar | 8. Visualizar |
| 4. Modificar_probabilidades | 9. Acerca_de |
| 5. Generar_cadenas | |

La partitura se despliega de acuerdo a reglas de notación musical utilizadas por Maldonado y Zizumbo [9].

Las probabilidades se asignan al autómata utilizando el cuadro de diálogo de la Figura 28. Cada estado presenta del lado derecho la probabilidad acumulada en las aristas correspondientes.

Una vez que se selecciona la opción de *Generar_cadenas*, el programa presenta una descripción de los grados armónicos generados, como se muestra en la Figura 29.

Posteriormente se muestra la partitura correspondiente a la pieza que se generó (Figura 30).

La selección de las reglas que se usarán en el proceso de reescritura de la pieza, se hará por medio del cuadro de diálogo de la Figura 31.

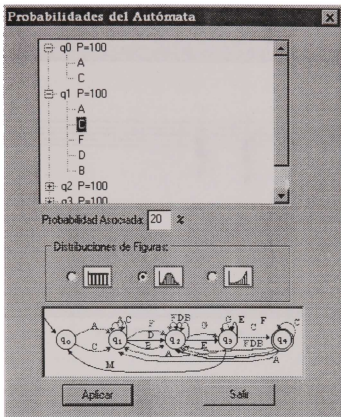


Figura 28 Diálogo para asignar probabilidades al autómata

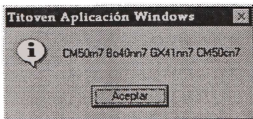


Figura 29 Grados generados por la aplicación

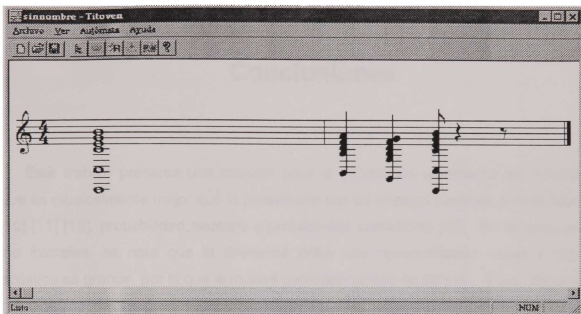


Figura 30 Partitura de la pieza

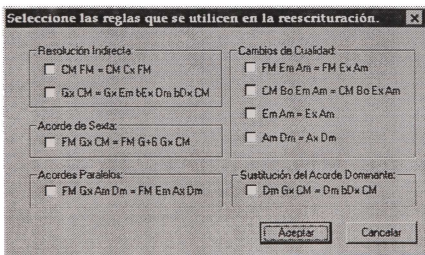


Figura 31 Diálogo para seleccionar reglas

Como siguiente paso se muestran los grados generados y el número de unificaciones que se produjeron. Por último se muestra la partitura, de igual forma que cuando se genera la pieza.

Conclusiones

Este trabajo presenta una solución para la generación automática de música que es musicalmente mejor que la presentada por los trabajos basados en fractales [10] [11] [13], probabilidad aleatoria y probabilidad condicional [12]. En el caso de los fractales, se nota que la diferencia entre una representación visual y una acústica es grande, por lo que la música generada carece de sentido. El uso de una clasificación de los grados armónicos con base en la teoría musical nos permite generar música de una forma más eficiente y simple. Es más eficiente porque el usuario aporta menos información siendo una ventaja si el usuario no tiene conocimiento musical para generar una pieza. En el caso de sistemas basados en probabilidad, generar una pieza con sentido requiere que el usuario tenga un cierto nivel de preparación musical, además, requiere de tiempo para proporcionar ese conocimiento a la aplicación.

El uso de sistemas formales para la representación de lenguajes, nos permitió manipular de una mejor manera el lenguaje musical. El uso de autómatas estocásticos permitió al usuario del sistema expresar sus preferencias en los tonos que debían emplearse en las piezas generadas.

Este trabajo extiende las áreas de aplicación del proceso de unificación al permitir la reescritura o arreglo de una pieza musical, utilizando algunas reglas armónicas.

Trabajos Futuros

Nuestro trabajo puede extenderse de la siguiente manera:

El manejo de tiempo, esto es, el manejo de ritmo o cambios en la métrica, es una posibilidad para un trabajo futuro.

Otra extensión sería un sistema en donde dada una pieza musical, determinar su belleza considerando el apego de la misma a las progresiones y demás recursos de la armonía.

Otra posibilidad es estudiar la generación automática de música utilizando gramáticas de Lindenmayer, ya que en la música cualquier sucesión de sonidos puede considerarse una pieza musical formada por terminales.

Glosario

- ◆ *Acorde*: Conjunto de tres tonos o más que se tocan en forma simultánea.
- ◆ *Autómata finito*: Mecanismo de especificación finita de lenguajes infinitos que consta de un conjunto de estados y un conjunto de transiciones entre estados.
- ◆ *Autómata probabilístico*: Autómata en donde las transiciones tienen asociada una probabilidad.
- ◆ *Axioma*: Estatuto considerado como verdadero sin necesidad de demostración.
- ◆ *Escala cromática*: Escala formada por un conjunto de doce tonos sobre los que se basa la música occidental.
- ◆ *Escala diatónica*: Escala que se forma por siete tonos.
- ◆ *Grado armónico*: Acorde asociado a un tono perteneciente a una escala.
- ◆ *Métrica*: Valores que determinan la duración de un compás.
- ◆ *Modo de la tonalidad*: Distribución interválica de los siete tonos que forman una escala diatónica.
- ◆ *Progresión*: Relación que considera las cualidades de los grados armónicos.
- ◆ *Serie funcional*: Secuencia de grados armónicos que respeta el orden y usa la cualidad dinámica de la escala.
- ◆ *Teorema*: Estatuto considerado como verdadero que puede ser demostrado.

Bibliografía

- [1] Paz Azaria. *"Introduction to Probabilistic Automata"*, Academic Press, 1971.
- [2] Salomaa Arto. *"Formal Languages"*, Academic Press, 1973.
- [3] Hopcroft John E. & Ullman Jeffrey D. *"Introduction to Automata Theory, Languages and Computation"*, Adisson Wesley, 1979.
- [4] Hopcroft John E. & Ullman Jeffrey D. *"Formal Languages and their Relation to Automata"*, Adisson Wesley, 1969.
- [5] Zuckerkandl Victor. *"The Sense of Music"*, Princeton University Press, 1959.
- [6] Lendvai Ernő. *"Duality and Synthesis in the Music of Bela Bartok"*
- [7] De la Cerda César A. *"Armonía Tonal Moderna"*, Segunda Edición, 1984.
- [8] Chapa Vergara Sergio V. *"Autómatas Probabilísticos"*
<http://www.cs.cinvestav.mx/Frame.html>
- [9] Maldonado Escobedo Juan Felipe & Zizumbo Castañeda Salvador Ernesto. *"El protocolo MIDI y la notación musical"*. Instituto Tecnológico de Morelia. 1997.
- [10] Massachusetts Institute of Technology. *"Research: Fractal Music"*,
<http://mitpress.mit.edu/e-journals/Computer-Music-Journal/CMJ.links.html>.
- [11] Fractal Music Fans. *"Recursion: A Paradigm For Future Music?"*
<http://www-ks.rus.uni-stuttgart.de/people/schulz/fmusic>

- [12] Christensen Carl M. *"Aleatoric Composer"*, M.S.C.S. Project. Temple University Philadelphia, PA. 1993.
- [13] Przemyslaw Prusinkiewicz & Aristid Lindermayer. *"The Algorithmic Beauty of Plants"*, Springer-Verlag. 1990.
- [14] Curtis Roads. *"Research in Music and Artificial Intelligence"*, MIT, Association for Computing Machinery, New York, N.Y. Vol. 17, Num. 2, Junio 1985.
- [15] C. S. Wetherell. *"Probabilistic Languages: A Review and Some Open Questions"*, Bell Laboratories. Association for Computing Machinery, New York, N.Y. Vol. 12, Num. 4, Diciembre 1980.
- [16] King Sun Fu. *"Pattern Recognition and Applications"* Perdue University. West Lafayette, India. 1982.
- [17] Ulrich Michels. *"Atlas de Música"*. Alianza Editorial S.A., Madrid. 1992.
- [18] Jörg H. Siekmann. *"Unification Theory"* Journal of Symbolic Computation, Volumen 7, 1989.
- [19] Jia-Huai You. *"A Class of Confluent Term Rewriting Systems and Unification"* Journal of Automated Reasoning, Volumen 2, Número 4, D. Reidel Publishing Company, 1986.
- [20] Chapa Vergara Sergio V. *"Lógica Matemática y Aplicaciones"*
<http://www.cs.cinvestav.mx/Frame.html>
- [21] Hogger Christopher John. *"Essentials of Logic Programming"* Imperial College of Science and Technology, London. Ed. Clarendon Press, Oxford. 1990.

- [22] Lerdahl & Jackendoff. *"A Generative Theory of Tonal Harmony"* MIT Press, Cambridge, Mass. 1983.
- [23] Minsky, M. *"A Framework for Representing Knowledge"* M.I.T. Artificial Intelligence Lab. Cambridge, Mass. 1975.
- [24] Fry, C. *"Computer Improvisation"* Computer Music Journal, Vol. 4,3. 1980.

Referencias:

- [25] Villanueva Molina David. Estudiante de música en el Conservatorio de las Rosas, Morelia, Michoacán.
- [26] Zúñiga Peregrina Pablo Andrés. Licenciado en Composición y Arreglo, CCDMAC, Durango, Durango.

Índice de Figuras

Figura 1 El pentagrama	9
Figura 2 Partes de la nota	10
Figura 3 Series y Octavas	12
Figura 4 Acordes y líneas auxiliares	12
Figura 5 Agrupación de notas de acuerdo a la métrica	18
Figura 6 Fragmento de partitura	24
Figura 7 Octava de un piano	26
Figura 8 Diferentes modos del orden diatónico	26
Figura 9 Intervalos en los modos mayor y menor	27
Figura 10 Círculo de quintas	28
Figura 11 Cualidades	31
Figura 12 Círculo de quintas con cualidades funcionales	33
Figura 13 Sustitución del acorde dominante	34
Figura 14 Acordes paralelos	35
Figura 15 Grafo dirigido	38
Figura 16 Árbol de derivación	44
Figura 17 Daigramas de transición	51
Figura 18 Automata finito no-determinístico	52
Figura 19 Automata probabilístico de dos estados	54
Figura 20 AFN para la generación de música	55
Figura 21 Primer AFN para composición	68
Figura 22 Primer AFD para la composición	69
Figura 23 Segundo AFN para la composición	71
Figura 24 Segundo AFD para la composición	71
Figura 25 AFD con el reemplazo de cualidad funcional	71
Figura 26 Esquema operativo de la aplicación	77
Figura 27 Barra de herramientas de la aplicación	89
Figura 28 Diálogo para asignar probabilidades al autómata	90
Figura 29 Grados generados por la aplicación	90
Figura 30 Partitura de la pieza	91
Figura 31 Diálogo para seleccionar reglas	91

Índice de Tablas

<i>Tabla 1 Valor de duración de las notas musicales</i>	10
<i>Tabla 2 Equivalencias entre notas</i>	11
<i>Tabla 3 Clases de llaves</i>	13
<i>Tabla 4 Figuras y valor relativo de los Silencios</i>	15
<i>Tabla 5 Las alteraciones</i>	16
<i>Tabla 6 Términos que se usan para los movimientos</i>	19
<i>Tabla 7 Términos auxiliares</i>	19
<i>Tabla 8 Adverbios para los movimientos</i>	20
<i>Tabla 9 Términos de Acentuación</i>	22
<i>Tabla 10 Términos para matices con cambios graduales</i>	22
<i>Tabla 11 Términos para matices constantes</i>	23
<i>Tabla 12 Ejemplos de escalas</i>	27
<i>Tabla 13 Matriz de transición de un autómata de dos estados</i>	54
<i>Tabla 14 Transiciones del primer AFN para composición</i>	69
<i>Tabla 15 Reglas de la aplicación</i>	78

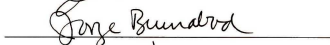
Los abajo firmantes, integrantes de jurado para el examen de grado que sustentará el **Ing. Ernesto Salvador Zizumbo Castañeda**, declaramos que hemos revisado la tesis titulada: **“Generación Automática de Música”** consideramos que cumple con los requisitos para obtener el Grado de Maestro en Ciencias, con especialidad en Ingeniería Eléctrica.

Atentamente,

Dr. Sergio V. Chapa Vergara

A handwritten signature in black ink, appearing to read 'Sergio V. Chapa Vergara', written over a horizontal line.

Dr. Jorge Buenabad Chávez

A handwritten signature in black ink, appearing to read 'Jorge Buenabad Chávez', written over a horizontal line.

Dr. Manuel González Hernández

A handwritten signature in black ink, appearing to read 'Manuel González Hernández', written over a horizontal line.

CENTRO DE INVESTIGACION Y DE ESTUDIOS AVANZADOS DE
INSTITUTO POLITECNICO NACIONAL

BIBLIOTECA DE INGENIERIA ELECTRICA
FECHA DE DEVOLUCION

El lector está obligado a devolver este libro
antes del vencimiento de préstamo señalado
por el último sello.

24 NOV. 2000

DEVOLUCION

