UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

Programa de Maestría y Doctorado en Música

Facultad de Música

LIVE CODING Y MODELOS MATEMÁTICOS EN LA MÚSICA ALGORÍTMICA

TESIS / TESINA
QUE, PARA OPTAR POR EL GRADO DE

MAESTRA EN MÚSICA (Tecnología Musical)

PRESENTA

ANA OFELIA NEGRETE FERNÁNDEZ

TUTOR

DR. PABLO PADILLA LONGORIA (INSTITUTO DE MATEMÁTICAS APLICADAS Y SISTEMAS UNAM)

Ciudad Universitaria, CD. MX, ENERO, 2024

Declaro conocer el Código de Ética de la Universidad Nacional Autónoma de México, plasmado en la Legislación Universitaria. Con base en las definiciones de integridad y honestidad ahí especificadas, aseguro mediante mi firma al calce que el presente trabajo es original y enteramente de mi autoría. Todas las citas de obras elaboradas por otros autores, o sus referencias, aparecen aquí debida y adecuadamente señaladas, así como acreditadas mediante las convenciones editoriales correspondientes.

«A mis fans.»

Agradecimientos

Gracias a:

El Dr. Pablo Padilla Longoria, tutor de esta tesis, con quien repasé todo el contenido aquí mostrado.

Por sus observaciones pertinentes al trabajo, a mis sinodales: Dr. Hugo Solís, Mtro. Patricio Calatayud, Dr. Roberto Morales, Dr. Leonardo Martínez y Mtra. Valeria Jonard.

Mis profesores de teoría musical del CMS, O. Chango, R. Rábago, AB Gómez.

Wilder Medrano, mi profesor de guitarra eléctrica.

Mi abuela Norma, que murió.

Mi mamá, por haberme comprado una guitarra.

Índice general

Agradecimientos III

1. Panorama de la música algorítmica **9**
1.1. ¿Qué son las matemáticas? 9
1.2. ¿Qué es un algoritmo? 19
1.2.1. ¿Qué es un algoritmo musical? 22
1.3. Panorama de la música algorítmica 29
1.3.1. ¿Qué es la música algorítmica? 30
1.3.2. Manifestaciones de la música algorímica 30
1.3.3. Matrices musicales 34
1.3.4. Juegos de dados 38
1.3.5. Cadenas de Markov 39
1.3.6. Escalas temporales del sonido 46
1.3.7. Iannis Xenakis y la música estocástica 50

2. Herramientas computacionales **63**
2.1. Notación musical 63
2.2. Combinatoria musical 69
2.2.1. ¿Qué son la combinatoria y la combinatoria musical? 69
2.2.2. Cadenas, collares y particiones 70
2.2.3. El Tonnetz 76
2.3. Síntesis de sonido 78
2.3.1. ¿Qué es la síntesis de sonido? 78
2.3.2. Osciladores 78
2.3.3. Flujos de señales (combinaciones de osciladores y efectos) . . . 81
2.3.4. Tipos de síntesis 82
2.4. Ritmos euclidianos y divisibilidad en la música 88
2.4.1. El algoritmo de Euclides 89
2.4.2. Versión musical de Tussaint del algoritmo de Euclides 90
2.4.3. Timelines 92

3. Desarrollo de obras musicales **93**
3.1. Pieza 1: "*Ir*" . . . 93
3.2. Pieza 2: "*Tenemos que hablar de Sara*" . . . 96
3.3. Pieza 3: "*60s & 90s*" . . . 98
3.4. Pieza 4: "*La mujer que quiso actuar como hombre*" . . . 100
3.4.1. Objetivo . . . 100
3.4.2. Marco teórico . . . 100
3.4.3. Planteamiento del problema . . . 105
3.4.4. Hipótesis . . . 107
3.4.5. Justificación . . . 107
3.4.6. Metodología . . . 108
3.4.7. Fuentes de investigación . . . 109
3.4.8. Realización . . . 109
3.4.9. Resultados experimentales . . . 110

4. Conclusiones **117**

Apéndice A. Solución al problema matemático **119**
A.1. Matrices de transición regulares . . . 119
A.2. Matrices doblemente estocásticas . . . 119

Bibliografía **126**

Motivación

Esta tesis tiene como uno de sus objetivos el de proponer una vía hacia el gusto y la comprensión de las matemáticas a través de la música.

Otro objetivo algo inverso, es acercar las matemáticas a la comunidad musical, pues ellas pueden facilitar el entendimiento de la teoría musical, el análisis de ciertas composiciones musicales o la creación de nuevas obras.

Más que considerar que directamente las matemáticas sean tecnología musical, opino que éstas yacen implícitas en las aplicaciones musicales, por lo que tener una buena noción sobre ellas conllevaría más ingeniosas conexiones con la música (sino es que sólo una mejor comprensión), fueran a manera de desarrollo tecnológigo o compositivo. Mi pregunta de investigación conlleva explorar las retroalimentaciones que se dan entre música y matemáticas, siendo la programación una posible vertiente.

El lenguaje coloquial sin uso de fórmulas es una manera eficiente de transmitir conceptos y conocimientos matemáticos. Este quehacer es implícito en la vida del matemático, pero la práctica de divulgación al mundo exterior está algo relegada. Sin embargo ésta tiene gran potencial, ya que la cultura matemática es aún sumamente desconocida.

A la presente tesis le acompañan: la creación de una obra musical propia, recursos en web tales como videos, tutoriales y entradas de blog que profundizan los ámbitos más prácticos de este proyecto. Algunos contenidos han quedado excluidos del trabajo para lograr un mayor enfoque en aspectos prioritarios.

Al momento de la creación de este proyecto se desconoce el alcance que éste puede tener, ya sea como obra artística o como herramienta educativa. Aún así planea sentar un método de trabajo, así como la exploración de ciertos contenidos particulares. Es un tema que podría desarrollarse aún más, durante el lapso de varios años venideros, aunque bajo agradables condiciones de trabajo. Se desea que una extensión sea posible, ya que la música implica la compra de material muy caro y sería una lástima tener que desecharlo.

Dos objetivos adicionales para la tesis son: la comprensión y la construcción a partir de elementos muy esenciales. Para lo segundo, debe haber una comprensión de lo primero: dilucidar cuáles son tales elementos básicos y cómo construir con ellos. Estos elementos pueden adquirir la forma de "herramientas teóricas", "código de programación", "ejecución de específicos fraseos en un instrumento", o "interconexión entre diferentes interfaces de producción de sonido". Se ha realizado una exploración en estos rubros y se han seleccionado fundamentos base para cada uno de ellos. Espero que este trabajo lo refleje a lo largo de sus páginas.

Un objetivo final para el proyecto es el de conseguir visibilidad respecto a la interpretación musical.

El material está disponible en el sitio: `ofeliayorquesta.com/tecmu`.

Introducción

Música algorítmica, según McLean y Dean ([McLean et al., 2018]), es un campo de estudio en el que, "con urgencia se explora y extiende el pensamiento musical mediante abstracciones formales. Y es en el proceso de crear música con algoritmos, que ésta queda representada a través de sistemas de notación formales, en el entendido de que la música prevalece como el orden más alto de interacción entre las ideas."

Este trabajo de tesis adoptará tal punto de vista en cuanto a lo que significa música algorítmica, pues vago como pudiera parecer, en realidad abarca un abanico de manifestaciones diversas de creación musical.[1]

Se asumirá además, que el lector sabe que una gran cantidad de lenguajes de programación musicales existen en la actualidad, los cuales contienen implementaciones exactas de algoritmos recurrentes para la música algorítmica. Más bien es el modo de trabajar con tales implementaciones sobre lo que se hará hincapié.

La presente investigación tiene por principal objetivo describir algunos métodos relevantes que he decidido emplear en la creación de una obra musical propia. La elección de éstos se basa en experiencias y reflexiones que he ido coleccionando a lo largo de más de diez años y que me han llevado a adoptar ciertos caminos por sobre otros. Conversaciones sucedidas con personajes clave del *live coding* (Alex McLean) y desarrolladores de software musical (Daniel Mayer) proveen veracidad, y varias de mis investigaciones centrales están basadas en sus aportes a la música algorítmica, los que están presentes a manera de artículos y *software* de código abierto. En particular, empleo *TidalCycles* y *Miscellaneus Lib*, paqueterías de estos programadores con las que yo creo porciones de mi música.

El *live coding*, la música algorítmica, los lenguajes de programación musical, la complejidad computacional de *SuperCollider*, las cadenas de Markov musicales, las ecuaciones diferenciales musicales, la teoría de gráficas aplicadas a la música, la notación musical, la música electroacústica, las cribas, los ritmos euclidianos y las matrices musicales, son temas cubiertos en esta disertación.

[1] La justificación se desarrolla en las próximas páginas.

Me interesa aún más destacar la existencia de una conexión plausible y contemporánea que se puede lograr entre la música y las matemáticas teóricas, visto esto desde el punto de vista de la creación musical. Lo que es relevante en el sentido de que hay que considerar un límite de tiempo para desarrollar la música, y con base en eso, se debe ir decidiendo cuáles herramientas habrían de usarse y cuáles quedan fuera. Pero aunque la metodología parezca en ocasiones muy técnica, una pregunta de investigación en mente es la siguiente:

> "*¿Cómo influyen las matemáticas como discurso artístico, en una obra electroacústica que emplea música algorítmica?*"

Un matemático que utiliza las matemáticas en el día a día, tratándose de un individuo consciente, reflexiona e intenta darle sentido a los símbolos, números, teoremas y problemas con los que se topa, o los relaciona con otros sucesos de su vida. La experiencia de la persona influye en el significado de estos teoremas y símbolos. De este modo, tal significado adquirido puede emplearse en otras áreas del conocimiento, como lo es la música. No se trata meramente de traducir fórmulas a un lenguaje de programación, aunque la posibilidad de que un planteamiento matemático exacto se programe y sea capaz de producir sonido, es una manera de establecer comunicación entre las disciplinas. El apartado "*¿Qué son las matemáticas?*" del Capítulo 1, profundiza sobre la aplicabilidad del pensamiento matemático en la música.

Los algoritmos no implican el eliminar o disminuir el proceso creativo del compositor. Más bien estos pueden facilitar la creación, sirviendo como método; proveyendo orden y coherencia, así como ayudando a construir la música de manera sistemática, desde lo simple a lo complicado. Los algoritmos tienen que ver con el cómo una obra está siendo creada y estructurada. La sección sobre matrices musicales y el panorama de la música algorítmica retomarán esto.

Ya que estamos situados en un momento de la historia en que un algoritmo resulta fácilmente ejecutable para el usuario no programador por medio de los llamados lenguajes de alto nivel (como pudieran ser *Python*, *SuperCollider* o *TidalCycles*), vale la pena preguntarse:

¿En qué me beneficiaría programar algoritmos desde cero, dado que tantos de ellos ya existen?

¿Es necesario entender la implementación de un algoritmo para poder comprender cómo funciona este, o para poder otorgarle un significado?

¿Sirve más la programación para entender la relación entre música y matemáticas, o de otro modo, qué parte de las matemáticas me ayudarían a componer y analizar mejor cierta música?

¿Cuál es el aporte que tiene el *live coding* a las ciencias de la computación, si es que lo tiene?

¿Cuál es el aporte que tiene el *live coding* en la música y en las matemáticas?

Ya que estamos hablando de una obra musical propia, también es importante dar a conocer la configuración espacial de la música y cómo será presentada a la audiencia:

Una guitarra eléctrica se conecta a un pedal de efectos (con delays, simuladores de amplificadores, distorsiones, reverberaciones, y efectos de modulación). El sonido que emana de la pedalera puede dirigirse hacia un pedal de *loop* y considerarse varias alternativas para la señal de salida; a saber, separando el sonido en dos señales o encadenarlas una tras otra. La señal resultante se envía, a través de una mezcladora o interfaz, a la computadora (o computadoras). Una vez ahí, la señal se procesa digitalmente usando programación musical o software comercial.

El audio algorítmico que emiten *TidalCycles* y *SuperCollider* es ruteado de manera estéreo o multicanal desde *SuperCollider* hacia una *DAW* o bocinas y se maneja conjuntamente a la interpretación de guitarra eléctrica (siempre y cuando la computadora lo permita).

Varios practicantes de la música algorítmica coinciden en que la música emitida por *TidalCycles* es "muy agradable al oído". Los patrones repetitivos, presentes en todas las culturas, son una primera razón de predilección. Y aunque se corre el riesgo de monotonía, la modificación en vivo de los algoritmos plantea la posibilidad de desarrollar la música e ir cambiando parámetros sonoros y sonidos individuales. Así pues, en *TidalCycles* es sencillo generar varias capas de ritmos complicados, algunos de ellos imposibles de ser ejecutados por percusionistas análogicos.

Al tocar guitarra, la intérprete efectúa un conteo mental de la música, sea esto llevado a cabo en el nivel de los compases, o en el de cada nota individual, o en de las frases. Se puede hacer, por tanto, una correspondencia entre el tempo de la música con el tempo de un reloj en el lenguaje de programación. Y a pesar de que el lenguaje con que piensa el músico la música sea sintácticamente diferente del de un lenguaje de programación, es a través de esta correspondencia, el que un ordenador y un músico pueden establecer comunicación.

Suele ocurrir en una pieza músical que una misma frase o variaciones de ésta, sean ejecutadas por los diferentes instrumentos presentes en una composición, por lo que, en un escenario de música algorítmica e instrumentación, resultaría útil traducir algunas frases desde una notación musical a una computacional o viceversa.[2] Basta saber que

[2] Contrariamente, Daniel Mayer, en *Algorithms in sound synthesis, processing, and composition: a dialectic game,* [Mayer, 2022], argumenta que la más eficiente exploración de algoritmos musicales para fines de síntesis sonora, se logra dándole toda la libertad a éstos, sin subordinarlos a una partitura.

esto es posible de lograrse. La más sencilla justificación de este hecho sería presentando una misma frase musical en sus dos notaciones (una escrita en pentagrama y la otra en su formato de código).

También veremos que, en un contexto de improvisación musical, funcionará elaborar un *score* guía para el músico, así como iniciar con arreglos algorítmicos ya preparados, sólo algunos de los cuales tendrán algo que ver con lo que tocará el músico. No toda la parte algorítmica deberá ser traducida a pentagrama, dado el carácter improvisatorio de la música. Además, en ocasiones esa tarea sería imposible, pues hay gestos musicales que no pueden transcribirse al formato de una partitura tradicional.

La estructura global de la tesis, por capítulos, es la siguiente:

El Capítulo 1 muestra un panorama de las matemáticas en la música, la música algorítmica y el *live coding*. Se profundizará en el significado e historia de cada uno de estos conceptos, primero independientemente, para luego poder relacionarlos entre sí. Nos interesa descubrir cuál es el rol de la música algorítmica y del *live coding* en la música, así como entender algunas diferencias esenciales que entre ellos hay. Finalmente, queremos proveer un fundamento para mi obra de música algorítmica.

Los Capítulos 2 y 3 cubren la metodología y el desarrollo de las piezas. Se profundiza en las herramientas computacionales y la modelación algorítmica.

Elegí desarrollar cuatro piezas; cada una de ellas fue compuesta usando un algoritmo base, situado desde una diferente perspectiva:

1. La primera pieza parte de un problema musical: ver los múltiplos como herramienta de conteo desde la ejecución de un instrumento y como una manera de complementarle con una orquestación algorítmica. La justificación de usar los "múltiplos" como ente aislado de un pentagrama, se debe a los ya implementados algoritmos de euclides, las cribas, y otras funciones más específicas de *TidalCycles* (tales como `whenmod` o `every`), que aplican este concepto para generar patrones rítmicos.

2. La segunda pieza busca partir de un problema matemático. Por lo que primero se plantea la teoría necesaria de los procesos de Markov y de un problema concreto, para luego traducirlo en música.

3. La tercera pieza tiene como objetivo partir desde un problema computacional. Inicia con la lectura del artículo "*PbindFX - an interface for sequencing effect graphs in the SuperCollider audio programming language*" (Daniel Mayer), a lo que le siguió realizar una experimentación sonora en *PBindFX* (un patrón de *SuperCollider* perteneciente a la biblioteca *Miscellaneous Lib*).

4. La cuarta pieza, "*La mujer que quiso actuar como hombre*", intenta describir un

escenario musical mediante la elaboración de correspondencias matemáticas y filosóficas entre los elementos clave de la música y algunos modelos matemáticos. La relevancia de este enfoque vira más hacia la conceptualización de una pieza musical de manera global, la posibilidad de plantear la trama de una pieza usando una fórmula matemática, y también aporta significados subjetivos.

El Capítulo 5 concluye el texto discutiendo los resultados obtenidos, tanto desde una perspectiva científica, como desde una perspectiva artística.

La tesis busca ser autocontenida; es decir, presenta la información y definiciones necesarias para construir un escrito en el que se van desenvolviendo las tramas y resultados, y se reflexione acerca del aporte que el trabajo consigue. Cada capítulo sugiere una bibliografía adicional que puede servir para acompañar al texto y profundizar en los contenidos.

1 Panorama de la música algorítmica

Esta primera sección del primer capítulo comienza por desarrollar los temas: "*¿Qué son las matemáticas?*" y "*¿Qué es un algoritmo?*". A veces se suele creer que las matemáticas son algoritmos. Yo diría que las primeras contienen a los segundos y que las matemáticas son más que eso. Son una ciencia creada por personas, que se puede aprender de diferentes maneras y cubre muchas especializaciones. La sección concluye con "*¿Qué es un algoritmo musical?*", y con eso tenemos ya material para reflexionar mientras leamos una segunda parte del capítulo, que se enfocará totalmente a la música. La bibliografía consultada para esta primera parte es [MacLane, 2012], [Frenkel, 2014], [Haack, 1991], [Corona and de los Ángeles Ancona, 2011], [McLean et al., 2018], [Petzold, 2008], [Burkholder et al., 2019], [Boolos et al., 2002], [Ashton, 2003] y [Parviainen, 2017].

1.1. ¿Qué son las matemáticas?

Una definición popular para las matemáticas es *"la ciencia de los patrones"* (Devlin, 1998). También, las matemáticas son un *lenguaje abstracto* que facilita el estudio de ciertos fenómenos de la naturaleza y del pensamiento (que también es naturaleza). Es por ejemplo, el lenguaje de la física.

Mi labor de divulgación de las matemáticas consiste en convencer al lector de que éstas son mucho más que números o algoritmos. Una visión demasiado restringida de lo que son, sólo fomenta desinterés. Sería mejor acercarnos a ellas con una mente más abierta, sin presuponer que ya sabemos cómo son y de qué tratan en toda ocasión. El lector debe a su vez creerse capaz de descubrirles algo nuevo, e incluso de que puede siempre investigarlas a su propio ritmo y nivel para algún fin necesario.

Aunque "seamos malos" en matemáticas, mantener una curiosidad permitirá retener al menos una cosa pequeña: una fórmula, diagrama, notación o idea, y sobre ello la mente podrá libremente reflexionar.

Vivimos en un mundo gobernado por los algoritmos de redes sociales, dispositivos móviles, e incluso de publicidad política. Con más razón, una inspección al cómo funcionan estos mecanismos nos ayudaría a comprender parte importante del mundo en que vivimos; ayudarnos a tomar decisiones.

Argumentaría que hay fenómenos matemáticos que pueden estudiarse y entenderse desde el ámbito de la divulgación y las palabras (leyendo por ejemplo, un libro de Steven Strogatz), pero que por otro lado está la investigación formal. Una perspectiva considerablemente más difícil y que como tal, involucra la inmersión en la resolución de problemas y en un nuevo lenguaje. Hay quien debatiría que quitar estos dos elementos de la ecuación implicaría no reconocer la importancia que las matemáticas tienen. Yo no estoy 100 % de acuerdo, pues el aprendizaje activo no es la única ni la mejor manera de adquirir conocimientos y no necesariamente lleva a la resolución de un problema. Conocer lo que en matemáticas se ha hecho no implica tener que recorrer a pata andada las vidas de todos los matemáticos habidos en la historia de la humanidad, tal como leer un libro no significa reescribir uno mismo el mismo libro. Eso sí, siempre rechazar resolver un problema significa ignorar algo fundamental. Sería imposible que sin eso, alguien realmente llegara a comprender el pensamiento matemático.

¿Qué son las matemáticas? No sólo está en mí responder esa pregunta. Es de cada quién averiguarlo. Sin embargo, Saunders MacLane (1909 - 2005), prolífico matemático estadounidense, en [MacLane, 2012],[1] en correspondencia con algunas de mis palabras, menciona:

> *"Para poder lidiar (de manera efectiva) con la pregunta de cuál es la función y forma de las matemáticas, primero hay que observar lo que en la realidad son éstas. Una filosofía de las matemáticas no convencerá, a menos que esté fundada en una examinación de las matemáticas mismas."*

En las matemáticas se hacen explícitas desde las más mínimas estructuras de pensamiento hasta las más grandes y se provee una metodología basada en axiomas, proposiciones y teoremas que se conectan mediante la lógica matemática, la cual permite verificar si cierta proposición es verdadera o falsa; y a partir de suponer válidos una lista de fundamentos (axiomas) deben poderse demostrar (casi) todos los demás teoremas de una teoría matemática. Debemos quedar seguros de que tales demostraciones son correctas, pues de otro modo la teoría se cae.

El quehacer del matemático es garantizar que sus proposiciones sean sólidas para que puedan luego aplicarse dentro de la misma teoría y fuera de ella.

Las matemáticas contemporáneas se dividen en áreas, donde cada una se enfoca a la resolución de ciertos tipos de problemas, usando las metodologías en esas áreas, aunque tales áreas van conectándose entre sí. Álgebra y geometría convergen en la

[1]"*Mathematics, Form and Function*", *Springer*, 2012

geometría analítica; cálculo y topología ambas estudian continuidad; topología y teoría de gráficas estudian nudos; teoría de gráficas y programación comparten el diseño de algoritmos; álgebra lineal y ecuaciones diferenciales trabajan conjuntamente en soluciones por métodos numéricos de funciones multivariables, o la teoría de la probabilidad encuentra aplicaciones en áreas de la física tales como fenómenos colectivos, física estadística, o cuántica.

Ahondaré en algunas conexiones más específicas que sean de interés para esta tesis, y que no serán por supuesto, exhaustivas. Por ejemplo, el diseño de algoritmos, la probabilidad o los sistemas dinámicos son de interés recurrente para la composición algorítmica y la programación musical.

Para responderse la pregunta de qué son las matemáticas, MacLane propone indagar sobre cinco preguntas, que son: 1. ¿Cuál es el origen de las matemáticas? 2. ¿Cómo se organizan las matemáticas? 3. ¿Son los formalismos de las matemáticas basados en, o derivados de los hechos? 4. ¿Cómo sucede el desarrollo de las matemáticas? 5. ¿Cómo uno evalúa la profundidad e importancia de la investigación matemática?

Es mi opinión, que para una efectiva transdisciplinareidad entre matemáticas y cualquier otra cosa, debe enfatizarse constantemente cuáles son los elementos y asociaciones básicas sobre los que se pretende dar un enfoque interdisciplinario; de otro modo o no se llega a nada o suele dársele más importancia solamente a una de las partes. Un ejemplo típico es el de un doctor en composición musical que acepte lo que una matemática tiene que decir sólo si ella cursa la carrera en música, -desde la licenciatura hasta el doctorado-. Y viceversa: que la matemática crea que el músico no entiende nada de matemáticas avanzadas.

Existen estudios que sugieren que las habilidades cognitivas que un músico utiliza, son las mismas que las de un matemático (un caso a favor es el "*Efecto Mozart*"); de tal modo que, por ejemplo, se ha encontrado que tocar un instrumento musical es beneficioso para sacar buenas calificaciones en matemáticas. Esto a la vez podría sugerir que, aunque dos personas hayan pasado por formaciones diferentes, pueden, si hay disposición, entenderse en niveles muy abstractos, a pesar de que los lenguajes parezcan totalmente disímiles.[2]

Es parte importante de esta tesis manifestar que, a pesar de las similitudes que música y matemáticas puedan tener, no son una misma disciplina, ni sustituibles una por otra. Las matemáticas estudian problemas abstractos, algunos de los cuáles, por su nivel de dificultad, serían difíciles de ser solucionados por músicos (a menos que esos músicos también fueran matemáticos), y sin embargo, tales soluciones podrían ser de interés para ellos. Algunos problemas matemáticos, por su nivel de generalidad,

[2]Ver: *Análisis del Efecto Mozart en el desarrollo intelectual de las personas adultas y niños*, [Morales et al., 2011].

abarcan muchos más casos que los de aplicación musical. Mientras un matemático teórico se preocuparía por extender su paso en las diversas ramas de las matemáticas o sub-áreas de su área, así como por dedicarle días y noches a la resolución de un sólo problema abierto para luego publicar un artículo, el músico que viviera en el mismo condominio preferiría mientras tanto, ampliar su repertorio musical, tocar un instrumento, manejar la notación musical al vuelo y componer sobre los varios sistemas compositivos. Así escrito, en ambos casos estoy hablando de gente sobresaliente, obviando muchas de las dificultades más locales.

Analicemos un fenómeno como el caos. Verbalmente se le puede comprender y directamente relacionarlo con todo tipo de experiencias: "como el desastre que es mi vida" o con "una pieza donde múltiples instrumentos suenan simultáneamente sin que parezca haber un acuerdo entre ellos." Un individuo que escucha esta última pieza podría asociarlo con lo primero: "así suena mi vida hoy en día." E independientemente, el caos también tiene una interpretación rigurosa en física. La pregunta es: ¿De alguna manera ese rigor puede hacerse coincidir con la interpretación cotidiana? Una respuesta rápida sería exhibir ejemplos donde eso ocurra, por ejemplo las obras de Ligeti, Varèse o Xenakis. Aunque una observación también pertinente es la de que, al referirnos al "caos de una vida" estamos implícitamente usando la intuición para "estar seguros" de que ese fenómeno realmente significa caos. La intuición juega un papel importante en matemáticas. Es cuando una situación se ve medio borrosa, sin embargo algo nos indica el camino tal vez correcto: es una corazonada. Así pues, no hay que ser matemático para intuir "por dónde van las matemáticas". El sentido común elabora asociaciones entre el pensamiento y los objetos de estudio (o la forma de proceder) de las matemáticas. Por ejemplo, la noción de espacio, figura geométrica, número, abstracción, algoritmo, resolución de problemas o movimiento, sugieren pertenecer a ellas.

MacLane ha proporcionado una tabla "no dogmática", que relaciona las ideas intuitivas y actividades que son propias de las matemáticas (y que todos podemos por sentido común entender) con sus formalizaciones en el mundo de las matemáticas de actualidad (ver Tabla 1.1).

Precisamente, a partir de las ideas intuitivas es plausible trazar puentes entre disciplinas distintas, ya que desde la experiencia personal podemos encontrar posibles relaciones entre estas nociones y nuestro campo de estudio.

Un músico puede relacionar el conteo con varias cosas: la lectura a primera vista, o el tocar bajo el metrónomo. La enumeración con la cantidad de notas que toca en cada tiempo. O con las distintas secciones que conforman una pieza. La recolección podría asociarla con las escalas y motivos que va a tocar. Los reacomodos podría asociarlos con los modos jonio, dorio, frigio, lidio, mixolidio, eolio y locrio de una escala. También podría ver reacomodos de acordes (por triadas): tónica, primera inversión o segunda

Actividad	**Idea**	**Formulación**
Recolectar	Colección	Conjunto (de elementos)
Contar	Siguiente	Sucesor; orden; número cardinal
Comparar	Enumerar	Biyección; número cardinal
Calcular	Combinación (de números)	Reglas de adición. Reglas de multiplicación. Grupo abeliano.
Reacomodar	Permutación	Biyección. Grupo de permutación.
Manejo del tiempo	El antes y el después	Orden lineal
Observar	Simetría	Grupo de transformaciones
Construir; dar forma	Figura; simetría	Colección de puntos
Medir	Distancia; alcance	Espacio métrico
Movimiento	Cambio	Movimiento rígido. Grupo de transformaciones. Tasa de cambio.
Estimar	Aproximación Cercanía	Continuidad; límite. Espacio topológico.
Seleccionar	Porción	Subconjunto. Álgebra Booleana
Argumentar	Demostrar	Conectivos lógicos
Elegir	Aleatoreidad	Probabilidad (Favorable / Total)
Acciones Sucesivas	Seguido de	Composición. Grupo de transformaciones

Tabla 1.1: Ideas intuitivas y actividades que están presentes en las matemáticas, según Saunders MacLane.

inversión. Sería capaz de ver simetrías en el pentagrama. Asociar la métrica con los tipos diferentes de compases. La selección y la elección con las notas de la escala o acordes que toca en una improvisación. O indicar por manejo del tiempo cuánto le toma practicar una pieza en específico. Podría relacionar la argumentación con la enseñanza constructiva y razonada de la teoría musical. Y las acciones sucesivas con la necesidad de componer una pieza usando ciertas melodías estudiadas. Podría definir muchísimas cosas muy diferentes dependiendo del contexto.

A la tabla anterior yo agregaría, previo a "acciones sucesivas", la actividad de *transformar*, –o quizá eso refiriera la idea–, y la formulación sería la de *función*.

En matemáticas, una función es una regla de asociación entre dos conjuntos A y B llamados respectivamente, dominio y contradominio. Podemos imaginarlo como una máquina que recibe una entrada y devuelve una salida, a partir de usar y modificar la información de entrada.

McLane está refiriendo de manera implícita este criterio, al usar términos específicos que detallan propiedades más interesantes de funciones, como el suceder, biyectar, sumar, o multiplicar. También, la composición no es más que aplicar varias funciones seguidas. Pero tal vez MacLane pecó de obviar un concepto que es fundamental entender para poder poder definir un montón de abstracciones matemáticas diferentes.

También, añadiría a la lista la acción de *simbolizar* y la relacionaría con la *notación matemática*, conformada por números, letras y todo tipo de abreviaturas que simplifican objetos e ideas. ¿No es sorprendente que el símbolo $\mathbb{R}$ refiera una entera estructura numérica infinita y no numerable? ¿O que sea posible nombrar mediante una función, a saber, $|B|^{|A|}$, al conjunto de todas las funciones real-valuadas de A en B?

A su vez, como actividad se me ocurriría agregar la *simplificación*, la cual puede tener que ver con simbolizar, pero también incluiría otras necesidades. Encontrar la fórmula correcta en un problema, significa, de cierta manera, que sólo hay una respuesta tal; la más simple. En un espectro más amplio del quehacer matemático, la optimización numérica también busca desarrollar los algoritmos más rápidos y eficientes, aunque los resultados no son únicos necesariamente.

La génesis de las estructuras matemáticas más complicadas, dice MacLane, tiene lugar en el ámbito meramente matemático, donde existen una variedad de procesos capaces de generar nuevas ideas y nociones. Algunos de ellos son:

1. **Acertijos (resolución de problemas).** Por ejemplo, Fermat, sin dar una demostración, afirmó que las ecuaciones $x^n + y^n = z^n$ para $n > 2$ no tienen solución en los números enteros. Pero esta aparentemente inocente ecuación diofantina provocó el desarrollo de la teoría algebraica de números en el Siglo

XIX.[3]

2. **Completación.** Por ejemplo, la no existencia de números negativos en la estructura de los números naturales requirió construir una nueva estructura: la de los números enteros. Pero en los enteros no siempre se puede hacer división, lo que dio lugar a los números racionales. Sin embargo, algunos números no son racionales. Esto sugiere la construcción de los números reales. Pero en los números reales no existe la raíz cuadrada de números negativos y de ahí aparecieron los números complejos. Éstos últimos sí son completos, pues todas las ecuaciones algebraicas tienen solución en este sistema numérico, lo que se tuvo que demostrar en un teorema llamado el teorema fundamental del álgebra.[4]

3. **Invarianza.** Un ejemplo es la definición contemporánea de geometría, dada por Félix Klein (en 1872): "es el estudio de las propiedades geométricas que permanecen invariantes bajo un grupo de transformaciones."[5]

4. **Estructutra común (analogía).** Por ejemplo, Bach, en sus cánones, usó todos los distintos tipos de simetrías, lo que sugiere un implícito uso de la estructura de grupo, en el sentido matemático.[6]

5. **Estructura intrínseca.** Es cuando una estructura "oculta" parece dar explicación a cierta situación. Por ejemplo, Gell-Mann y Zweig predijeron la existencia de los *quarks* (estos son los componentes de los neutrones y protones, que conforman el núcleo de los átomos), después de que intentaran aplicar en física la teoría de representaciones del grupo $SU(3)$. Específicamente, el hecho de que $SU(3)$ tiene dos diferentes representaciones tridimensionales.[7]

6. **Generalización.** ¿Cuántos ritmos de 8 pulsos se pueden crear usando únicamente sílabas largas y cortas? Esta pregunta data del año 500 a.C y se puede generalizar matemáticamente de la siguiente manera: ¿De cuántas maneras puede escribirse un número n como la suma de términos que sólo pueden ser 1 o 2? Hay 34 maneras, hecho demostrable con la sucesión de Fibonacci: $0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, \ldots$ La segunda pregunta responde la primera pregunta, pero también responde la más general pregunta de encontrar cuántos ritmos binarios caben en cualquier longitud n de pulsos.[8]

[3] *Mathematics form and function*, en [MacLane, 2012].

[4] En *Álgebra Superior*, en [Bravo et al., 2006]; *Mathematics form and function*, en [MacLane, 2012], capítulo 2; *Galois Theory*, en [Stewart, 2022].

[5] *Introducción a la geometría avanzada*, en [Ramírez-Galarza and Seade-Kuri, 2002].

[6] *Music, a mathematical offering*, en [Benson, 2006], capítulo 9; y *Symmetry-canon: Music and mathematics, painting and graphicization perpetuum mobile*, en [Arlandi, 1996].

[7] *Love and math: The heart of hidden reality*, en [Frenkel, 2013], página 11.

[8] Bhargava, Manjul, en Fields Medal Symposium.(s.f) Videos [Fields Institute]. Youtube. Recuperado el 26 de noviembre de 2016. de `https://youtu.be/LP253wHIoO8?si=i8kWpNNfqUmr1wCD`

La generalización adopta diferentes facetas en matemáticas. Hay generalizaciones que parten de varios casos concretos y se busca encontrar una ley general que cubra todas ellas. Casos más sutiles de generalización son aquéllos en que se puede dar cierta generalización pero algunas propiedades se pierden, por ejemplo: los cardinales finitos sugieren la existencia de los cardinales infinitos, pero esas leyes finitas no se preservan. Similarmente, los grupos finitos son generalizables a grupos infinitos, pero los teoremas de orden en los subgrupos ya no se mantienen.[9]

7. **Abstracción.** La abstracción, según MacLane, considera una estructura a partir de negar la proveniencia de sus elementos, pero manteniendo "todas" las operaciones sobre estos elementos.

   Un *modelo matemático*, por ejemplo, se puede definir como: "una descripción abstracta, en lenguaje y concepto matemático, de un sistema concreto."

   Una definición alternativa de *modelo matemático* es: "el proceso de crear la representación matemática de un escenario real para fines de elaborar alguna predicción u obtener mayor comprensión."

8. **Axiomatización.** Este proceso, según MacLane, típicamente pregunta: dada una lista larga de "todos" los teoremas en un tema dado, ¿puede uno deducirlos a partir de una más pequeña lista? Tal lista constituirá los axiomas para ese tema.[10]

9. **Análisis de una demostración.** Una manera de encontrar axiomas es preguntarse por una lista mínima de las propiedades necesarias para llevar a cabo una o varias demostraciones. Por ejemplo, los axiomas de un anillo conmutativo son esencialmente, la mínima lista requerida para realizar las manipulaciones algebraicas estándares de adición, resta y multiplicación. Nuevas ideas matemáticas pueden muchas veces ver luz al analizar qué es lo que hace que una demostración funcione.[11]

En varios de los incisos anteriores mencioné ya el concepto de grupo, para el que su estudio formal tiene lugar en la materia de álgebra moderna, en el quinto semestre de la carrera de matemáticas de la UNAM. Sin embargo, la noción de grupo es tan intuitiva que resulta sorprendente por qué los cursos de matemáticas de la secundaria ni siquiera lo mencionan.

La manera de enseñar las matemáticas a los niños influye en la percepción global que se tiene de ellas.

[9] *Mathematics form and function*, en [MacLane, 2012].
[10] *Mathematics form and function*, en [MacLane, 2012].
[11] *Mathematics form and function*, en [MacLane, 2012].

Mundialmente, matemáticos de la actualidad critican los planes de estudios de matemáticas en niveles pre-universitarios, ya que éstas suelen enseñar sólo algoritmos numéricos. Sumas, restas, multiplicaciones, divisiones y raíces cuadradas de números y polinomios, factorizaciones de polinomios, trigonometría, soluciones de sistemas de ecuaciones de primer y segundo grado, límites, derivadas e integrales de funciones. Pero ¿quién o quiénes inventaron todo eso? ¿Por qué en clase de historia sí valen los nombres de los emperadores, revolucionarios, aquéllos personajes, lugares y armamentos de las guerras mundiales, mientras que al ciudadano promedio le preguntas por Hilbert, Von Neumann o Grothendieck y no tiene idea de qué le hablas? Por ello el común denominador de la gente sigue creyendo que las matemáticas son sólo algoritmos, y que no hay una cultura detrás. Falta una visión más femenina de las matemáticas, algunos dirían.

El matemático y divulgador Edward Frenkel argumenta:

> *"Imagina que tuvieras que tomar una clase de arte en la que te enseñaran cómo pintar una reja o pared pero nunca te mostraran las pinturas de los grandes maestros y ni siquiera te dijeran que tales pinturas existen. Pronto te estarías preguntando, ¿por qué estudiar arte?"*[12]

Tal vez, el plan de estudios de matemáticas en niveles pre-universitarios es para bien, ya que así relacionamos a las matemáticas con la visión más algorítmica y difícil del pensamiento. De otro modo, esta desaparecería y tal vez se generaría un oscurantismo que sólo algunas pocas personas podrían dilucidar. Pero es cierto que tanto en la música como en los rompecabezas (por ejemplo, los cubos de Rubik), que son actividades con las que desde pequeños interactuamos, está implícita la noción de grupo, y podría ser un incentivo reconocerlo por anticipado.

Respecto al tema "música y matemáticas" es similar. Algunos conceptos avanzados de matemáticas se presentan quizá más alcanzables, a través de preguntas musicales.

Es posible disfrutar la música a partir de sólo escucharla sin necesidad de ejecutarla. De esta manera, es posible reconocer, a través de piezas musicales, estructuras no triviales.

Recurramos como ejemplo a la pieza "*Clapping music*" de Steve Reich, citando algunos párrafos del artículo "*Clapping Music - A combinatorial problem*" ([Haack, 1991]):

> "Si primero limitamos nuestra atención a los posibles acomodos de 8 aplausos y 4 silencios en una medida de doce pulsos, encontraremos, por supuesto, $\binom{12}{8} = 495$ posibilidades. Sin embargo, en una composición como ésta, es razonable imponer algunas restricciones sobre los patrones que

[12] *How our 1,000-year-old math curriculum cheats America's kids*, en [Frenkel, 2014].

> serán permitidos. No todos los 495 posibles patrones son igualmente estéticos o del mismo interés intelectual. Por ejemplo, Reich elige comenzar la pieza con un aplauso, ¡misma cosa que nos permite saber que una pieza ha iniciado! También, Reich ha decidido no permitir dos silencios consecutivos. A partir de estas dos restricciones podemos ahora contar todos los posibles acomodos, considerando dónde sí podemos ubicar las combinaciones de 8 aplausos y 4 silencios, llegando a $\binom{8}{4} = 70$ posibilidades.
>
> La estructura de *Clapping Music* nos lleva a imponer una relación de equivalencia sobre estos 70 patrones. Si pensamos en *Clapping Music* como una pieza sin principio, capaz de ciclarse infinitamente, podríamos ver como equivalentes a cualesquiera dos patrones que fueran capaces de producir la misma pieza. En otras palabras, dos patrones son equivalentes si son permutaciones cíclicas el uno del otro. Para contar el número de piezas que podrían crearse mediante la permutación cíclica de un patrón de 8 aplausos y 4 silencios (sin silencios consecutivos), podemos aplicar el lema de Burnside:
>
> *Sea G un grupo finito de permutaciones de orden n en un conjunto S de patrones. Para cada ϕ en G, sea $n(\phi)$ el número de patrones en S fijados por ϕ. Entonces, el número N de patrones inequivalentes es*
>
> $$N = \frac{1}{n} \sum_{G} n(\phi).$$
>
> [...]
>
> Luego de escuchar la grabación de esta pieza, los estudiantes también disfrutan de analizarla. Encuentran que el atractivo estético y la estimulación intelectual en "*Clapping Music*" puede en parte, explicarse por tres factores: la complejidad del patrón inducido por 8 aplausos y 4 silencios; las variaciones que resultan de la simple aplicación de una permutación cíclica y la síncopa que provee el patrón $3, 2, 1, 2$ usado por Reich. ¡Los estudiantes encuentran estas consideraciones mucho más interesantes que contar collares $m-$coloreados! Usar en las clases de matemáticas materiales provenientes de las humanidades puede resultar invaluable para mantener su interés."[13]

Pueda parecer intrincado incluir estos párrafos en una sección que trata sobre el "¿qué son las matemáticas?", sin embargo ello presenta un ejemplo de lo que la investigación en matemáticas teóricas llega a lograr dilucidar respecto a algunas de nuestras percepciones cotidianas (en este caso, siendo escuchas musicales), y cuyas soluciones formales pueden ser nada fáciles. En efecto, esas percepciones son de rigor y requieren

[13] En *Clapping music, a combinatorial problem*, en [Haack, 1991].

de la profesión del matemático. El oído está sirviendo como un “visualizador” cognitivo de patrones en los niveles local y global, y podemos decidir si jugar con algunos de ellos.

En resumen. ¿Qué son las matemáticas? En este apartado presenté un par de definiciones para ellas; proporcioné un panorama sobre el quehacer del matemático hoy en día, las varias áreas de las matemáticas, y brindé una lista de procesos que en las matemáticas suelen llevar a nuevas conjeturas o descubrimientos. Además di una tabla de actividades mentales que todos los humanos usamos, y con las que podemos asociar específicos conceptos matemáticos. Brindé unos ejemplos músico-matemáticos que relacionan, valga la redundancia, a la música y las matemáticas.

Finalmente, solicité al lector considerar ampliar su visión de las matemáticas tomando en cuenta los prejuicios sociales que se tienen de ellas. Diversos autores, como MacLane, Courant, o Frenkel han intentado promover las matemáticas a una amplia comunidad, y afortunadamente, el número de divulgadores de la ciencia está creciendo.

1.2. ¿Qué es un algoritmo?

Corona y Ancona, en [Corona and de los Ángeles Ancona, 2011], definen algoritmo como una secuencia ordenada y finita de pasos que permiten resolver un problema o tarea específica:[14]

> **Definición:** *Un algoritmo finito es una secuencia finita y ordenada de pasos que permiten resolver un problema o tarea específica.*

Aseveran ellas (Corona y Ancona) que todo algoritmo debe ser, además de finito y ordenado, *definido* y *general*. Con lo primero se refieren a que no puede haber dobles interpretaciones para un mismo algoritmo; el resultado será único para cada una de las entradas. Con general, se enfatiza que un algoritmo debe tolerar cambios que se puedan presentar en la definición de un problema.

En particular, la posibilidad de permitir adaptaciones, vuelve propenso el considerar que un algoritmo pueda ser infinito, —en una situación que sugiriera algo de ese calibre—. En matemáticas, una manera de definir nuevos objetos es a partir de tomar cierto objeto matemático y modificar su definición tal que se cree otro objeto matemático.

Propongamos por ejemplo, la siguiente definición:

[14] *Diseño de algoritmos y su codificación en lenguaje C*, p. 2.

Definición: *Un algoritmo infinito es un algoritmo que no es finito.*

En términos de lógica matemática, sea A la proposición

A: Un algoritmo es finito.

Según la definición, un algoritmo infinito significa negar la proposición A.

$\neg A$: *Un algoritmo no es finito.*

Negar que un algoritmo sea finito consiste en negar la definición de algoritmo finito:

$\neg A$: *Un algoritmo infinito no es una secuencia finita y ordenada de pasos que resuelven un problema.*

En general, para negar una conjunción de propiedades, basta que una de ellas no se cumpla. Habrían varios casos, según su tabla de verdad. Un algoritmo infinito sería un algoritmo que:

- NO es una secuencia finita de pasos, SÍ es ordenada, y SÍ resuelve un problema.
- SÍ es una secuencia finita de pasos, NO es ordenada, y SÍ resuelve un problema.
- SÍ es una secuencia finita de pasos, SÍ es ordenada, y NO resuelve un problema.
- NO es una secuencia finita de pasos, NO es ordenada, y SÍ resuelve un problema.
- NO es una secuencia finita de pasos, SÍ es ordenada, y NO resuelve un problema.
- SÍ es una secuencia finita de pasos, NO es ordenada, y NO resuelve un problema.
- NO es una secuencia finita de pasos, NO es ordenada, y NO resuelve un problema.

Pero estas opciones merecerían una discusión que conviniera qué cosas tienen sentido. Poco lo tendría, y más bien absurdo sería permitir que un algoritmo infinito sí pudiera ser una secuencia finita de pasos. Lo que se busca es que ésta sea infinita.

Ya sin tanto formalismo de por medio, todo podría quedar así:

Definición: *Un algoritmo infinito es una sucesión infinita de pasos ordenados o no, que resuelven una tarea específica.*

Quito el caso de que un algoritmo no resuelva un problema porque también la esencia de la algoritmia recae en que haya un resultado de salida.

Luego, un caso particular de nuestra nueva definición sería:

Una sucesión infinita de algoritmos finitos es un algoritmo infinito.[15]

Con esto, estamos mostrando un posible método para construir algoritmos infinitos. Aunque ello no quiere decir que todos los algoritmos infinitos deban ser una sucesión infinita de finitos. De hecho, una sucesión infinita de finitos sería numerable y podría intuirse la existencia de algún algoritmo no numerable (que por lo tanto, también sería infinito).[16] Esta noción de algoritmo infinito es en principio, independiente de que tenga o no una implementación computacional. En otras palabras, está basada en la manera de imaginar infinitos matemáticos. Galileo ya intuía que el infinito es intrínseco al movimiento de los cuerpos que caen, lo que puede corroborarse en su libro "*Diálogos sobre los dos sistemas máximos del mundo*" (ver [Galilei, 2010]). Es difícil establecer que se aseverara un algoritmo infinito convergente al suelo. La noción de infinitud es difícil de imaginar.

Aunque el término algoritmo proviene de las ciencias de la computación, la utilización de éstos proviene desde mucho tiempo antes; desde las primeras civilizaciones organizadas. Un método para cazar un animal usando un hacha, o un grito en señal de alerta puede ser considerado un algoritmo. Aunque si relacionamos al algoritmo con un pensamiento abstracto que emplea artimética, podemos pensar que los árabes, hindús, chinos, griegos o mayas fueron quienes los inventaron, ya que de manera separada, estas civilizaciones manejaban sus propios sistemas numéricos. Es coloquial que actualmente, los humanos designemos por algoritmos a varias de nuestras actividades diarias: preparar un platillo, establecer horarios para despertar, comer y dormir, regar las plantas, sacar a pasear al perro, lavar los trastes seguido de lavar los dientes seguido de ir a la escuela, etc. En todos estos casos también es aplicable la definición de algoritmo (sucesión finita y ordenada de pasos que resuelven una tarea específica).

En el ámbito de la música algorítmica es útil considerar distintos niveles de precisión para los algoritmos, siendo los matemáticos y los computacionales los más estrictos. En particular, para la computación suele importar mucho definir a los algoritmos como finitos, ya que sólo así se puede calcular su tiempo de ejecución. Un ejemplo de precisión sutilmente menor lo da el canal de *YouTube* "*kamal yassin*", donde se muestra la representación de los diferentes algoritmos de ordenamiento provenientes de las ciencias de la computación (*bubble sort*, *quick sort*, etc) a manera de coreografías donde varios bailarines ejecutan danzas tradicionales húngaras.

Es decir, aunque se trata de una exacta representación de un algoritmo computacional, hay muchos grados de libertad en estas obras, como lo son el vestuario, el tipo

[15]Esta idea, más aún puede ser apoyada al observar el cómo se definen las gráficas infinitas de la teoría de gráficas o las funciones generadoras de la combinatoria.

[16]La mera existencia de un algoritmo no numerable, es decir un algoritmo no computable mediante una máquina de Turing finita, sugiere considerar a las máquinas de Turing infinitas. Ver *The annotated Turing*, en [Petzold, 2008], y *Computability and logic*, en [Boolos et al., 2002].

de danza, o quienes participan en el ensamble. En los diferentes niveles de la pieza también se pueden identificar algoritmos que determinan los movimientos de los bailarines: cómo realizan ellos pasos hacia enfrente, hacia los lados, cómo giran, alzan las manos, o cómo tapean con los pies. Y estas maneras caracterizan al tipo de danza.[17] Los algoritmos de baile son totalmente diferentes al algoritmo de ordenamiento que determina tanto las posiciones de los bailarines en el escenario como la trama de la danza, y en ningún momento éstos se conflictúan, ya que están actuando en diferentes niveles.

También, en el ámbito de la música algorítmica servirá considerar separadamente de la teoría de computabilidad y según sea el contexto, a los conceptos de finitud e infinitud de los procesos sonoros.[18]

Conviene notar que, en el diseño de algoritmos por programación estructurada, los tipos de algoritmos más comunes son:

- Algoritmos de búsqueda y ordenamiento.
- Algoritmos de programación dinámica.
- Algoritmos recursivos.
- Algoritmos codiciosos.
- Algoritmos de cadena.

Éstos pueden llegar a utilizarse para producir música y sonido. Efectivamente, juegan un rol fundamental en la implementación de patrones musicales, métodos de síntesis y la arquitectura del servidor de *SuperCollider*.

1.2.1. ¿Qué es un algoritmo musical?

Mi propia definición de algoritmo musical sería:

> *Un algoritmo musical es un algoritmo que participa en la producción de música o sonido.*

La globalidad de la definición permite abarcar diversos contextos, aunque tipificar los algoritmos musicales no resulta tan sencillo. Para una clasificación acertada sería deseable primero estudiar la música algorítmica y ver el amplio panorama.

En primera instancia, habría varias maneras útiles de clasificar algoritmos musicales. Una de ellas sería por jerarquías. El ejemplo de los bailes húngaros muestra al menos

[17] Por ejemplo, elegir una danza irlandesa, en vez de una húngara, provocaría otro resultado visual.
[18] Más sobre esto en la sección "*¿Qué es un algoritmo musical?*"

dos jerarquías posibles para definir algoritmos en un mismo acto dancístico.

Las escalas temporales del sonido de Curtis Roads (*Microsound*, 2001) muestran 9 jerarquías diferentes donde pudieran ocurrir algoritmos. Argumentaría yo que las más frecuentes, y por ende, más importantes para la creación musical son sólo 3: microsonido, mesosonido y macrosonido. Sin embargo, el total de las 9 escalas abarcan más que la creación musical; incluyen también el desarrollo de software, los métodos matemáticos, construcción de instrumentos musicales y experimentación sonora, entre otros.[19]

Otra manera de clasificar algoritmos musicales podría ser según su función y comportamiento. Algunos algoritmos estarán más arraigados a fórmulas matemáticas, como pudiera ser un proceso de síntesis, pero otros algoritmos servirán como una interfaz para hacer interactuar sistemas distintos, por ejemplo, audio y visuales, danza y *live coding*, o guitarra y *live coding*. También, el uso de algoritmos en la música podría ser parcial; con sólo porciones de una obra artística siendo dictadas por algoritmos. O total; donde los algoritmos estuvieran presentes en cada dimensión sujeta a creatividad musical.[20] Podría haber también una dialéctica entre algoritmos muy precisos y algoritmos muy imprecisos.

Veamos otras dos clasificaciones para algoritmos musicales, presentes en la literatura.

Levtov, en [McLean et al., 2018],[21] categoriza los tipos de algoritmos musicales por:

- **Generativos.** Son ejecutados sin que el usuario proporcione una entrada para el algoritmo.
- **Reactivos.** Reaccionan al ambiente.
- **Interactivos.** Aquéllos en donde los usuarios influencian, de algún modo u otro, el curso de la música.[22]

Aunque DuBreuil, en [DuBreuil, 2020], hace una tipificación algo más local. Establece que *Musikalisches Wüferspiel* (traducido como "juegos musicales") es la primera obra musical generativa. A continuación, DuBreuil cataloga los algoritmos generativos

[19] Una manera algo más osada de definir música algorítmica podría ser: "*Música algorítmica se refiere a todo algoritmo que pueda suceder en alguna o algunas de las escalas temporales de la música propuestas por Roads*". Interesante como pregunta de investigación, aunque fácilmente prescindible si no se está de acuerdo con la manera de clasificación (por ejemplo, si se interpreta que Roads insinúa que todo el sonido debe estar en función del tiempo; no todo compositor musical está de acuerdo con ello). La prosa con que tales escalas están definidas sucita interpretaciones varias y muchos propondrían crear clasificaciones alternativas.

[20] ¿Esto sería plausible? En un contexto de pura computación musical, pareciera que sí.

[21] En "*Algorithmic Music for Mass Consumption and Universal Production*", Capítulo 34.

[22] En este entendido, música algorítmica en vivo y *live coding* forman parte de la música algorítmica interactiva.

según alguna de las tres propiedades que estos suelen cumplir, que son:

- **Azar o aleatoreidad.** Cuando el resultado del algoritmo es parcial o totalmente definido por el azar.
- **Determinismo.** Cuando el conjunto de reglas en el algoritmo determinan totalmente su resultado, sin intervención del azar.
- **Estocasticidad.** Cuando el algoritmo está basado en probabilidades de eventos.

DuBreuil asegura que el *Algorave* surgió de las técnicas arriba referidas, donde el software como *TidalCycles* y *Orca* permiten que el músico defina reglas en tiempo real y que el sistema cree música autónomamente.[23]

Durante un acto en vivo, los algoritmos que ya están implementados dentro de un lenguaje de *live coding* son llamados a escena y modificados en tiempo real. Así que un algoritmo generativo puede ser también interactivo. Parece ser que las categorías dadas por Levtov no necesariamente son mutuamente excluyentes, mientras que las de DuBreuil sí.

En conclusión, la manera correcta de clasificar a los algoritmos musicales dependería del contexto particular, así como de la opinión del autor y la bibliografía abordada. El creador musical siempre podría elaborar terminología propia.

Reiteraría además, la necesidad de disociar el término *algoritmo* con el de *algoritmo computacional*. Es decir, un algoritmo musical puede manifestarse sin que hayan computadoras interviniendo, del mismo modo en que sucede con los algoritmos no musicales.[24] Tal posibilidad flexibiliza conceptualizar una obra artística. De poco sirve restringir una definición de algoritmo musical al ámbito exclusivamente computacional si ello excluirá las múltiples formas que en el arte ya han sucedido, por ejemplo los cánones de Bach, los cuales se compusieron mediante procesos altamente algorítmicos. Otros ejemplos de algoritmos musicales que no requieren de un ordenador son: la imitación de la melodía que una persona A hace a la melodía cantada por una persona B. La variación que persona A hace a la melodía cantada por persona B. La traducción de una melodía que sucede en guitarra hacia un piano.

Examinando la historia de la música, hay dos algoritmos musicales que resulta indispensable conocer. Estos son: la escala pitagórica y el pentagrama. Además, para fines de esta tesis, hay que discutir lo que podría significar un algoritmo musical infinito; un término que existe y es válido de usarse en música algorítmica. También convendrá

[23] *Algorave* significa "fiesta de algoritmos" y refiere a aquéllos eventos donde varios artistas hacen *performances* de *live coding* para una audiencia. Particularmente, *Orca* es un lenguaje de *live coding* en el que cada letra del alfabeto funciona como un operador.

[24] Una receta de cocina es un algoritmo.

destacar y decir algo acerca de los algoritmos bailables; éstos son *algoritmos reactivos* que al ser ejecutados, provocan que alguien baile.

La escala pitagórica

Pitágoras, matemático de la Grecia Antigua, descubrió una escala musical de siete notas, conocida como *Escala pitagórica*. Tomó una cuerda vibrante afinada en una nota fundamental y notó que al tocar subdivisiones de cuerda en proporciones de $\frac{1}{2}$, $\frac{2}{3}$, $\frac{3}{4}$, $\frac{4}{5}$, $\frac{5}{6}$, $\frac{8}{9}$, y $\frac{9}{10}$, se obtienen, respectivamente, los intervalos de: octava, quinta justa, cuarta justa, tercera mayor, tercera menor, segunda mayor y segunda menor, con respecto a la nota fundamental.

Los intervalos de sexta y séptima diatónicos se obtienen, respectivamente, de: multiplicar una cuarta justa con una tercera mayor ($\frac{3}{4} \cdot \frac{4}{5} = \frac{3}{5}$), y de multiplicar una quinta justa por una tercera mayor ($\frac{2}{3} \cdot \frac{4}{5} = \frac{8}{15}$).[25] Es decir, una sexta está en razón de $5:3$ respecto a la nota fundamental, y una séptima, en una razón de $15:8$.

Esta escala tiene la estructura de escala mayor T T S T T T S, pero no permite transposiciones, lo que puede observarse en el hecho de que multiplicar dos segundas mayores es diferente de una tercera mayor. Este problema de afinación se resolvió modificando la escala pitagórica hasta llegar a una nueva, la bien temperada de doce tonos equitativamente distribuidos, también llamada *Escala cromática*.

Se puede entonces interpretar que la construcción de una escala es un algoritmo, –y determinista–, en el que el primer paso consiste en establecer una nota fundamental, y los pasos consecutivos serían tocar las siete subdivisiones de la cuerda.

El pentagrama

El pentagrama es una disposición gráfica de 5 líneas y cuatro espacios sobre el que se escribe la música. Al iniciar la escritura musical se indican la clave, armadura y la fracción de medida de compás con que la música será tocada. Las notas se dibujan sobre y entre las líneas del pentagrama, lo que indica la altura de cada nota. La duración de una nota se especifica con una *figura de nota*: redonda, blanca, negra, corchea, semicorchea, semisemicorchea, puntillo, etc. En algunos casos, la duración de nota se denota con ligaduras. Se añaden otros indicadores al pentagrama para establecer la velocidad y carácter de una ejecución, la intensidad, acentos, adornos, efectos, o maneras específicas de tocar un instrumento (por ejemplo, un *glissando* o un *pizzicato*).

[25] Lo que, en términos de distancias, significa sumar los intervalos.

La invención del pentagrama data de la Edad Media y le precedieron otros sistemas de notación musical: el monograma, bigrama, trigrama y tetragrama, que, como su nombre indica, usaban, una línea horizontal para denotar la música en el caso del monograma, dos líneas el bigrama, tres y cuatro líneas el trigrama y tetragrama.

Los mencionados sistemas de notación musical ocurrieron en el transcurso de dos siglos (XII y XIII). Antes de ello se creía imposible que la música pudiera traducirse al papel. Por mucho tiempo la música fue enseñada oralmente, dado que no había un mecanismo para transcribirla.[26]

Las primeras formas de escritura musical, del monograma al trigrama, ayudaban a los monjes a recordar melodías, aunque no fuera de una manera exacta. Por ejemplo, era posible señalar si una melodía ascendía o descendía, o cuántas notas intervenían, pero no quedaba registrada la altura de las notas, ni su duración. El tetragrama, inventado por Güido D'Arezzo, ya implementaba una visualización de las alturas y duraciones rítmicas, así como una clave y armadura.

Se puede decir entonces, que tomó tiempo descubrir todas las "variables" requeridas para transcribir ideas musicales precisas, y que el pentagrama es un algoritmo que nos "indica las instrucciones para reproducir una pieza musical totalmente definida." Esto por supuesto, puede someterse a discusión, y mencionarse algunos contraejemplos de contemporáneos sistemas de notación musical imprecisos (véase el Capítulo 2, Sección 2.2 de esta tesis). Sin embargo, lo que pretendo resaltar al interpretar que el tradicional pentagrama es un algoritmo, es que nos permite transcribir la música con total precisión.

Algoritmos musicales infinitos

Steve Reich, para su composición "*It's gonna rain*" (1965) diseñó un sistema mecánico con dos reproductores de cinta. Grabó un sampleo de voz, el cual reprodujo en ambas cintas con una diferencia de milisegundos de velocidad. Como una cinta avanza más rápido que la otra se empieza a crear una diferencia de fase entre ambos sampleos; un efecto que comienza a producir resultados sonoros tales como la ilusión auditiva, donde el oído empieza a imaginar que en la grabación hay cosas que en realidad no están, como un sonido de pájaros.

Luego de haber descubierto el efecto de desfase con las cintas, Reich lo aplicó en otras situaciones: en música con péndulos o en grabaciones de pianos.

Brian Eno tomó inspiración de "*It's gonna rain*", usando el mismo método en "*Music for Airports*" (1978), en un diseño con siete reproductores de cinta, en vez de dos.

[26]Ver: *A history of western music: Tenth international student edition*, en [Burkholder et al., 2019].

En cada *loop* había grabada una nota musical. Además, los *loops* de audio eran más largos y de diferentes longitudes. Estas piezas nunca se reconfiguran de la misma manera, por lo que se puede decir que tienen *longitud infinita.*

A este tipo de música Brian Eno le llamó "*Música generativa*". Un fenómeno que también está presente, según él, en las campanas de viento, pues siempre es impredecible su comportamiento, consiguéndose infinitas posibilidades sonoras.

La música generativa es pues, una noción que hace alusión a la infinitud en la música. En palabras de Norbert Herber,

> *El "infinito" a como es discutido en la música generativa, no es infinito en todo el sentido del término. En estas obras, el infinito puede por ejemplo, referir a la potencial longitud del performance o a la manera en que se percibe el desarrollo melódico y textural de una pieza. No sugiere necesariamente que un comprensivo acervo musical se ha encapsulado a un programa computacional que arrojará melodía tras melodía.*

Así pues, una forma de infinito se refiere a la longitud de una pieza, pero otra hace alusión a las texturas musicales y las posibilidades que hay para crear siempre nuevas de ellas. Se habla de restricciones compositivas, donde la infinitud está parametrizada o subordinada al contexto musical.

No hay que confundir pues, a la computabilidad infinita (es decir, máquinas de Turing infinitas), con el significado que al infinito puede dársele en una obra músical. Una pieza bien puede, a partir de un conjunto finito de elementos, generar un *output* infinito.

"*En C*" (1966), de Terry Riley, también es considerada una obra generativa:

- Cualquier número y tipo de instrumentos pueden participar en la pieza.
- Cada ejecutante elige cuántas veces repetir cada compás de la pieza antes de cambiarlo. Aunque, como el promedio de duración de una pieza es de 45 minutos, es de esperarse que cada compás sea tocado por intervalos de 45 segundos a un minuto y medio. Hay en total 53 compases.

Bajo estas restricciones se puede decir que la pieza "*En C*" tiene un alto carácter aleatorio, otra característica de los procedimientos generativos.

El album "*Reflection*" (2017), de Eno, consiste en un flujo infinito de música que se puede escuchar descargando una *app.*

Del mismo autor, la pieza "*Trope*" (2009) es un sistema generativo, e interactivo, pues permite que el escucha complete la música. De este modo, se desdibuja la línea divisoria entre compositor y escucha. Aunque el compositor deja incompletas algunas decisiones, sigue éste siendo el diseñador y controlador máximo del sistema, proveyendo el espacio sobre el cual el escucha explora la música.

Los sistemas generativos interactivos pueden diseñarse para dar al usuario variables grados de libertad. Entre más hay, más la interacción se vuelve una co-creación entre el sistema y el músico.

En conclusión, podemos vincular la noción de infinito con varios ejemplos de la música generativa, la cual es muy versátil, existiendo pues, en sus versiones instrumentales y computacionales. Por su origen, la música generativa alude a la música electrónica y forma parte de la vanguardia, a manera de sistemas y aplicaciones para *web*, *tablets*, instalaciones inmersivas, etc.

Los algoritmos aleatorios son típicos en la música generativa, siendo las cadenas de Markov el principal antecedente de nuevos métodos musicales aleatorios como las redes neuronales profundas, las cuáles aprenden a crear melodías basándose en un conjunto preescrito de datos. Las gramáticas generativas son otra opción. Una idea desarrollada por Chomsky en 1950, que consiste en formalizar las reglas del lenguaje hacia un sistema que sea capaz de generar frases en tal lenguaje. En particular esto también es aplicable a las gramáticas musicales.

Algoritmos bailables

El término *Algorave* surgió en el año 2011 para designar fiestas de *live coding*, y desde entonces éstos han seguido realizándose. El artículo "*Algorave: Live Performance of Algorithmic Electronic Dance Music*" ([Collins and McLean, 2014]) menciona dieciocho fiestas de algoritmos ocurridas entre 2012 y 2014; principalmente en ciudades inglesas, aunque también figuran países como Japón, España, México, Eslovenia, Canadá, Australia, Holanda y Alemania. El mismo artículo precisa las cifras aproximadas de asistentes y de la gente que en cada evento bailó.

Diversas revistas de música y periódicos como "*The Wire*", "*mixmag*", "*Vice*", o "*DJ Mag*" han llegado a realizar coberturas acerca de estos eventos que se han llevado el calificativo de "vibrantes y *geeks*".

Entonces, podemos decir que existen algoritmos bailables; aquéllos producidos por programas de *live coding* como son *Ixi Lang*, *TidalCycles*, *Fluxus* o *Sonic Pi*.

Live Coding significa generar música y sonido con una computadora, en los ánimos de improvisar con *laptops* como si de jazz se tratase; aunque modificar en tiempo real algoritmos de este tipo resulta ligeramente más tardado que tocar un instrumento, ya que: primero ha de modificarse la sintaxis del código a ser ejecutado y a continuación ha de evaluarse. En particular, ¡reescribir líneas de código puede tomar minutos (¡sí, un tercio de la duración de una canción!)!, e implica anticiparse al sonido que sus líneas especifican. En cambio al tocar un instrumento uno modifica la música y al mismo tiempo eso es escuchado. Esta crucial diferencia entre improvisación analógica e improvisación con ordenador es inevitable. En la actualidad es imposible modificar código al mismo tiempo que éste suena. Pero para mejorar esa situación lo que se suele hacer es reducir la sintaxis del código a menos caracteres, a partir de reinventar los nombres de las funciones en uso, así como los atajos del teclado.

El *algorave* está inspirado en el *techno* y en aquélla época en que surgió la música electrónica de baile, donde un primer elemento clave es la repetición, y un segundo elemento es la sensación de trance que producen los sonidos sintetizados; cosa que en los raves se acompaña con drogas de todos tipos, mismas que enaltecen las experiencias.

Podemos también decir que los patrones repetitivos son la manera que el humano tiene de percibir la simetría musical, la cual viene siendo el fundamento tanto de la música popular como las músicas regionales del mundo.

Contrastantemente a la música clásica, la música bailable es una especie de trivialidad. Aunque se puede hablar de *ballets* y *waltzes*, todo cambió con la llegada de la música atonal. Con ello la música académica se volvió más difícil de comprender para el común denominador de la gente. Afortunadamente, compositores como Steve Reich, Phillip Glass o Terry Riley, llegaron para repensar sobre las raíces.

En palabras de Brian Eno:

> "*Acordes simples. Intervalos simples. Ritmos fáciles que no están en 15/8 o cosas así. Es música que prácticamente te pone a bailar.*"

Es esta noción de bailabilidad que yo también considero para esta tesis, fundamental; La cognición ciertamente lo explica, pero yo lo veo más en términos de diversión y recreación. Una vez entendido esto, lograr un efecto exitoso desde el código resulta más difícil de lo que parece. El reto consiste en romper "la monotonía del *loop* repetitivo" a partir de la constante y acertada modificación del código y de las sonoridades involucradas.

1.3. Panorama de la música algorítmica

Esta tercera parte del primer capítulo repasa "*¿Qué es la música algorítmica?*" y menciona ejemplos de ésta. Se enfatizan las siguientes temáticas: a) Métodos tradicionales de composición. b) La música de Xenakis. c) Se intenta conectar el juego de dados de Mozart con las cadenas de Markov, y se intenta visualizar cómo influye ésto en la música de Xenakis. d) Se da una prioridad al concepto de matrices musicales.

Para una mayor profundización en los temas se sugiere consultar [Toussaint, 2019], [McLean et al., 2018], [Nierhaus, 2009], [Nierhaus et al., 2015], [Roads, 2001], [Xenakis, 1992], [Ames, 1989], [Noll et al., 2006].

1.3.1. ¿Qué es la música algorítmica?

Se suele emplear el término música algorítmica para referir:

1. La música que para su creación hace uso de algoritmos y métodos algorítmicos, donde por algoritmo se entiende el establecer instrucciones formales para obtener resultados precisos, como lo es una obra musical.

2. La música que hace uso parcial o total de un ordenador, lo que nos remite a esa escuela musical e inquietudes de los compositores de la primera mitad del siglo XX, cuyas pretensiones estaban encaminadas a algo conflictivo con lo que hasta ese momento había sido la música clásica. Surgió así la música electrónica creada con sintetizadores u ordenadores, música aleatoria, música estocástica, etc.

En este texto consideraremos que ambas cosas son música algorítmica, ya que el interés es el de encontrar puntos de vista matemáticos en ambas de las vertientes. Un algoritmo no necesariamente es computacional. A su vez, se puede encontrar geometría, acústica o simetría en la música y eso se puede definir en términos meramente matemáticos o formulaicos, sin la necesidad de programar.

1.3.2. Manifestaciones de la música algorímica

Consideraré tres facetas para la música algorítmica.

1. La idea de David Cope de que:

   > *"Todos los compositores, ya sea de manera consciente o inconsciente, son compositores algorítmicos", "todos los estilos musicales son algorítmicos (rock, jazz, música clásica, etc)" y que "para analizar las obras de los compositores, se requiere descubrir los algoritmos subyacentes, pues una vez logrado esto, el entendimiento acerca de los algoritmos incrementará la apreciación de la música por parte del escucha."*

2. Las diferencias entre la música basada en notas y la basada en el sonido. Lo primero se enfoca en sistemas compositivos, teoría musical o modelos de las

matemáticas discretas. Lo segundo se refiere a métodos de síntesis, teorema de Fourier, acústica, psiocoacústica, etc.

3. La computación musical, implementación de modelos e interconexión de interfaces.

Los planteamientos algorítmicos han estado presentes desde los cantos gregorianos, detalla Cope en [Nierhaus et al., 2015]. Una práctica que, de acuerdo a muchos musicólogos, significó el inicio de la música Clásica, aproximadamente en el siglo 9 a.C. En el caso de estos cantos, el algoritmo intrínseco es una melodía monofónica que sólo se mueve en pasos pequeños (es decir, en secuencias de intervalos de segundas menores y mayores, y donde muy raramente existen saltos por intervalos de tercera). La predilección hacia las melodías monofónicas simples y lineales permaneció por varios siglos, previo al surgimiento del contrapunto y armonía. Una restricción que vuelve sencilla la ejecución de las melodías y más aún, permite predecir las notas musicales que vendrán a continuación.[27]

Para Cope, lo que más parece alejarse de una algoritmia son el dodecafonismo y el serialismo; la "*Suite Op.25*" de Schoenberg, y las variaciones para piano de Anton Webern siendo ejemplos de esta música. Sin embargo, ello debería ser sujeto a debate. Tal vez, cuando lo dijo, Cope tenía en mente la incomprensibilidad sonora de la música atonal, en el sentido de que ya no se basa en consonancias contra disonancias, y se pierde algo de la lógica auditiva. Pero un posible contraejemplo sería la pieza "*Désordre*", de György Ligeti. Este es un estudio sobre polirritmias rápidas en piano, en el que la mano derecha únicamente toca teclas blancas mientras que la mano izquierda queda restringida a las teclas negras, separándose así las manos en dos tipos: la música de la mano derecha es diatónica y la de la mano izquierda es pentatónica.

Recalcando sobre la música tonal, –también llamada por Hugo Riemann (1849 - 1919), armonía funcional–, las progresiones armónicas desencadenan patrones *predecibles*. Las cadencias perfecta (I – V – I, o I – IV - V - I), plagal (I – IV - I), imperfecta (I – V, II – V, o IV - V) o interrumpida (I – V – VI, o I – IV – V - VI) son algoritmos. Es decir, son pequeñas estructuras preescritas que sirven de colchón para las frases melódico-armónicas. Al analizar sucesiones largas de armonías en una pieza, se suele poder descomponer la música en uniones de cadencias elementales.

La música atonal ya no considera grados tonales con algunos grados siendo más importantes que otros. El dodecafonismo propone en sus reglas la igualdad de todas las notas comprendidas en una octava, eliminando la tónica como aquélla en torno a la cual se ordenan el resto de los sonidos. La distinción fundamental entre serialismo y dodecafonismo estriba en que el principio serial se puede aplicar a varios parámetros

[27] David Cope (1941) es un compositor y científico radicado en California, cuyas investigaciones principales se centran en inteligencia artificial y música.

musicales (ritmo, dinámica, timbre, etc) y no sólo a la altura de las notas, como lo sugería la propuesta dodecafónica original.

El método dodecafónico consiste en seleccionar una línea con las doce notas de la escala cromática en algún orden elegido por el compositor, y a continuación aplicarle a esa línea varias transformaciones posibles, como son el retrógrado, inverso retrógado, inverso, o transposición, eventualmente regresando a la melodía original. Nuevamente, estas transformaciones significan simetrías y traslaciones de una sucesión de puntos en un plano, y de ahí que pueda considerársele un método algorítmico de composición, independientemente de que la noción de consonancia esté debilitada.

Música y matemáticas han confluido desde hace milenios y a través de las distintas culturas. Normalmente se piensa que lo que éstas disciplinas tienen en común se localiza en la técnica con que las piezas musicales son producidas, pero también pueden éstas unirse desde los conceptos. Citando a Ligeti:

> *El lugar en donde las bellezas musical y matemática empatan, no es en el nivel de las máquinas que crean música mediante cálculos algorítmicos, sino que la conexión se da en niveles más profundos.*

Según Collins, ejemplos de música y matemáticas son: la escala musical que descubrió Pitágoras, la afinación de los instrumentos musicales, el método de Güido D'Arezzo (1026), los autómatas y cajas de música, los escritos de Ada Lovelace o el sistema compositivo de Joseph Schillinger.[28]

Yo enfatizaría rescatar la discusión de Cope acerca de la música medieval y de su paso a la polifonía. Lo que comenzó partiendo de melodías monofónicas con ciertas reglas de movimientos permitidos y no permitidos de notas, evolucionó al más complicado problema de tocar líneas melódicas simultáneas; dos, tres o hasta cuatro de ellas, lo cual conlleva observar cómo se comportan armónicamente conjuntos de notas para con base en ello decir qué suena bien. Fue un proceso de tiempo el que se requirió para pasar de la melodía a la armonía, y de la mano se llegó al contrapunto, inventado por Johann Sebastian Bach, el cual también introdujo las formas musicales del canon y la fuga.

En general los sistemas compositivos son altamente algorítmicos. La misma definición de sistema nos habla de "un conjunto de principios o procedimientos a través de los cuáles algo es realizado; es un esquema o método organizado." Cuando hablamos de una sonata, nos referimos a dejar fija una *macroforma* sobre la cual toda sonata debería ser compuesta: Primero está la exposición, donde se presentan, en dos grupos o temas distintos (a forma de llamada y respuesta) las figuras melódicas principales

[28] Ver: *Ada Lovelace: The Making of a Computer Scientist*, en [Hollings et al., 2018]; y *The Schillinger system of musical composition*, en [Slonimsky, 1946].

para la pieza. Luego está el desarrollo, donde se exploran y varían las mismas figuras que aparecieron durante la exposición y donde además se realizan varios cambios de tonalidades. A continuación llega la recapitulación, en donde se regresa a algo muy parecido a la exposición, pero con algunas diferencias menos contrastantes entre los temas. Podemos decir, entonces, que la estructura global de la pieza es un algoritmo que alude a las secciones que una sonata debe tener. Eso no quiere decir que toda sonata sea igual, ya que la vista global no determina todo el contenido de una pieza musical. También se puede optar por no seguir el algoritmo, lo que convendrá en una forma variada de sonata o en otro tipo de forma musical.

Los sistemas de ejes de Béla Bartok, o el método Neo-Riemanniano también prescriben, aunque de una manera más local, cómo usar progresiones de acordes para componer piezas. El primero de ellos sigue siendo considerado un esquema tonal, mientras que el segundo ya es considerado un método atonal. A grandes rasgos podemos decir que, en la música basada en notas, hay dos clasificaciones para los sistemas compositivos: tonales y atonales. En ambos casos, los algoritmos a seguir pueden especificarse en diferentes escalas de una pieza. Por ejemplo, una progresión armónica abarca uno o varios compases, mientras que una forma global considera la composición completa.

Métodos de composición no basados en notas surgieron con el nacimiento de la música electrónica. Los sintetizadores o los métodos de grabación de audio abrieron nuevas posibilidades para generar sonido, organizarlo y modificarlo.

En el año de 1897, Thaddeus Cahill inventó el primer instrumento musical electrónico, llamado Telarmonio, con el que se realizaron conciertos vía telefónica para las audiencias; ello marcó una serie de subsecuentes invenciones tales como el *Audion*, *Theremin*, *Sphärophon*, *Dynaphone*, *Trautonio* u *Ondas Martenot*. Todo esto propició el surgimiento de la música electrónica (la cual le precede a la música computacional). Compositores como Messiaen, Milhaud, Ibert, Babbit, Varèse, Boulez, Stockhausen, o Cage, vieron en estos generadores sonoros, la posibilidad de explorar sonidos que iban más allá de la notación musical.[29]

Por su parte, Russolo, Dalapicolla y en general, los futuristas italianos influyeron en el cine y la música electrónica, introduciendo ruidos de trenes, aeroplanos o fábricas en su música, haciéndose alusión a esa época de la Revolución Industrial.

Laurens Hammond, en 1935, desarrolló el primer instrumento electrónico comercial, el órgano Hammond. Y en los años 50 aparecieron las grabadoras de cinta y sintetizadores; siendo el *Mark I* y *Mark II* los primeros. De aquí se derivaron aún más posibilidades de manipulación de material sonoro. En vez de frecuencias discretas, éstas podían ser barridas de manera continua mediante *knobs*. La lógica para manipular estos aparatos era mera consecuencia de la maquinaria del sintetizador. Ciertamen-

[29] *Electric Sound: The Past and Promise of Electronic Music*, en [Chadabe, 1997].

te, para muchos compositores de la época, las posibilidades tímbricas y de ejecución de un instrumento convencional quedaron rebasados por los sintetizadores. No fue sino hasta Wendy Carlos y el sintetizador *Moog* que comenzaron a aparecer modelos de sintetizadores con teclados añadidos; es decir, cuando ya fue posible tocar notas musicales con sintetizadores.

La primera composición musical hecha en su totalidad por una computadora fue *Illiac Suite* (Hiller, Isaacson, 1957), que hacía uso de cadenas de Markov. A esto le siguió la música estocástica de Xenakis, quien además desarrolló programas de computadora para producir música sintetizada.

La historia de los lenguajes de programación musicales data desde 1957, cuando apareció *MUSIC I* de Max Matthews. Le siguieron *MUSIC II-MUSIC IV*, *MUSIC 360* y *MUSIC11*.[30] Desde *MUSIC III*, *Csound* comparte características con estos programas antecesores, si bien *Csound* es uno entre muchos lenguajes de programación musicales que actualmente existen para computadora personal, entre ellos *SuperCollider*, *PureData*, *Max/Msp*, etc.

En particular la música electrónica de los años sesenta (Ligeti, Stockhausen, Berio, Reich, Glass, etc) puede confundirse con la música que emplea ordenadores. Esto es porque la computación musical es, parcialmente, una compactificación de los métodos de grabación, reproducción, alteración y generación de sonidos para los que antes eran necesarias grandes máquinas. Hay gran similitud, por ejemplo entre la síntesis granular y los efectos que figuran en la pieza *Omaggio a Joyce*, de Luciano Berio. O un proceso de alteración del *pitch*, presente desde los tiempos de Pierre Schaeffer y su música concreta, en la actualidad se reduce a ejecutar una línea de código.

1.3.3. Matrices musicales

L.M. Frimberger, en *The Music Theory Matrix: Squaring the Circle of Fifths* (página 3), define una matriz como "*un arreglo rectangular de cantidades o expresiones en renglones y columnas, tomada como una sola entidad y manipulada de acuerdo a reglas particulares* ."

Simplemente, una matriz es una ordenación de elementos en una tabla.

Cualquier tabla es una matriz.

Las tablas encuentran varios usos cotidianos: agendar las actividades de la semana, llevar una lista de calificaciones de los alumnos de una clase, llevar el presupuesto de todos los materiales requeridos para la construcción de una casa o llevar un registro

[30] *Csound: a sound and music computing system*, en [Lazzarini et al., 2016].

de los gastos que una persona hace cada mes. En este sentido, el uso de una tabla puede quedarse en un uso muy básico que es el de enlistar elementos, en el entendido de que esa configuración es práctica para almacenar datos.

Un uso que en música se le pueden encontrar a las matrices es este mismo: una tabla contieniendo información que será consultada para la ejecución de la pieza. Puede ser que la matriz indique el orden en que en el tiempo serán tocados los acordes y melodías que ésta establece. Pero el empleo y contenido una matriz musical varía enormemente dependiendo del contexto.

Usos de matrices en música son: el juego de dados de Mozart; la matriz serial de doce tonos; el *Iching* en la obra "*Music Of Changes*" (Cage, 1951); las piezas *Achorripsis* (Xenakis, 1957) y *Analogique B* (Xenakis, 1959) de Xenakis, donde, respectivamente él recurre a la teoría de juegos y las cadenas de Markov. Otro ejemplo es la matriz interválica, que enlista las distancias que hay entre dos notas de una escala.

Parte de la práctica de las matemáticas consiste en deducir patrones. Si la música manifiesta comportamientos parecidos, parece pertinente confiar en que, a pesar de todo el bagaje de información que actualmente en libros esté publicado, mucho de ésto sea imprescindible para completar las propias deducciones. En particular, se puede componer una pieza original usando matrices, y en el proceso ir deduciendo cómo tiene que ser su comportamiento.

Podríamos, por ejemplo, construir una matriz de intervalos musicales sin tener que previamente conocerla: ya que en una melodía polifónica hay intervalos simultáneos y sucesivos, es natural designar que los renglones definan intervalos simultáneos y que las columnas refieran intervalos sucesivos (dos renglones marcados con uno en cierta columna corresponderían a que, en la región de tiempo asignada a esa columna, sonarían dos notas produciendo un intervalo).

Una matriz interválica puede entonces: o abstraer lo que en una específica región musical de una pieza está ocurriendo o simplemente analizar todos los intervalos posibles de manifestarse en una pieza que use una escala cromática.

Así pues, las matrices musicales se prestan para adoptar al menos dos tipos de roles: el de estructurar piezas específicas o el de definir el espectro de posibilidades que en una pieza ciertos elementos puedan ocurrir.

Un tipo de matriz musical muy diferente surge en la síntesis de sonido; le podemos llamar "matriz de modulación", y sirve para definir flujos de señales (ver Figura 1.1). Puede modelarse por ejemplo, la situación de un oscilador modulando la frecuencia de otro oscilador que a su vez modula la amplitud de otro oscilador.

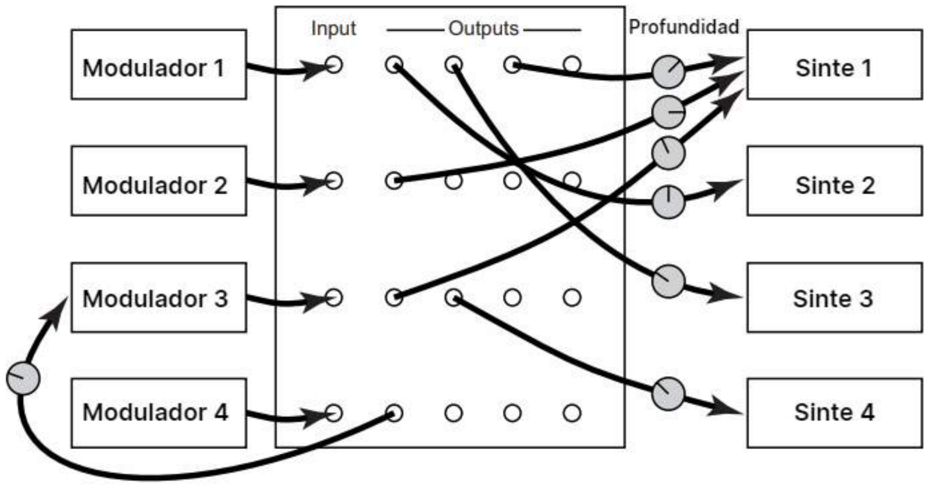


Figura 1.1: Matriz de modulación en la síntesis de sonido.

En el proceso de síntesis, una matriz de modulación determinaría las posibles conexiones a efectuarse sobre señales, envolventes, filtros y efectos. Además, los sintetizadores suelen implementar un puñado de algoritmos específicos para configurar señales (por ejemplo, 32). Son entonces una colección finita de algoritmos preestablecidos con los que se puede operar el sintetizador.

En *SuperCollider*, existe un árbol de nodos encargado de combinar y secuenciar señales. Es algo más sutil pensar que eso también puede reinterpretarse como una matriz.

Sin embargo, la teoría matemática sugiere ir un nivel más allá, respecto a la manipulación de matrices. Por ejemplo:

- La teoría de matrices propone varios métodos para encontrar soluciones a sistemas de ecuaciones difíciles.
- Para calcular potencias de matrices se suele descomponer una matriz potencia, en combinaciones lineales de matrices más elementales, como son las matrices de permutación.
- Otro resultado famoso es la forma canónica de Jordan donde, para resolver un polinomio, se busca diagonalizar una matriz, particionando una matriz grande en submatrices $A_1, A_2, ..., A_n$ tales que todas ellas sean casi diagonales.

Todo esto suscita la pregunta de si tales observaciones del álgebra lineal llegarían a tener una aplicación a la creación musical.

Xenakis planteó una cuestión así cuando inventó la música estocástica, dado que consideró que al serialismo le faltaba más análisis. Esto en tanto que los compositores, al considerar la música sólo desde la ejecución de los pasajes musicales, estaban ignorando el comportamiento más global del método serial; es decir, no tomaban en cuenta cómo se comportan las masas de sonidos en su conjunto, lo cual sí puede predecirse tomando promedios o encontrando distribuciones de probabilidad.

Xenakis, en *Analogique B*, utiliza el recurso de construir una matriz de transición de dimensión 8×8, a partir de su descomposición en varias matrices 2×2. Está de más decir que él era totalmente consciente de que era posible llegar a la matriz grande a partir de considerar otras más pequeñas.

Una vertiente que ofrece atractivas aplicaciones es la combinatoria en matrices. Por ejemplo, para la matriz interválica que enlista todos los posibles intervalos que hay entre notas de una escala, se pueden idear restricciones al problema de seleccionar sólo ciertos intervalos, aplicándoles luego un algoritmo aleatorio que les eligiera para sonar dentro de una melodía. Se puede hacer algo similar para generar acordes. Más aún, con la información obtenida podría construirse una matriz gigante que consistiera de una partición de ésta en varios módulos: uno para las melodías, otro para las armonías, y otro que relacionara los dos anteriores, de tal manera que el programa resultante hiciera interactuar melodías contra acordes, a partir de primero seleccionar ciertas melodías y acordes.

En conclusión, hay varios tipos de matrices musicales, que sirven ya sea para:

- Estructurar la música, ya sea de forma global (en donde se aprecien todos los elementos que participan en la pieza) o de una forma local (en donde las matrices sean porciones de una pieza total, modelando cómo se comporta alguno o algunos de los elementos que participan en la pieza).
- Representar una partitura de una manera numérica o computacional.
- Proveer el conjunto de posibilidades (o espacio muestral) para algún o algunos elementos que formen parte de una pieza.
- Determinar flujos de señales.

Refraseando lo anterior, otra clasificación alternativa de matrices musicales es en estáticas y dinámicas. La primeras definiendo espacios muestrales y las segundas fijando la específica presentación en el tiempo de la música (a manera de *score*).

Se pueden implementar algoritmos sobre tales matrices con la finalidad de componer piezas.

La manera de aplicar el concepto de matriz musical va desde lo ambiguo hasta lo más preciso, ya que una matriz, siendo una herramienta intuitiva de organización de elementos, puede aplicarse tanto sobre ideas algo indefinidas, como en números. También, hay varios niveles con los que trabajar las matrices: usándolas sólo como tablas que almacenan datos o como elementos sobre los que se buscará efectuar algún algoritmo.

1.3.4. Juegos de dados

Una ascepción de *Musikalisches Würfelspiel* es el *juego de dados de Mozart*. Previo a Mozart ya existían composiciones aleatorias similares a ésta (las de Haydn o Kirnberger), que no necesariamente requieren dados pero sí se les considera juegos musicales. Sin embargo es el juego de dados de Mozart el que ha sido más famoso. Consiste en lanzamientos consecutivos de dos dados, hasta generarse una pieza aleatoria de longitud de 16 compases.

Antes de iniciarse, el juego consiste de: 1. Una matriz con números fijos. 2. Una partitura fija cuyos compases están numerados. Mozart especificó que la matriz tuviera 11 renglones y 16 columnas, donde a cada entrada le asoció un número, el cual refiere al compás musical numerado por tal (ver Figura 1.2 y Figura 1.3). Con cada tirada de los dados, el resultado se suma; obteniéndose el correspondiente renglón de la matriz (por eso, como las posibles sumas de dados va del 2 al 12, es que la matriz tiene once renglones). La primera tirada de los dados nos indicará lo que hay que poner en el primer compás de la pieza aleatoria. La enésima tirada de los dados indicará lo que poner en el enésimo compás. Así, las columnas en la matriz representan el número de compás de la pieza. En total, los dados habrán sido lanzados dieciséis veces y se habrá cubierto el total de las columnas.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
2	96	22	141	41	105	122	11	30	70	121	26	9	112	49	109	14
3	32	6	128	63	146	46	134	81	117	39	126	56	174	18	116	83
4	69	95	158	13	153	55	110	24	66	139	15	132	73	58	145	79
5	40	17	113	85	161	2	159	100	90	176	7	34	67	160	52	170
6	148	74	163	45	80	97	36	107	25	143	64	125	76	136	1	93
7	104	157	27	167	154	68	118	91	138	71	150	29	101	162	23	151
8	152	60	171	53	99	133	21	127	16	155	57	175	43	168	89	172
9	119	84	114	50	140	86	169	94	120	88	48	166	51	115	72	111
10	98	142	42	156	7	129	62	123	65	77	19	82	137	38	149	8
11	3	87	165	61	135	47	147	33	102	4	31	164	144	59	173	78
12	54	130	10	103	28	37	106	5	35	20	108	92	12	124	44	131

Figura 1.2: La matriz aleatoria de *Musikalisches Würfelspiel*.

Musikalisches Wüferlspiel también ilustra la intrínseca relación que existe entre partitura y matriz, la primera siendo un elemento musical esencial y la segunda, un objeto extensivamente estudiado en matemáticas. En el Capítulo 2, Sección 2.2 se detalla esto.

Figura 1.3: "*Musikalisches Würfelspiel*", primeros 8 compases.

1.3.5. Cadenas de Markov

En la teoría de la probabilidad se le llama "evento" a la propiedad relativa de un experimento aleatorio cuya característica es que una vez realizado éste, podemos decir si ocurre o no ocurre.

Así pues, una cadena de Markov modela una sucesión de eventos y predice lo que sucederá en un futuro inmediato basado en lo ya ocurrido.

Las cadenas de Markov fueron propuestas en el año de 1906 por Andrei Andreevich Markov (1856-1922). Son fundamentales en Procesos Estocásticos, donde se utilizan para modelar procesos aleatorios tales como "la ruina del apostador" o "la fluctuación del precio de una acción en un mercado que está siendo afectado por la ley de oferta y demanda".

Se le llama cadena de Markov infinita a aquélla que tiene un infinito número posible de estados. Pero es el modelo finito el que solemos usar para los algoritmos computacionales.

Una cadena de Markov se puede representar o bien gráficamente, o mediante una expresión algebraica, o por medio una matriz de transición.

La gráfica que se observa en la Figura 1.4 muestra una cadena de Markov de tres estados, donde cada estado es un vértice. Las aristas entre nodos (nombre alternativo para los vértices) indican la probablidad de cambiar hacia tal estado, dado que nos encontremos en cierto estado específico. Es decir, si el estado 1 significa clima lluvioso, el estado 2 significa clima soleado y el estado 3 significa clima nevado, la gráfica nos dice que hay 1/2 de probabilidad de que mañana sea clima lluvioso dado que hoy fue clima lluvioso, hay 0.4 de probabilidad de que mañana sea clima soleado dado que hoy llovió, y hay 0.1 de probabilidad de que mañana neve considerando que hoy llovió.

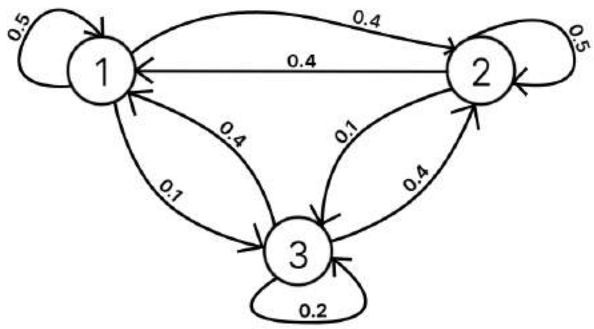


Figura 1.4: Una cadena de tres estados en su representación gráfica.

La matriz de transición provee la misma información que su gráfica:

	Lluvia	Sol	Nieve
Lluvia	0.5	0.4	0.1
Sol	0.4	0.5	0.1
Nieve	0.4	0.4	0.2

Figura 1.5: Matriz de transición que modela el clima.

Cuando la cadena de Markov es pequeña (como es este ejemplo de sólo tres estados), de un rápido vistazo a la gráfica podemos saber si la cadena de Markov es irreducible.[31] Alternativamente, para una cadena de Markov en su versión matricial se podrán realizar las correspondientes operaciones que determinen si ésta es irreducible o no (aunque fuese muy complicada), así como nos permitirá encontrar algún vector estacionario.[32] Formalmente, decimos que una cadena de Markov finita se conforma por:

- Un conjunto $Q = \{q_1, q_2, \ldots, q_N\}$ de estados.
- Una matriz $A = (a_{11}, a_{12}, \ldots, a_{n1}, \ldots, a_{NN})$ llamada **matriz de transición** (o matriz de probabilidades de transición), donde cada a_{ij} representa la probabi-

[31] Una cadena de Markov es irreducible si es posible ir de cualquier estado a cualquier otro en un número finito de pasos.

[32] La definición de vector estacionario se encuentra en la subsección de cadenas de Markov en el largo plazo, algunos párrafos más adelante.

lidad de pasar del estado i al estado j. Cada renglón en la matriz cumple que la suma de sus entradas es 1, es decir: $\sum_{j=1}^{N} a_{ij} = 1 \ \forall i$.

- Una **distribución inicial de probabilidades** $\pi = (\pi_1, \pi_2, \ldots, \pi_N)$ sobre los estados, donde π_i es la probabilidad de que la cadena de Markov empiece en el estado i. Algunos estados j pueden tomar el valor $\pi_j = 0$, lo que significa que no pueden ser estados iniciales. Además se cumple que $\sum_{i=1}^{N} \pi_j = 1$.

Se dice que una cadena de Markov es de *primer orden* cuando su estado actual únicamente depende del estado anterior. Una cadena de Markov de *segundo orden* es aquélla cuyo estado actual únicamente depende de los dos estados previos. Análogamente, una cadena de Markov de n-*ésimo orden* es aquélla cuyo estado actual depende de los n estados que le preceden.

Cadenas de Markov musicales

En música, acorde con la sección anterior, un evento bien puede interpretarse como un compás con notas, o una sucesión de compases en una partitura, aunque puede adoptar significados diversos; por ejemplo, en el ámbito de la síntesis sonora. Xenakis denotó por eventos a las regiones temporales de diversas longitudes en las cuáles habitaban sus granulaciones y cadenas de Markov. Así pues, en el contexto de la creación y programación musical, varía la acepción de "cadena de Markov", y en realidad este modelo ha sido muy popular con aplicaciones varias, como da cuenta Ames.[33] Más recientemente, las cadenas de Markov de orden variable se han empleado para recrear el estilo musical de pianistas.[34]

Una cadena de Markov musical de primer orden es capaz de modelar aspectos muy simples de una pieza, como lo sería tocar notas de una escala pidiendo que las notas consecutivas siempre sean distintas. Así mismo, se pueden elaborar cadenas de Markov que generen únicamente secuencias armónicas, y otras que produzcan frases; para despues combinar los tres sistemas independientes.

Si se pretende recrear un estilo, el uso exclusivo del primer orden es insuficiente, ya que, tanto en la técnica instrumental como en una composición intervienen muchos más parámetros (en una pieza para guitarra los *bends* serían una variable, así mismo los vibratos, *slides*, *hammer-ons*, *pull-offs*, escalas, melodías, fraseos, armonías, ritmos, o variaciones) que se van manifestando e interrelacionando entre sí, multidimensionalmente, y en un intervalo de tiempo que no es necesariamente el inmediato anterior al estado actual. Sin embargo, con los métodos computacionales que actualmente hay, una cadena de Markov de n-ésimo orden lograría replicar exactamente al

[33] *The markov process as a compositional model: A survey and tutorial*, en [Ames, 1989], p. 177
[34] Harnik, Holger, Nierhaus, en *Patterns of Intuition*, en [Nierhaus et al., 2015], pp. 9 - 32.

corpus original.

Las cadenas de Markov de n-ésimo orden, según Nierhaus ([Nierhaus, 2009]), suelen proporcionar los mejores resultados musicales, pero requieren más recursos computacionales, ya que agregar más memoria a una cadena de Markov aumenta el número de parámetros del programa.

En el contexto de aprendizaje de máquinas, por lo tanto, es en un punto intermedio en que las "cadenas de Markov de orden variable" resultan aptas para crear música original, pero que continúan preservando aspectos importantes de la pieza modelo, consiguiendo así emularse un estilo.

El modelo de música generativa de Ames, - quien, para elaborarlo, a su vez estudió varios modelos ya preexistentes como los de Hiller y Baker o Conklin -, consiste en una cadena de cadenas Markov ordenadas jerárquicamente, donde una cadena de Markov que construye sucesiones de notas aisladas con duraciones variables es considerada de bajo nivel, mientras que una cadena de Markov operando los cambios armónicos o los fraseos tiene un nivel estrictamente mayor. Y a su vez, una cadena de Markov de un nivel mayor a las dos anteriores serviría para combinar las matrices de transición de los niveles más bajos.

El esquema anterior equivaldría, o mejor dicho, sería capaz de recrear un modelo de improvisación dirigida en piano (ver Figura 1.6), donde, para una nota antecedente habría varias posibles realizaciones de notas consecuentes, y la selección de una u otra podría competerle a una primera cadena de Markov. Luego, para un compás musical antecedente habría algún compás consecuente que también podría ser elegido mediante una cadena de Markov, y finalmente, dados dos compases antecedentes habría una cadena de Markov que eligiera un fraseo consecuente con longitud de dos compases. Pero así como el músico en un determinado instante de tiempo tiene la libertad de elegir tocar consecuentemente o bien en el nivel de las notas o bien en el nivel de un compás o varios de ellos, así mismo lo debería poder hacer un programa que implementara varios niveles de cadenas de Markov.

Figura 1.6: Antecedentes y consecuentes en la improvisación dirigida.

Desde "*Illiac Suite*" (1957), la primera composición musical elaborada con una computadora, las cadenas de Markov encontraron un uso. Xenakis también las empleó en sus composiciones orquestales y computacionales, especialmente enfocándose en su comportamiento en el largo plazo.

Más actualmente, Volchenkov y Dawin han mostrado que el juego de dados de Mozart es programable mediante cadenas de Markov.[35]

En vista de lo anterior, podemos afirmar la efectividad que el modelo matemático de cadena de Markov tiene para la interpretación de fenómenos musicales.

Las cadenas de Markov han influido en el desarrollo de nuevos modelos probabilísticos partícipes en el auge de la música con inteligencia artificial.

Un modelo de Markov para la armonía funcional

A continuación muestro un modelo de Markov de primer orden que genera sucesiones plausibles de acordes, considerando que el esquema tonal de la armonía está basado en estabilidad contra inestabilidad. A una tonalidad dada le corresponde una escala diatónica mayor y sobre ésta se construye una armonización por acordes, lo que da lugar a los grados armónicos I, II, III, IV, V, VI y VII. Por su estabilidad e inestabilidad, estos acordes pueden clasificarse en:

Estables	Tónicos	I, III, VI
Más o menos estables	Subdominantes	II, IV
Inestables	Dominantes	V, VII

De acuerdo con la tabla anterior, si pedimos que el estado inicial sea el primer grado (I), que es estable, es muy probable que a éste le responda un acorde inestable. Por lo que, asignémosle un 40 % de probabilidad de que a I le responda un V. Ello toma en cuenta que aunque el séptimo grado (VII) es también un acorde inestable, casi no se utiliza en las cadencias.

La siguiente tabla asigna las probabilidades de que el estado I pase al resto de los acordes (o de que se quede en el mismo estado).

	I	**II**	**III**	**IV**	**V**	**VI**	**VII**
I	0.02	0.2	0.08	0.21	0.4	0.08	0.01

[35]"*Musical Markov Chains*", en [Volchenkov and Dawin, 2012].

Ello tiene presente que son sustituibles entre sí los acordes de cada categoría, aunque con la excepción del acorde disminuido y considerando que la cadencia I – IV – V es común.

Para construir el resto de la matriz de transición, la elección debe hacerse caso por caso, pues por ejemplo, dado el cuarto grado como punto inicial, sería altamente probable que le respondiese un primer grado o un quinto grado, dando así lugar a una cadencia plagal o en el otro caso, abriendo paso a una cadencia perfecta. Por ende, las probabilidades de transición cambian dependiendo del estado en que se inicie.

	I	II	III	IV	V	VI	VII
I	0.02	0.2	0.08	0.21	0.4	0.08	0.01
II	0.1	0.1	0.1	0.1	0.4	0.1	0.1
III	0.1	0.1	0.1	0.1	0.4	0.1	0.1
IV	0.3	0.06	0.06	0.06	0.4	0.06	0.06
V	0.5	0.1	0.025	0.025	0.025	0.3	0.025
VI	0.1	0.1	0.1	0.1	0.4	0.1	0.1
VII	0.5	0.025	0.2	0.025	0.025	0.2	0.025

Figura 1.7: Una matriz de probabilidades que modela la armonía tonal.

La cantidad de datos hace que, en este caso sea más sencillo trabajar con la matriz en vez de con su gráfica.

Cadenas de Markov en el largo plazo

Se dice que una cadena de Markov es irreducible si es posible ir de cualquier estado a cualquier otro en un número finito de pasos. Es decir, para todo i, j, existe k tal que:

$$P(X_{n+k} = j \mid X_n = i > 0).$$

Se dice que un vector π de una matriz de transición regular Q es estacionario (o que π es una distribución es estacionaria) si es el único vector de probabilidad que satisface la ecuación $Q\pi = \pi$.

Una cadena de Markov es aperiódica si para cada estado i existe una constante M_i tal que, para todo $n > M_i$ existe una trayectoria de probabilidad positiva que conduce de i a i en n pasos.

De la teoría de procesos estocásticos son relevantes los siguientes resultados:

- Una cadena de Markov con un espacio muestral finito tiene al menos una distribución estacionaria.
- Una cadena de Markov finita e irreducible tiene una única distribución estacionaria.
- Una cadena de Markov finita, irreducible y aperiódica, en el largo plazo convergerá a su única distribución estacionaria.

A manera de ejemplo, examinemos el siguiente problema:

"*Sea Y_n la suma de N tiradas de dado independientes. Determina, en el largo plazo, con qué probabilidad Y_n resulta ser múltiplo de 7.*"

Solución.

Sea Y_n la suma de las N tiradas de dado que se han realizado.

Y_n es múltiplo de 7 o no lo es. Es decir, al dividir Y_n entre 7 obtendremos un residuo, que es alguno de los enteros $0, 1, 2, 3, 4, 5, 6$. El número $Y_n = 7k+b$ estará, por lo tanto, en alguna de la siguientes clases de equivalencia: $\overline{0}$, $\overline{1}$, $\overline{2}$, $\overline{3}$, $\overline{4}$, $\overline{5}$, o $\overline{6}$ (dependiendo de quién sea b).

Esto induce una cadena de Markov de siete estados, que se puede representar mediante la siguiente matriz:

$$\mathbf{P} = \begin{pmatrix} 0 & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\ \frac{1}{6} & 0 & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{6} & 0 & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & 0 & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & 0 & \frac{1}{6} & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & 0 & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & 0 \end{pmatrix}.$$

Ahora utilizamos el resultado de que si una matriz es regular entonces converge a una única distribución límite $\pi = (\frac{1}{N}, \ldots, \frac{1}{N})$. En este caso, $N = 7$. Así, el vector estacionario para esta cadena es $\pi = (\frac{1}{7}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7})$.[36]

Como Y_n es múltiplo de 7 si y sólo si $X_n = \overline{0}$, la primera entrada de π verifica que, en el largo plazo (es decir, cuando N es muy grande), la probabilidad de ser múltiplo de 7 es $\frac{1}{7}$.

El apéndice al final de la tesis detalla la explicación de este problema.

Mikka (para violín solo, 1971), *Mikka "S"* (1976), *N'Shima* (1975), *La Légende d'Eer* (1977), *Gendy 3* (1991), y *S.709* (1994), obras de Xenakis, muestran el uso de las caminatas aleatorias y movimiento browniano en el largo plazo.

¿Pero a qué suena el largo plazo?

"*Analysing Mikka "S": from Micro to Macrocomposition and Perception Features*" (Rossetti, Antunes, Manzoli, 2022) es un estudio de *Mikka "S"*, el cual, mediante un seccionamiento en pedacitos de audio y su posterior análisis en *software*, se demuestra gráficamente que, para esa composición, el largo plazo del movimiento browniano converge en aspereza (en inglés, *Roughness*).

Aunque esto responde sólo parcialmente la pregunta.

La síntesis estocástica de Xenakis se ocupó de interrelacionar parámetros sonoros en la microescala, es decir desde los componentes más pequeños de un timbre sonoro y su modificación temporal. Pero sin duda, el concepto de cadena de Markov de largo plazo podría pensarse en el nivel del ritmo o sobre el ya mencionado "sistema jerárquico de cadenas de cadenas de Markov", el cual funciona como la interpretación matemática de una pieza escrita para partitura. El contexto en que se aplique determinará si el largo plazo encuentra ahí un significado y cuál es.

1.3.6. Escalas temporales del sonido

Curtis Roads, en *Microsound* ([Roads, 2001]), tipifica nueve diferentes escalas de tiempo que hay en la música:

1. **Infinita.** El tiempo ideal de las duraciones matemáticas, por ejemplo, las ondas sinusoidales.

2. **Suprasonido.** Una escala de tiempo que va más allá de una composición individual, abarcando meses, años, décadas y centenarios. Aquí se encuentran los géneros musicales o las estaciones de radio.

[36] *An Introduction to Stochastic Modeling*, en [Pinsky, 2011], Transition probability matrix en ScienceDirect y el apéndice.

3. **Macrosonido.** La escala de tiempo que engloba una obra musical en su totalidad, determinando su forma. Se mide en minutos, horas o días.

4. **Mesosonido.** Subdivisiones de la forma. Es la agrupación de los objetos sonoros en jerarquías de frases y estructuras de varios tamaños. Se mide en minutos o segundos.

5. **Objeto sonoro.** La unidad básica de la estructura musical que generaliza el tradicional concepto de nota, incluyendo así a las transformaciones en el tiempo de un sonido que abarquen duraciones desde una fracción de segundo a varios segundos.

6. **Microsonido.** Partículas sonoras perceptíbles hasta los límites de la audición, medidas en centésimas o milésimas de segundo.

7. **Muestreo.** Es el nivel atómico de los sistemas de audio digitales; es decir, conforma las muestras binarias individuales y los valores numéricos de la amplitud. La sucesión de varias muestras ocurre de manera equitativamente distribuida en intervalos medidos en millonésimas de segundo (o microsegundos).

8. **Submuestreo.** Fluctuaciones en una escala de tiempo tan breves que es difícil grabarlas o percibirlas concretamente. Se miden en millonésimas de segundo (nanosegundos) o menos.

9. **Infinitesimal.** La extensión temporal ideal para algunas duraciones matemáticas tales como las funciones delta.

Las escalas de tiempo musicales siempre se interrelacionan (ya sea a conciencia o inconsciencia del compositor). Supra, macro, mesosonido y objeto sonoro son aquéllas que le competían a la música clásica de Bach, Mozart, o Beethoven. Es decir, la composición de piezas de este tipo, mayormente considera tales escalas.

El suprasonido, en mi apreciación, significa la influencia (o huella) que deja una pieza musical en la sociedad. Ya que ésta, una vez completa, permance en la memoria y resuena cada vez que es reintepretada para una audiencia o cada vez que se transmite vía radio. Y es a través de esta comunicación de la música, que otros compositores copian algunas de las formas y patrones musicales, incluyéndolas en sus propias piezas, creando tendencias. Es así que muchas obras contrapuntísticas existen que usan el recurso de la fuga. No por ello las piezas son iguales, aunque sí parecidas.

En la música clásica existe, dentro del método de composición, la consideración de un esquema general, el cual determina las secciones de toda una pieza. Es esto lo que se entiende por macroforma. Un ejemplo es la forma del minueto.

La nota musical es el primordial objeto sonoro, subordinado a las posibilidades del instrumento productor de sonido (registro, maneras de articular, etc).

Todo lo que comprende secuencias de notas, compases y la modificación de sonidos en el tiempo como son las dinámicas, conforman la mesoforma. Digamos, es agrupar varios sonidos, los cuáles conforman las frases.

La evolución de la tecnología permitió, en su dado momento, extender el espectro de composición hacia el microsonido e incluso al muestreo. Era antes imposible, previo a la existencia del ordenador, controlar los parámetros sonoros con toda precisón. En realidad, en la música clásica tradicional, éstos eran más o menos delegados a cada instrumentista y por eso es que cada interpretación de una misma obra suena diferente.

Con la computadora es posible asignar valores numéricos exactos a cada variable de un sintetizador o pieza musical. Por eso ya se habla de la posibilidad de controlar la música desde el objeto sonoro o del microsonido. Objeto sonoro se refiere a una entidad de la cual podemos percibir una duración, amplitud, frecuencia y timbre, pero cuya combinación de valores determinan una identidad audible que el oído no fragmenta. Un ejemplo de esto es el sonido de nota de un clarinete.

El microsonido por otro lado, toma en cuenta aquéllos sonidos de muy poca duración, para los que el oído ya no logra necesariamente distinguir una frecuencia, un ritmo o un volumen. Esta escala también abarca ilusiones acústicas y masas de partículas sonoras. Un oído es incapaz de escuchar cada partícula individual, pero sí logra oír un conjunto de cientos de ellas, así como el efecto estadístico que produce la modificación temporal de las propiedades de cada partícula. Los algoritmos de síntesis granular consiguen pues, que el compositor controle la densidad de partículas o sus duraciones, entre otros. Nos abrimos así paso a nuevas formas de creación musical.

Se le conoce como muestreo al proceso mediante el cual una señal analógica se digitaliza. Para ello, se necesita discretizar la señal analógica. Es decir, un número finito de puntos en la señal son los que sobrevivirán en la versión digital, no obstante el oído escuchará la onda resultante como la misma. Con la computadora se vuelve plausible operar en el nivel de cada muestra, aunque no se suele hacer con el fin de producir composiciones musicales, ya que se requieren miles de muestras para escuchar algo. Una muestra sería equivalente a lo que en la representación computacional de imágenes es un pixel y su amplitud a un color.

El submuestreo sucede a intervalos de nanosegundos, abarcando la información que ocurre entre dos sampleos consecutivos. Nuevamente, esto tiene sentido de conseguirse gracias al teorema de Nyquist, pero sólo a partir de miles de estos datos esta escala ocasionaría un efecto audible.

Las escalas restantes, la infinita y la infinitesimal se pudieran entender como opuestos, y cuya relevancia radica en que en las matemáticas resulta indispensable definir un infinito. Si bien este concepto es muy difícil de imaginar, muchas funciones matemáticas divergen hacia el infinito o contienen infinitos puntos en el infinito. También las estructuras numéricas (números naturales, enteros, racionales, reales o complejos) son infinitas. Es entonces que la escala infinita se refiere a aquélla cosa que es tan grande e incuantificable, como una lista que nunca acaba. Por ejemplo, el $sen(x)$ y $cos(x)$ son funciones que se extienden infinitamente en sus dos direcciones. Podríamos caminar a lo largo de la gráfica del $sen(x)$ hasta morirnos en el camino. Más aún, una transformada de Fourier, cuyo rango de aplicaciones al procesamiento de señales es vasto, se puede expresar como una suma infinita de senos y cosenos.

Una cantidad infinitesimal por otro lado, se suele referir a un fenómeno que se va colapsando hacia un punto (o un estado constante), atravesando las infinitas subdivisiones de una distancia. Podemos decir por ejemplo, que para valores de x mayores a 0, la función $\frac{1}{x}$ converge a 0 cuanto x tiende a infinito. O que es posible aplastar una función constante hacia un punto, de tal modo que su integral sea 1.

Ambas escalas (infinita e infinitesimal), según Roads, deben mencionarse, puesto que la definición de varios de los objetos con aplicaciones sonoras es matemática, y a veces tendremos que describir formalmente en qué consisten.

Esta manera de definir las escalas temporales de la música me parece apropiada para discutir mi música:

Arreglos e improvisación en guitarra - descontando el procesamiento de la señal de audio - serán comprendidos como mesosonido.

Los beats electrónicos abarcarán desde el microsonido hasta el mesosonido, ya que, evidementemente, algunas funciones de *TidalCycles* granulan los samples, tratándose en tal caso, de microsonido. Además, las maneras que este programa tiene de procesar audio son tan diversas que puede el oído pensar que se trata de sonidos aislados o de un gran sonido homogéneo en el tiempo (objeto sonoro). Este sería el caso de controlar un sintetizador de frecuencia modulada vía *TidalCycles*. Mientras tanto, otras funciones en *TidalCycles* son enteramente rítmicas y las identificaremos como arreglos musicales en el mesosonido.

Ejemplos de macrosonido en mi música son: 1. El esquema global con que realicé un remix de una pieza género house. 2. El esqueleto de todo el arreglo percusivo de una pieza que conforma la base de composición de todo lo demás.

Una criba de Xenakis tiene tanto una definición en partitura como una definición matemática. En el primer caso, éstas (las cribas) abarcan del mesosonido al macrosonido. En el segundo caso, la escala infinita parece más adecuada para situar a esas rejillas donde transcurren las cribas, y que se corresponden con la recta de los números enteros y una selección de puntos en ella.

Una fórmula matemática no figuraría dentro de la clasificación dada por Curtis Roads, pero sí la forma de mapearla en la música. A veces lo haría infinitamente; por ejemplo, un sonido de bombo que es ejecutado en *TidalCycles* mediante la función `d1 $ sound ''bd''`, se evaluará infinitamente cada ciclo de tiempo (hasta detener o modificar el proceso). A veces lo haría infinitesimalmente, aunque en este caso, escucharíamos un efecto sonoro de subdivisiones o tal vez de aspereza, y el número de pasos efectuados por la computadora sería finito; de lo contrarío el lenguaje de programación se trabaría.

1.3.7. Iannis Xenakis y la música estocástica

Iannis Xenakis fue músico, ingeniero y arquitecto; pionero de la música electrónica y la computación musical. Alumno de Olivier Messiaen (compositor francés) y profesor de Julio Estrada (compositor mexicano).

En sus varios escritos (el más famoso, *Formalized Music*, de 1971) desarrolló una teoría musical inventada por él, la cual asumía los axiomas de Zermelo-Fraenkel de la teoría de conjuntos; también observó que los objetos sonoros que cumplían la propiedad de ser grupos abelianos eran especialmente útiles. Así mismo dio la definición de criba musical y de música estocástica, e inspirado en el modelo de Gabor de la física cuántica inventó la síntesis granular. Compuso una serie de obras arquitectónico-musicales, *Les Polytopes*, en donde exploró la interacción entre música y arquitectura, resignificando los conceptos de punto, línea, espacio, tiempo y distribución de probabilidad.

Fue pues, una persona preocupada por las teorías matemáticas de su época y fungió como traductor e intérprete entre este lenguaje y la música, buscando llevar las matemáticas al ámbito artístico, siempre a partir de la realización de obras musicales y documentando arduamente sus procesos. Algunos de estos son considerados oscuros e imposibles de dilucidar pero los analistas asiduos ayudan a desentrañar sus razonamientos y explican de mejores maneras la trascendencia de su obra.

A lo largo de su vida compuso aproximadamente 150 composiciones instrumentales, electrónicas, teatrales y electroacústicas.

Cribas Musicales

Xenakis desarrolló su teoría de cribas en el año de 1963, durante una estancia en Alemania. La teoría de cribas se ocupa de la creación de ritmos y nuevas escalas musicales, que se obtienen a partir de combinar clases residuales de números enteros mediante uniones, intersecciones y sumas.

Pueden emplearse cribas sobre variables musicales que puedan ser modelables por números enteros (escalas, duraciones, dinámicas, tempo, etc).

Hay cribas simples y compuestas.

Definición (Criba simple). *Sean* $x, m, b, q \in \mathbb{Z}$ *tales que* $x = mq + b$*. Una* ***criba simple*** *o* ***módulo*** *es una línea discreta y afín cuya expresión es:*

$$(m, x) = \{mq + b, q \in \mathbb{Z}\}.$$

A m *se le llama el periodo y a* x *el representante.*

Son cribas:

- El conjunto X de todos los enteros x tales que $x \equiv 5 \pmod 3$.
- El conjunto Y de los enteros y tales que $y \equiv 5 \pmod 2$.

Definición (Criba compuesta). *Si* X *y* Y *son cribas, entonces* $X \cup Y$*,* $X \cap Y$*,* X^C *y* Y^C *también lo son. Más generalmente, uniones e intersecciones finitas de cribas también son cribas. A este tipo de cribas se les llama* ***compuestas****.*

La forma más usual de criba que Xenakis manejó fue la de intersecar por pares y después unir todas ellas:

$$(A_1 \cap B_1) \cup (A_2 \cap B_2) \cup \ldots \cup (A_n \cap B_n).$$

Los tipos más comunes de cribas musicales que usó fueron las rítmicas y las melódicas.

Composiciones musicales de Xenakis que ejemplifican el uso de cribas son:

Psappha (1975); una pieza para percusiones escrita únicamente con cribas rítmicas. *Tetora* (1990), para cuarteto de cuerdas, es una composición basada en cribas melódicas.

En matemáticas, la criba de Eratóstenes es un algoritmo fundamental, el cual consiste en determinar todos los números primos que hay en una lista de n números naturales consecutivos. Se inicia seleccionando el primer primo, el 2. Luego se recorre de dos en dos la lista de los n naturales y se van removiendo todos los múltiplos de 2. A continuación se selecciona el segundo primo, el 3, y se remueven todos sus múltiplos. Se termina el proceso cuando se llega al mayor primo p tal que $p \leq n$. Los números que se quedan enlistados de 2 hasta p, es decir, los números que no fueron removidos, son todos primos.

En resumen, la criba de Eratóstenes usa varias cribas de diferentes longitudes, que filtran todos los números primos. Mediante este algoritmo, y asociándole sonidos a los números primos, es posible crear una pieza musical de duración infinita, que suene cada vez que aparece un número primo.

Geométricamente, un criba significa una colección de puntos en una línea recta. En palabras de Xenakis:

> "*Todo conjunto bien ordenado puede representarse como puntos en una línea, si damos un punto de referencia para el origen, y una longitud u para la distancia unitaria. Esto es una criba.*"

Según Noll-Andreatta (en [Noll et al., 2006]), las cribas son una abstracción que ya se había manifestado previamente en música. Por ejemplo, la formalización matemática que Xenakis dio de las cribas nos permite analizar cómo las utilizó Alexander Scriabin en su Estudio para piano Opus 65, Tercer Movimiento. Este hecho plantea que una criba encuentra utilidades musicales que van más allá de quien les dio nombre.

Efectivamente, la manera de dibujar geométricamente a las cribas, ritmos euclidianos y acordes, es igual. Mediante gráficas cíclicas, composiciones o descomposiciones entre ellas y colores para sus vértices. Ello significa una eficiente representación computacional sobre la que se pueden realizar operaciones con rápida ejecución.[37]

Síntesis granular

La síntesis granular consiste en dividir una onda sonora en pedacitos de muy corta duración, llamados granos, para luego reordenarlos y escuchar el resultado. Los granos se entretejen para crear sonidos.

Esta síntesis puede ser implementada mediante un sencillo instrumento: un oscilador senoidal siendo controlado por una envolvente (ver Figura 1.4). El mismo que empleó Xenakis.

La noción de "sonido granular cuántico" fue propuesto por el físico británico Dennis Gabor en dos artículos que combinaban sus observaciones en la física cuántica teórica y sus resultados experimentales. De acuerdo a esta teoría, una representación granular era capaz de describir cualquier sonido, hipótesis que fue verificada matemáticamente por Bastiaans. Xenakis en 1960 fue el primero en proponer una teoría compositiva usando granos sonoros.[38] Variantes posteriores de la síntesis granular consisten en: usar formas de onda distintas de la sinusoidal (cuadradas, triangulares, etc). O los denominados métodos síncronos y asíncronos de Rodet y Kaegi. Pero quizá el sistema

[37] Consultar *The geometry of musical rhythm*, en [Toussaint, 2019], *From Polychords to Polya*, en [Keith, 1991], *Computer-aided transformational analysis with tone sieves*, en [Noll et al., 2006].

[38] *The computer music tutorial*, en [Roads, 1996].

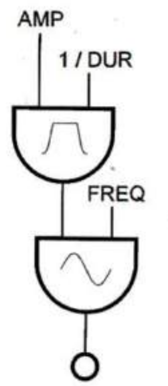


Figura 1.8: Diagrama de síntesis granular.

granular de Curtis Roads, el basado en eventos sonoros llamados nubes es el que podría proporcionar más pistas sobre cómo debe generarse un sonido granular.

En el mecanismo de Roads, algunos de los parámetros que se pueden controlar de una nube sonora son: A) El momento en que inicia y termina la duración de una nube. B) La densidad de los granos. C) La forma de onda de los granos. D) La banda de frecuencias de los granos. E) La amplitud de la envolvente. F) La dispersión espacial.

El UGEN `GrainBuf` de *SuperCollider* permite realizar síntesis granular sobre un archivo de audio guardado en un *buffer*. El código se ve así:

```

GrainBuf.ar(numChannels, trigger, dur, sndbuf, rate, pos, interp,
    pan, envbufnum, maxGrains, mul, add)
```

Entre paréntesis están los 10 argumentos, que son: el número de canales de salida de la señal (`numchannels`), un *trigger* para iniciar nuevos granos (`trigger`), el tamaño en segundos de un grano (`dur`), la ubicación del audio o audios a ser sintetizados (`sndbuf`), la velocidad de reproducción del sonido sampleado (`rate`), la ubicación en donde un grano sonará (`pos`), el cambio de frecuencia de los granos (`interp`), paneo (`pan`), el número de *buffer* de una señal que modulará la envolvente de los granos (`envbufnum`), el número máximo de granos que pueden solaparse (`maxGrains`), la amplitud de la señal (`mul`) y el desfase de la señal (`add`).

Como se puede observar, todas las variables A-F de una nube sonora que menciona Roads los tiene `GrainBuf`. Sin embargo, hay variantes de sintetizadores granulares, por lo que pueden sonar diferentes entre ellos.

Música estocástica

La música estocástica surgió entre los años 1953 y 1955 cuando Xenakis introdujo la teoría de la probabilidad a la composición musical. Primeramente, el cálculo de probabilidades fue usado en *Metástasis* (1954), luego en *Pithoprakta* (1956). Ambas obras orquestales en las que el parámetro aleatorio eran las velocidades que se representan como líneas en un espacio de frecuencia-tiempo.[39]

Analogique A-B fue la primera pieza musical que empleó granulación, e ilustra un puente entre la composición musical y la programación. Se conforma por dos piezas: *Analogique A* (1958) y *Analogique B* (1959), siendo tocadas simultáneamente. La primera de las piezas es orquestal y la segunda, electrónica. Específicamente, *Analogique B* comprende cientos de muestras muy pequeñas de cinta magnética, conformando un total de cuatro pistas que eran reproducidas a través de 8 bocinas ubicadas en lugares estratégicos de una sala de concierto. Ambas porciones de la pieza total fueron compuestas por medio de métodos estocásticos, con una posterior conversión de la parte A a su notación musical.

En *Achorripsis* (1957), ciertos algoritmos estocásticos se aplicaron para generar tanto los parámetros de las notas como la estructura global. La arquitectura de la pieza puede leerse en una matriz 2-dimensional que está definida en un espacio donde siete renglones representan siete grupos de instrumentos que van desarrollándose en el tiempo. En la época en la que surgió esta obra, todos los cálculos estocásticos se hacían a mano, con calculadoras.

En los años 60, Xenakis comenzó a usar el ordenador para automatizar y acelerar muchas de las operaciones que él requería, delegándole a la computadora decisiones que usualmente realizaría el compositor. La obra *ST10* lo ilustra. En los años 70, Xenakis experimentó con métodos de síntesis de sonido en ese entonces nuevos, basados en caminatas aleatorias, que se ven reflejados en la obra "*Le Polytope de Cluny*." Esta también gozó de ser la primera pieza musical, presentada en Francia, que empleó síntesis digital. Posteriormente diseñó un algoritmo para sintetizar sonido, al que llamó "síntesis dinámica estocástica", con el que creó "*La Legende D'Eer*", otra de sus piezas emblemáticas; el componente musical en "*Le Diatope*".

En 1991, Xenakis escribió un programa llamado *GENDY*, que en ese entonces era ejecutable en cualquier PC. Generaba tanto la estructura musical como los sonidos mismos. Cada sonido se sintetizaba vía un recién modificado algoritmo de síntesis dinámica estocástica, el cual generaba diversas familias de timbres sonoros. La única diferencia entre ambas versiones de síntesis dinámica estocástica radica en que la primera empleaba únicamente caminatas aleatorias de primer orden, y la segunda

[39] Ver *The Stochastic Synthesis of Iannis Xenakis*, en [Luque, 2009].

usaba caminatas aleatorias de segundo orden.

Una caminata aleatoria de segundo orden consiste de tres elementos: una distribución de probabilidad y dos caminatas aleatorias. La distribución de probabilidad genera los incrementos de los pasos (en inglés, *step-sizes*) de la primera caminata aleatoria. Las posiciones sucesivas de la primera caminata aleatoria determinan los incrementos en pasos de la segunda caminata aleatoria. Y de las sucesivas posiciones que toma la segunda caminata aleatoria, se obtienen los valores de la caminata aleatoria de segundo orden.

A continuación se analiza una obra de Xenakis y su programa *GENDY*. Estas dos referencias proponen un camino hacia el entendimiento de los métodos numéricos en la música y sugieren vías para diseñar algoritmos propios, cubriéndose un vasto espectro de acción. Desde el microsonido al macrosonido, atravesando la programación y síntesis.

Analogique B

En *Markovian Stochastic Music Theory* y *Markovian Stochastic Music Applications*, de *Formalized Music*, Xenakis refirió tres variables fundamentales que conformaban un *sonido complejo* (así le llamó): frecuencia, intensidad y distribución de densidad. Cada sonido complejo se conforma de *granos*. En palabras de Xenakis:

> "*Un sonido complejo puede ser imaginado como un destello de luces multicolores en las que cada punto de luz aparece y desaparece instantáneamente en un cielo negro. Una línea de puntos puede ser creada por una suficientemente grande multitud de puntos apareciendo y desapareciendo.*"

La propuesta de Xenakis consistió en dar dos configuraciones posibles para cada variable y fijarse en sus combinaciones. Para la frecuencia f, las dos configuraciones son f_0 y f_1. Para la intensidad g, son g_0 y g_1. Y para la densidad, d_0 y d_1 son sus configuraciones posibles.[40]

Ya que los parámetros son tres y las distribuciones ocurren conforme a una escala, esto podría dibujarse en un plano tridimensional. Sin embargo, la gráfica resultante es muy difícil de leer. Así que la lectura se simplifica proyectando uno de los valores hacia las entradas de matrices 2-dimensionales de por ejemplo, 4 renglones (variable intensidad) y 16 columnas (variable frecuencia); definiéndose así, una "pantalla". En cada posible pantalla, sólo 10 entradas de su matriz manifiestan sonido. Es decir, hay 10 nubes granulares por cada pantalla (ver Figura 1.10).

[40] Ver: *Genetic Analysis of Analogique B*, en [Hagan, 2005].

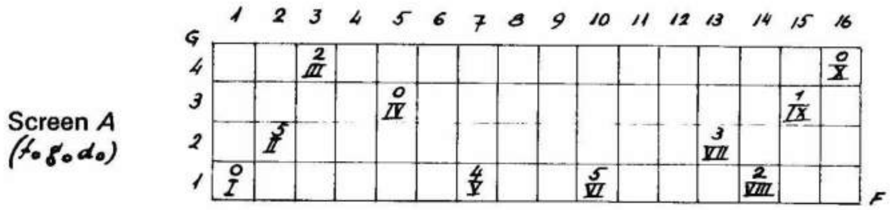


Figura 1.10: Hay 10 nubes granulares por cada pantalla.

Como hay dos configuraciones por cada parámetro, hay un total de $2 \times 2 \times 2 = 8$ pantallas diferentes. Las posibles combinaciones de f, g, d, que Xenakis denotó por A, B, C, D, F, G y H:

$$f_0 g_0 d_0 = \text{Pantalla A}$$
$$f_0 g_0 d_1 = \text{Pantalla B}$$
$$f_0 g_1 d_0 = \text{Pantalla C}$$
$$f_0 g_1 d_1 = \text{Pantalla D}$$
$$f_1 g_0 d_0 = \text{Pantalla E}$$
$$f_1 g_0 d_1 = \text{Pantalla F}$$
$$f_1 g_1 d_0 = \text{Pantalla G}$$
$$f_1 g_1 d_1 = \text{Pantalla H}$$

Un sonido complejo desenvolviéndose en el tiempo, es la concatenación de estas pantallas, y Xenakis propuso ordenarlas por medio de cadenas de Markov, el cual es un método numérico en el que las probabilidades de los eventos precedentes determinan a los consecuentes (ver Capítulo 1, Subsección 1.3.5).

Como ya se vio en la Subsección 1.3.5, una cadena de Markov se puede especificar mediante una matriz de de transición que especifique las probabilidades de pasar de cierto estado a otro. En este caso, la probabilidad de que cierta pantalla suceda a otra.

Cuando el proceso estocástico determinado por la cadena de Markov no se ve afectado por impulsos externos, siempre converge a un estado de equilibrio. Xenakis jugó con ambos casos: cuando hay perturbaciones en el sistema y cuando no los hay.

Xenakis usa dos matrices de transición 2×2 para cada dimensión. Es decir, dos matrices *alpha* y *beta* para la frecuencia, dos matrices *gamma* y *épsilon* para la intensidad; dos matrices *lambda* y *mu* para la densidad. Y para conjuntar las tres dimensiones, Xenakis propuso que los valores de dos de las dimensiones determinaran cuál matriz de transición sería usada para la tercera, lo que quedaría indicado por una tabla. Teniendo esto, el paso final es crear una matriz de transición de 8×8 que calcule la probabilidad de que cualquier pantalla x pase a cualquier pantalla y (ver Figura 1.11).

MTPZ

↓	A $(f_0g_0d_0)$	B $(f_0g_0d_1)$	C $(f_0g_1d_0)$	D $(f_0g_1d_1)$	E $(f_1g_0d_0)$	F $(f_1g_0d_1)$	G $(f_1g_1d_0)$	H $(f_1g_1d_1)$
$A(f_0g_0d_0)$	0.021	0.357	0.084	0.189	0.165	0.204	0.408	0.096
$B(f_0g_0d_1)$	0.084	0.089	0.076	0.126	0.150	0.136	0.072	0.144
$C(f_0g_1d_0)$	0.084	0.323	0.021	0.126	0.150	0.036	0.272	0.144
$D(f_0g_1d_1)$	0.336	0.081	0.019	0.084	0.135	0.024	0.048	0.216
$E(f_1g_0d_0)$	0.019	0.063	0.336	0.171	0.110	0.306	0.102	0.064
$F(f_1g_0d_1)$	0.076	0.016	0.304	0.114	0.100	0.204	0.018	0.096
$G(f_1g_1d_0)$	0.076	0.057	0.084	0.114	0.100	0.054	0.068	0.096
$H(f_1g_1d_1)$	0.304	0.014	0.076	0.076	0.090	0.036	0.012	0.144

Figura 1.11: Matriz de transición para todas las pantallas A-H.

GENDY

GENDY3 y *GENDY301*, ambas obras de 1991, fueron realizadas con el mismo programa computacional *GENDY*. Para una descripción apta del funcionamiento de *GENDY* conviene distinguir su microestructura (timbre estocástico) de su macroestructura (arquitectura estocástica).[41]

Microestructura

La intención de Xenakis fue recrear la variedad, riqueza, vitalidad y energía que vuelven a un sonido musicalmente interesante. Para Xenakis, la síntesis de Fourier, misma que se basa en apilar componentes de ondas senoidales para luego variar aperiódicamente en el tiempo parámetros como la frecuencia y amplitud de los armónicos, era inadecuada para lograr los sonidos que él deseaba. Así pues, efectuó un tipo de síntesis por descomposición espectral por métodos estocásticos, llevado a cabo desde el dominio del tiempo, que evidentemente difiere del análisis de Fourier. La síntesis estocástica dinámica consiste en usar varios tipos de caminatas aleatorias para sintetizar sonido. Se asigna a la posición de cada partícula la amplitud de un sampleo de sonido. La varias partículas se mueven aleatoriamente sobre un eje. Se añaden barreras elásticas (también llamadas absorbentes) para controlar las posiciones aleatorias de las partículas. El programa *GENDY* calcula una serie de sampleos numéricos y los

[41]Ver *Stochastic Composition and Stochastic Timbre: GENDY3 by Iannis Xenakis*, en [Serra, 1993].

almacena en un archivo de sonido que puede ser reproducido después de compilarse el programa. La amplitud de un sampleo es igual a la suma de las amplitudes dadas por las varias voces que la conforman.

Un conjunto de ciertos parámetros de entrada caracterizan una voz. Ahí se incluyen los parámetros de síntesis estocástica que controlan el sonido. Existen hasta 16 voces en una pieza producida por el programa. El modelo de síntesis estocástica dinámica consiste en generar una onda inicial y repetirla. Con cada repetición, la forma de onda sufre una modificación en tiempo y amplitud, a partir de aplicarle variaciones estocásticas a la onda anterior (ver Figura 1.12). La forma de onda está poligonizada; es decir, está cortada en segmentos, de donde cada uno de ellos queda determinado por las coordenadas de sus extremos. Sólo los extremos de los segmentos son sometidos a variaciones estocásticas.

El proceso de síntesis produciría una señal muy ruidosa, o ruido blanco, si las fluctuaciones estocásticas no se delimitaran a un intervalo cerrado. Es decir, si la modificación de la onda fuera demasiado distanciada con cada repetición, existiría muy poca o ninguna similaridad entre sucesivas ondas sonoras. Por lo cual es necesario balancear estabilidad (repeticiones con transformaciones débiles) e inestabilidad (repeticiones con transformaciones fuertes). Así pues, el programa forza a que, tanto los valores estocásticos como las coordenadas de los extremos de los segmentos se mantengan en intervalos delimitados; lo que se consigue mediante un procedimiento llamado espejo. Se trata de una función que toma 3 argumentos: un valor de entrada y dos amplitudes límite, y devuelve un valor situado en medio de éstas. Las bandas espejo también se aplican directamente sobre los parámetros del sonido, tales como la frecuencia, amplitud y timbre.

Una caminata aleatoria de primer orden con barreras elásticas se comporta de manera muy diferente que una caminata aleatoria de segundo orden con barreras elásticas, ya que la primera oscila alrededor de un punto de equilibrio que cambia arbitrariamente conforme al tiempo, mientras que una caminata aleatoria de segundo orden gravita alrededor de una de sus dos bandas elásticas (cuál barrera sirve como centro queda determinado por el signo de la primera caminata aleatoria).

Macroestructura

La estructura de una pieza salida de GENDY puede pensarse como un espacio 2-dimensional modelable por una matriz, donde el tiempo es el eje horizontal, y el eje vertical sirve para la organización de las diferentes voces. Ello se parece mucho a lo planteado desde *Achorripsis*, donde los grupos instrumentales se ordenan sobre el eje vertical, y el eje del tiempo queda subdividido en secciones (columnas).

Del mismo modo, *GENDY3* consiste de una yuxtaposición de voces por secciones.

Sobre el eje del tiempo, cada sección abarca una secuencia de intervalos de tiempo (en inglés *time-fields*) donde intervendrán las voces. Sobre el eje vertical, cada sección queda definida por una cierta configuración de voces, la cual queda especificada por el número de voces sonando, su ordenamiento en el eje vertical, y el conjunto de parámetros que sintetizan cada voz.

Un intervalo temporal (o campo temporal) queda definido a partir de dos parámetros: duración e indicación de sonido o silencio. El tiempo en que inicia cierto campo temporal es el mismo en donde termina el campo temporal previo.

Se le llama ensayo de Bernoulli al tipo de experimentos aleatorios que tienen dos posibles resultados. Por ejemplo: Éxito o Fracaso. Cara o Cruz. 0 o 1. A partir de realizar un ensayo de Bernoulli la computadora decide si en cierto intervalo de tiempo dado, un sonido será o no, escuchado.

Conclusiones de la obra de Xenakis para esta tesis

Razones por las que se destaca la importancia de la obra de Xenakis son:

El formato de sus obras (las arquitectónicas como las electrónicas) aún prevalece. Es pues, un antecedente a las instalaciones sonoras y prácticas multidisciplinarias contemporáneas.

Es un antecedente a la programación musical que conocemos actualmente.

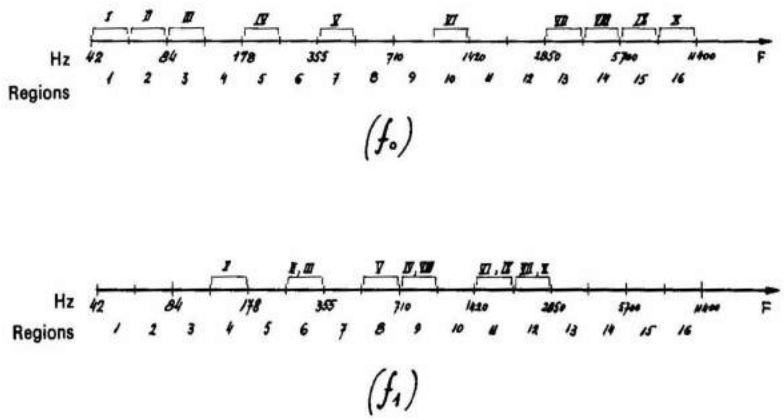
Hay una filosofía detrás de su obra más allá de la aplicación de meros algoritmos.

Fue el primero en matematizar la música (por medio de axiomas, teoremas y teorías).

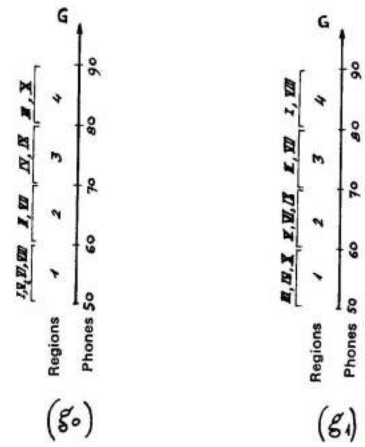
Xenakis y su obra pueden servir como un puente entre el diálogo de músicos y científicos.

¿Considerar la programación musical y la música matemática independientemente de Xenakis? Desde un punto de vista matemático que busca aplicar modelos matemáticos en la música, hay mucho más material que usar del mundo de las matemáticas puras que el que Xenakis trabajó. Consideremos que Xenakis tenía formación matemática, lo que naturalmente derivó en que realizara las composiciones que hizo. Podemos entonces, pensar como Xenakis y buscar estudiar las matemáticas para luego aplicarlas, sin ni siquiera necesitar conocer a este compositor. Opino que es posible entender fundamentos de la obra de Xenakis (como los mencionados en este capítulo) sin necesidad de conocer el lenguaje musical.

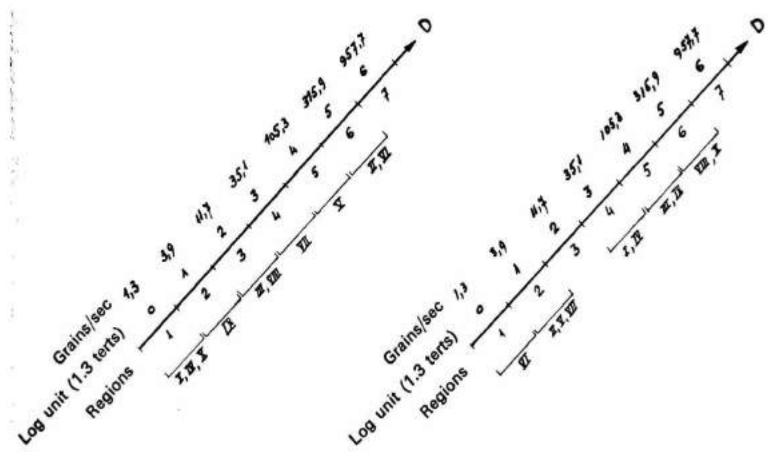
Desde un punto de vista humanista, Xenakis sí es importante en la música y a veces la gente debe estudiar su obra, lo que incluye conocer sus trabajos en matemáticas, música o arquitectura. Históricamente se destaca la importancia del compositor en tanto que juntó música con matemáticas; de tal manera que, para quien compartiera este mismo interés (la conjunción de estas dos disciplinas) debería ser obligado consultarlo como referencia. Ello podría facilitar algunas conclusiones y entendimientos, en particular lo que concierne al cómo aplicar las matemáticas a la música (no al revés).



(a) 2 configuraciones f_0 y f_1 para la frecuencia.



(b) 2 configuraciones g_0 y g_1 para la intensidad.



(c) 2 configuraciones d_0 y d_1 para la densidad.

Figura 1.9: Configuraciones para f, g, y d en *Analogique B*

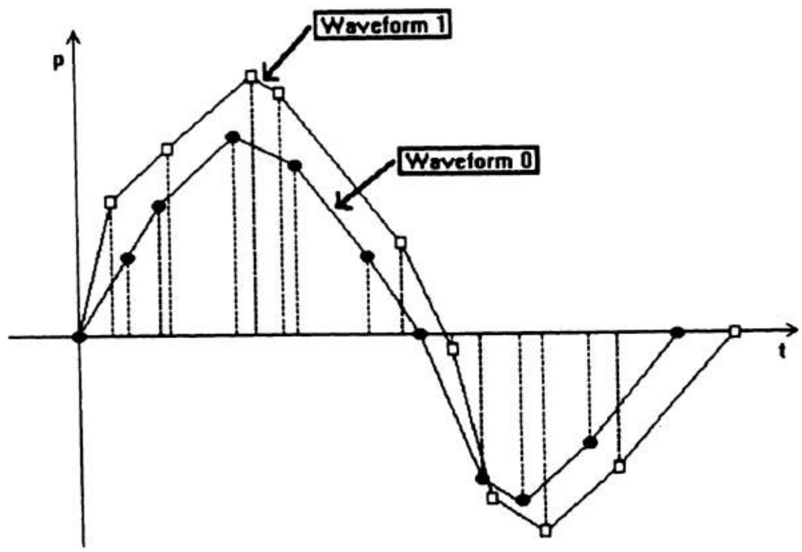


Figura 1.12: Una onda sonora (Waveform 0) y la que resulta de aplicarle variaciones estocásticas (Waveform 1).

2 Herramientas computacionales y de modelación algorítmica

Esta sección cubre una selección de temas que tratan la programación de música y síntesis.

En la Sección 2.1, *Notación musical*, se discute cómo traducir una partitura a un código (o cómo interpretar una partitura en código), o si acaso el código llega a ser una extensión de ésta. Se inicia con la discusión de un plano cartesiano siendo una generalización de una partitura, y de cómo un plano cartesiano es un objeto programable.

La subsección de combinatoria discute las cadenas, collares y particiones, que sirven para respectivamente contar escalas, acordes e intervalos musicales. La bibliografía para esta parte es [Keith, 1991] y [Mayer, 2019].

La subsección de síntesis de sonido repasa la síntesis aditiva, sustractiva, AM, FM y granular, donde los libros de referencia principales son [Shepard, 2013] y [Loy, 2007b].

La subsección de ritmos euclidianos explica una manera de explorar ritmos con *live coding* y la referencia bibliográfica es [Toussaint, 2019], [Bravo et al., 2006] y [Louridas, 2017].

2.1. Notación musical

La asociación entre música y pentagrama puede entenderse como una analogía entre una función matemática de una o varias variables y su gráfica en un plano cartesiano (ver Figura 2.2).[1]

Para verlo mejor, la Figura 2.1 muestra un instrumento musical que está produciendo notas, mismas que se han transcrito al pentagrama. Sobre el eje horizontal del

[1] Consultar el Capítulo 1 de esta tesis, el panorama de la música algorítmica, para una discusión más detallada de esta relación desde una perspectiva de las humanidades.

pentagrama se marca la duración de cada nota emitida mientras que el eje vertical especifica la altura de cada una de ellas. La duración está marcada por la notación usual en notas blancas, negras, corcheas, semicorcheas, puntillos, etc, y la altura se ubica sobre y entre las líneas del pentagrama. La Figura 2.2 ilustra cómo a una función que asocia ciertos valores de un conjunto A, con ciertos valores de un conjunto B, se le puede graficar en un plano cartesiano mediante parejas ordenadas. Cómo sólo hay dos variables que intervienen, este es el caso más simple y se puede identificar con una gráfica en $\mathbb{R}^2$.

Figura 2.1: Una melodía siendo transcrita a pentagrama

Un caso sutilmente más complicado es el el siguiente: la notación musical que en el pentagrama indica un aumento progresivo del volumen del instrumento, agrega una variable más a su función matemática. Esta ya no se puede graficar en un plano sino en un espacio tridimensional.

Cada parámetro sonoro extra del instrumento agregaría una dimensión más en su respectiva función. Es así que, en general, estaríamos hablando de relaciones n-dimensionales viviendo en espacios de Hilbert.[2]

Un espacio de Hilbert es la generalización de espacio euclidiano. Es la propiedad de ser métrico (la posibilidad de medir distancias entre puntos) lo que permite que un espacio de Hilbert sirva como modelo apropiado de un *score*. Es decir, el plano $\mathbb{R}^2$ como el de la figura 2.2 es espacio de Hilbert, y $\mathbb{R}^3$, $\mathbb{R}^n$ con $n \in \mathbb{N}$ también lo son. Por lo tanto, en una situación dada, se puede tomar un número tan grande como se quiera de variables musicales, para que éstas definan con toda precisión algún comportamiento musical deseado.

Pero aterrizando al tema de visualizar funciones musicales específicas para poder trabajar con ellas de manera práctica. Lo que entendemos por un acorde sólo podría ser una función si consideráramos a todo el sonido producido y dejáramos como paso previo independiente la construcción de la polifonía. Si tal fuera el caso, ésta podría interpretarse como una función de $\mathbb{R}^3 \longmapsto \mathbb{R}$, en la que a tres notas distintas se les

[2]Un espacio de Banach es un espacio vectorial normado que es completo. Un espacio de Hilbert es un espacio vectorial con producto interno que es un espacio de Banach.

asigna un punto en el tiempo de otra función que va registrando varios acordes en el Tiempo $\times$ Frecuencia (ver Figura 2.3).

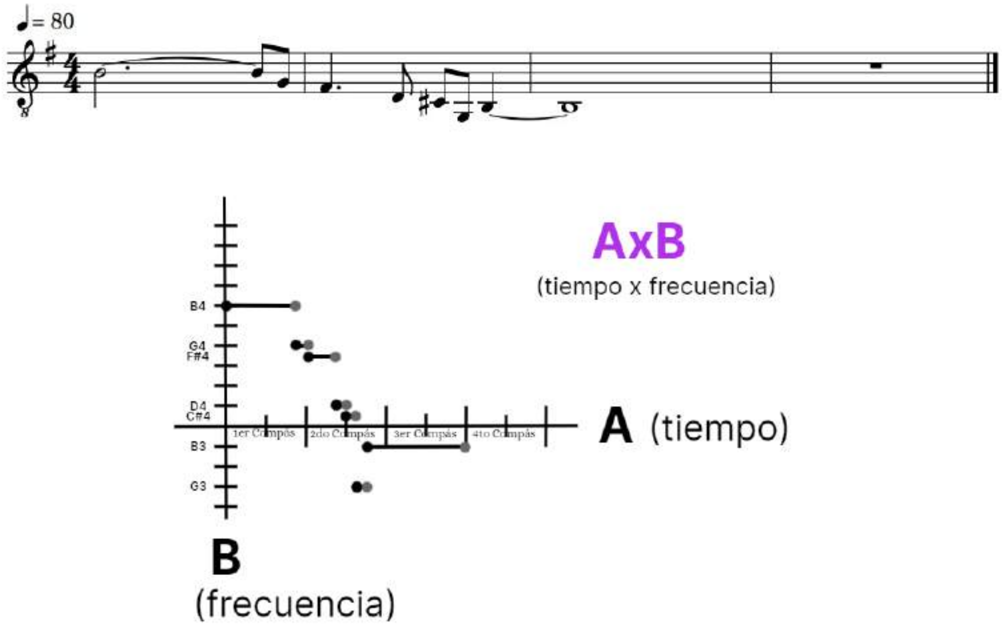


Figura 2.2: La traducción de la información que da una partitura, a un plano cartesiano.

En otras palabras, se tiene una composición de funciónes $g \circ f : \mathbb{R}^3 \longmapsto \mathbb{R}^2$.

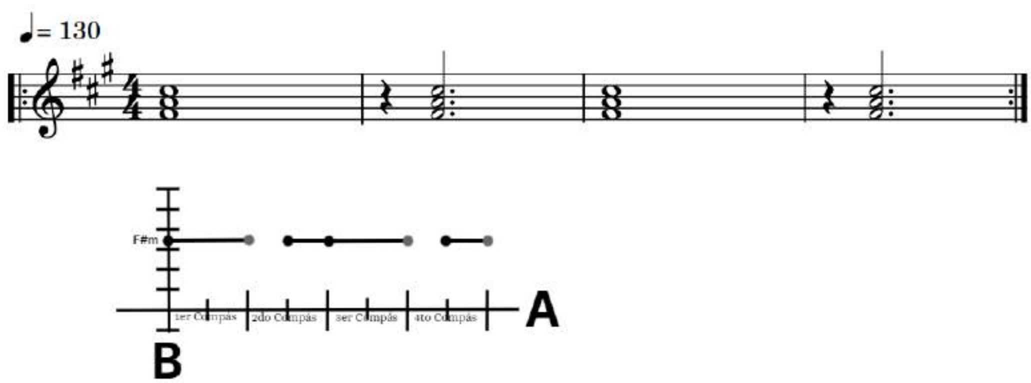


Figura 2.3: Graficando acordes en el tiempo.

El objetivo de imaginar estos modelos para un entendimiento conceptual puede o bien coincidir o bien diferir al implementarlo en un lenguaje de programación, pues una visualización geométrica es informal; sólo es una abstracción mental clarificadora de una idea. Programar desde cero lo anterior requeriría importar una biblioteca que produjera sonido, luego la programación de una rutina, y aún así habrían varias maneras de hacer sonar acordes (esto dependería del paradigma de programación, su interfaz visual, las bibliotecas elegidas o de la forma de escritura). Buscando optimizar el código y reducirlo a la mínima cantidad de caracteres, el programador seguramente se plantearía crear una clase que generara acordes. En *SuperCollider* esto ya está hecho. Sería muy directo realizarlo usando el pattern "`Ptuple`":

```
(
var a, b, c;
a = Pseq([0, 2, 4, 5, 7, 9, 11, 12], 1);
b = Pseq([4, 5, 7, 9, 11, 12, 14, 16], 1);
c = Pseq([7, 9, 11, 12, 14, 16, 17, 19], 1);

p = Pbind(
  \scale, #[0, 2, 4, 5, 7, 9, 11, 12, 14, 16, 17, 19],
      \degree, Ptuple([a, b, c], inf),
).play;
)
```

Secuencia de acordes en *SuperCollider*.

Un patrón de *SuperCollider* encapsula una rutina, por lo que le evita al usuario tener que escribirla nuevamente en cada programa que requiera ejecutar acordes. Así pues, en el código anterior, todo se reduce a una muy breve sintaxis:

```
    Ptuple([a, b, c], inf).
```

Que se corresponde con la intuición geométrica: `Ptuple` es una función que acepta como *input* dos o más secuencias numéricas y las va tocando simultáneamente de acuerdo a su índice en un *array* (indicado en el primer argumento de `Ptuple`). El segundo argumento de `Ptuple`, "`inf`", especifica el número de veces que se evaluará el primer argumento; infinitas en este caso.

Cualquier usuario es bienvenido a programar desde cero una rutina que toque acordes, creando su propia biblioteca si así lo desea; aunque los ya implementados *Patterns* son una aceptada convención de *SuperCollider* a la que recurren muchos usuarios de este programa, quienes suelen optar por extender esta biblioteca en vez de sustituirla toda (a menos que, precisamente, el ejercicio requerido consista en crear una clase que toque acordes e implemente algunos métodos de secuenciación). Una rutina que yo escribí, que suena acordes y no usa *Patterns*, es la siguiente:

```


(
r = Routine({
    var delta, cmaj, minT, maxT;

  cmaj = [60, 62, 64, 65, 67, 69, 71, 72];
  minT = [64, 65,67, 69, 71, 72, 74, 76];
```

```
 maxT = [67, 69, 71, 72, 74, 76, 77, 79];

    loop {
        delta = rrand(1, 3) * 0.5;
[Synth(\a, [freq: cmaj.midicps, amp: rrand(0.1, 0.3), sustain: delta]),
 Synth(\a, [freq: minT.midicps, amp: rrand(0.1, 0.3), sustain: delta]),
 Synth(\a, [freq: maxT.midicps, amp: rrand(0.1, 0.3), sustain: delta])
];
        delta.yield;
    }
});
)
```

Rutina de *SuperCollider* que genera acordes.

Se observa que ésta sólo toca la armonía de Do Mayor en una lista ordenada y sería un lío modificar este código en tiempo real. Habría que crear métodos que mejoraran y optimizaran los cambios plausibles para esta rutina. Podría idear transponer las triadas, cambiar el tipo de intervalos sonando simultáneamente o las escalas utilizadas, entre otras cosas. Además, un método deseable sería generar automáticamente cuantas instancias de rutinas fueran necesarias. Pero así como a mí se me ha ocurrido eso que es tan natural y recurrente para un músico, así otras personas previamente ya lo anticiparon. De este modo, es preciso que exista la clase de los *Patterns* en *SuperCollider*, una intuitiva biblioteca que contiene y flexibiliza las posibles construcciones y modificaciones que pueden efectuársele a una secuencia de sonidos. También, en el código vemos que la información correspondiente a una escala músical está almacenada en un *array*:

```
cmaj = [60, 62, 64, 65, 67, 69, 71, 72]
```

`cmaj` es una variable que almacena el conjunto de notas `do4, re4, mi4, fa4, sol4, la4, si4, do5`, mismas que conforman la raíz de cada acorde. Las variables `minT` y `maxT`, respectivamente, guardan las notas de terceras y quintas en cada acorde.

En los lenguajes de programación, los *arrays* (en español "arreglos"), son una lista de cosas varias como números, palabras, funciones, cadenas, sonidos, variables, etc, y en las que se pueden efectuar operaciones como: agregar nuevos elementos y quitar elementos al arreglo, sumar arreglos, obtener un elemento específico del arreglo, seleccionar subconjuntos de elementos, rotar elementos del arreglo, obtener su tamaño y longitud, promediar, entre otras cosas. Los *arrays* se diferencian de los conjuntos matemáticos en que los primeros están ordenados mientras que los segundos no. En este sentido, los arreglos se parecen más a las matrices. Usar elementos indizados es más práctico para programar, y éstos fungen como las colecciones. Es decir, matemáticamente denotaríamos por `{do4, re4, mi4, fa4, sol4, la4, si4, do5}` al conjunto que tiene esas 8 notas, pero en el lenguaje de programación, para una mayor eficiencia sustituiríamos los corchetes por los corchetes cuadrados.

¿Un programa que ejecuta acordes como los anteriores puede transcribirse a partitura?

Sí. Los ejemplos que presenté cuidan el hecho de que las notas usadas se pueden localizar en el pentagrama y el *tempo* también puede codificarse en tiempos de cuartos y octavos de nota. Bajo estos criterios realmente no importa qué instrumento esté produciendo música (en este caso, la computadora).

Sin embargo, *SuperCollider* es capaz de generar notas cuyas frecuencias se encuentren entre dos sonidos consecutivos de la escala cromática. Eso parece más difícil de graficarse, pero no es imposible. Una solución es: en vez de recurrir al pentagrama, pasar la información musical a un plano cartesiano. El eje x medirá el tiempo y el eje y las frecuencias. El eje y usará la escala de números reales en vez de la de los enteros; de tal modo, puede subdividirse la distancia entre dos notas y así encontrar las frecuencias que en medio se encuentren, con suficiente precisión. Aquí es importante mencionar que mi visión matemática me sugirió este camino, pero en la historia de la música, la notación musical es sujeto de mucha creatividad.

A lo largo de la historia de la música, compositores como Stockhausen, Nancarrow, Tudor o Xenakis, plantearon notaciones alternativas cuyo objetivo fue el de escribir música para la que el tradicional pentagrama no servía. En el caso de Stockhausen, fue con el objetivo de describir el sonido producido por instrumentos electrónicos (por ejemplo, los de la pieza *Kontakte*). Las pianolas de Nancarrow traducían hacia rollos de papel la música producida. Las partituras de David Tudor pretendieron dibujar la aleatoreidad y detalle de glissandos e instrumentación simultánea con las mismas características. Las piezas de Xenakis mostraban la evolución del movimiento browniano que ejecutaba una orquesta o una computadora.

La manera de graficar la música podía o no estar inspirada en el pentagrama, siendo pues, una variación de éste (Tudor y Xenakis son un buen ejemplo) o en el otro caso, había un dispositivo intermediario que registraba los sonidos producidos mediante sus señales eléctricas.

Ahora que existen gran variedad de lenguajes de programación musical, para Curtis Roads, un *score* de música puede significar o bien una partitura, un código, ¡o hasta un diagrama de flujo! Lo que nos lleva a la pregunta de: ¿es siempre necesario escribir la música en un pentagrama?

No siempre lo es. Depende del contexto musical y su para qué.[3] La motivación princi-

[3] *Escritura musical prescriptiva y descriptiva en música*, en [Seeger, 1958] ahonda en este tema.

pal para los músicos electrónicos de la primera mitad del siglo XX fue la exploración sonora, aunque ésta en sus inicios no pudiera escribirse o reinterpretarse (algo similar sucede con los *live coders* actualmente).

La notación tradicional en pentagrama sirve para que un instrumentista u orquesta ejecuten una composición preestablecida y para que el músico produzca una memoria musical. Es el lenguaje usual para su profesión.

Hay veces en que el interés de la creación musical consiste en improvisar. Según el tipo de improvisación en idiomática o libre, puede haber un previo acuerdo escrito entre los músicos. Los estándares de jazz son una alternativa que media entre lo escrito y lo espontáneo. Aquí hay un esquema armónico global predeterminado sobre el que se puede realizar una improvisación, cuya partitura consiste de compases, acordes y algunos ostinatos, pero de ningún modo hay frases específicas que ir leyendo en cada momento de la música. Más aún, Derek Bailey afirma que es imposibile denotar toda idea musical en partitura y que muchas veces, sus objetivos como pianista quedan alejados de tener que transcribir la música.

En la actualidad, es posible expresar el mundo sonoro mediante diferentes sintaxis, y pensándolo muy ilustrativamente a través de la cantidad de lenguajes de programación musicales que hay. La variedad de notaciones conlleva que sea menos entendido para todos lo que una cierta sintaxis significa e imposibilita el entendimiento colectivo de un *score*. Es apto bajo ciertas circunstancias en que hay un solista, o que el acuerdo musical es aportar sonido a escena independientemente de que otros lo comprendan. En otras situaciones se tiene que recurrir a la notación tradicional en pentagrama para que un instrumentista logre ejecutar su porción de una pieza en una composición.

La variedad de sintaxis provee libertad, abre camino al programador para producir música y es incluso fácil que un no músico pueda hacer también música. Por todo esto, yo abogo por la diversidad presentaciones para un *score* musical, tal como Roads plantea. Aunque siempre sin perder de vista posibles reglas de composición, modelos globales que fijan la música y la posibilidad de analizar sus elementos rigurosamente.

2.2. Combinatoria musical

2.2.1. ¿Qué son la combinatoria y la combinatoria musical?

El área de la combinatoria musical abarca composición algorítmica, teorías de la música generativa, programación, problemas de conteos de escalas, acordes e intervalos, métodos aleatorios de creación de obras, entre otros. Se puede trazar el origen de ésta práctica hacia la Edad Media, con la obra "*Ars Combinatoria*" de Ramon Llull, quien dio nombre a los juegos musicales con dados, los "*Musikalisches Würfelspiel*", con posteriores composiciones de Kirnberger, Haydn o Mozart. Otro exponente combinatorio importante es el *Tonnetz*, ideado por el matemático Leonhard Euler.

La combinatoria es un área de las matemáticas que principalmente consiste en contar elementos. Trata problemas del tipo: ¿cuántas maneras hay de ordenar ciertos objetos? O ¿cómo elegir ciertos subconjuntos de una colección?

Tales colecciones, en un contexto musical pueden significar notas, sonidos, acordes, ritmos, arreglos, secciones musicales o variables.

M. Keith, en el libro "*From Polychords to Pólya: Adventures in Combinatorics*" (en [Keith, 1991], pp. 15 - 38) analiza tres problemas de conteo que establece como fundamentalmente importantes para analizar la combinatoria musical. Estos son:

- Cómo contar el número de n-cadenas m-coloreadas, asumiendo que conocemos cuántos vértices hay de cada color.
- Cómo contar el número de n-collares m-coloreados, asumiendo que conocemos cuántos vértices hay de cada color.
- Cómo contar el número de particiones de n en exactamente k partes.

Las aplicaciones de estos tres problemas se dan haciendo las siguientes asociaciones:

- Contar cadenas equivale a contar escalas musicales.
- Contar collares equivale a contar acordes.
- Contar particiones equivale a contar conjuntos de intervalos.

El análisis de Keith ya ha sido aplicado a gran variedad de trabajos académicos, como son los de F. Ruskey y J. Sawada (1991, en [Ruskey and Sawada, 1999]), J. Haack (1991, en [Haack, 1991]), D. Benson (2006, en [Benson, 2006]) G. Toussaint (2013, en [Toussaint, 2019], pp. 74- 83), M. Mannone (2018, en [Mannone et al., 2018]) o C. Nachtmann, D. Mayer y G. Nierhaus (2015, en [Nierhaus et al., 2015]).

2.2.2. Cadenas, collares y particiones

Definiciones

Definimos una "n-cadena" (o simplemente cadena) como una trayectoria (conexa) coloreada en sus vértices, donde n refiere al número total de vértices de la cadena y m al número de colores. Ejemplo: en la Figura 2.1, ambas cadenas tienen $n = 7$ vértices y $m = 3$ colores.

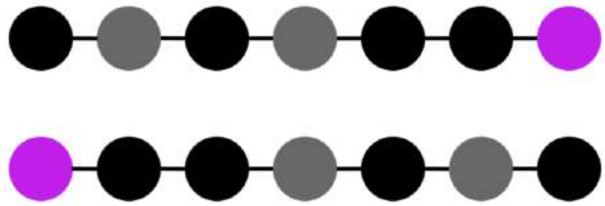


Figura 2.4: Dos cadenas con 7 vértices y 3 colores, inversas entre sí.

Dos n-cadenas serán "isomorfas" (es decir, serán consideradas la misma) si y sólo si cada vértice en correspondientes posiciones de las n-cadenas tiene el mismo color. Observamos que invertir el orden de una n-cadena suele ser diferente de la cadena original, a menos que la cadena original sea un palíndromo.[4]

Figura 2.5: Una cadena que es un palíndromo.

Definiremos un "n-collar" como una gráfica cíclica coloreada en sus vértices.

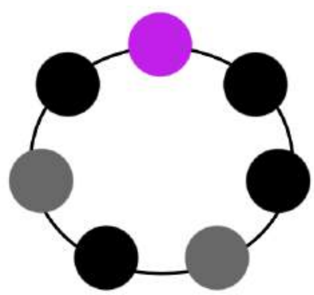


Figura 2.6: Una collar de 7 vértices y 3 colores.

Diremos que dos n-collares son isomorfos si uno puede rotarse hasta hacer corresponder cada uno de los colores de sus vértices con el otro collar.

Un collar y su reflejado no son isomorfos típicamente. Para incluir este caso en algún estudio pertinente, que sí los hay (ver, [Toussaint, 2019], capítulo 14), se utiliza el término "pulsera" (en inglés, *bracelet*). Dos pulseras son isomorfas si una de ellas

[4]Un ejemplo de una aplicación musical: dar cierta colección de compases que funcione como palíndromo. La programación sería así: a un conjunto de notas (submelodía de una melodía) se les asocia un vértice, y hay tantos colores como melodias distintas. La sucesión $(v_0, v_1, ..., v_n)$ muestra la ejecución en el tiempo de cada melodía. Una melodía palindrómica se codificaría de tal manera que la trayectoria resultante fuera un palíndromo.

puede reflejarse y/o rotarse de tal modo que exista una correspondendencia entre los colores de ambas pulseras.

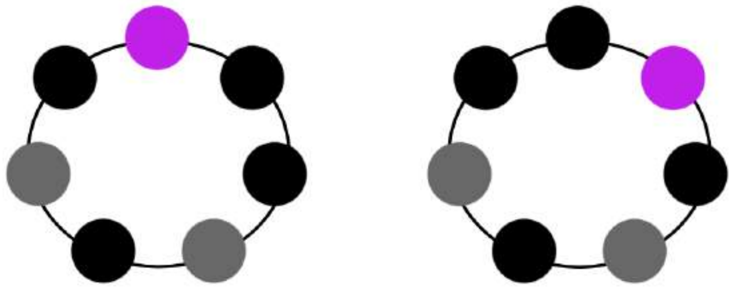


Figura 2.7: Una pulsera y su reflejada son isomorfas.

Finalmente, una "partición de un entero n" es una expresión de n en términos de sumas de enteros menores o iguales a n, y donde el número de sumandos se denota por k. Por ejemplo, $12 = 4 + 3 + 2 + 1 + 1 + 1$. En este caso $n = 12$ y $k = 6$.

Cómo contar cadenas

El problema más sencillo es el de contar cadenas, lo que, en muchas situaciones es reducible a un problema binario de cadenas con dos colores, y sobre el que se aplica la fórmula de las combinaciones:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}.$$

Pero el caso para las n-cadenas m-coloreadas con $m \geq 2$ no es tan complicado. Simplemente usa una generalización de la anterior fórmula:

$$\binom{n}{k_1, k_2, \ldots, k_m} = \frac{n!}{k_1!k_2! \cdot \ldots \cdot k_m!}.$$

Ejemplos.

- El número de combinaciones que hay de los dígitos 1,0,1,0,0 es
$$\binom{5}{2} = \frac{5!}{2!3!}.$$
- El número posible de caminos que pueden tomarse en una rejilla 2×2, donde los únicos movimentos permitidos son DERECHA y ARRIBA es
$$\binom{4}{2} = \frac{4!}{2!2!}.$$

- El número de arreglos DIFERENTES que pueden formarse con las letras de la palabra *PARANGARICUTI* es
$$\binom{13}{3,2,2} = \frac{13!}{3!2!2!}.$$

Los anteriores son problemas típicos de conteo y notemos que es totalmente equivalente hablar de contar el número de arreglos de cadenas coloreadas al de contar el número de arreglos de letras diferentes. Por ejemplo, para determinar cuántos arreglos se pueden formar con las letras de la palabra *PARANGARICUTI* primero debemos contar cuántas permutaciones hay para esa palabra. Son 13!. Luego necesitamos contar cuántas letras se repiten: hay 3 A's, 2 R's y 2 I's. Para quitar aquéllos casos en que se intercambian las A's entre sí (y respectivamente, las R's o las I's) hay que dividir entre 3!2!2!. Ahora, si en vez de letras se cambiaran éstas a colores, se obtendría una 13-cadena de 9 colores, donde un primer color se repitiría 3 veces, un segundo color se repetiría dos veces, un tercer color se repetiría 2 veces y el resto de los colores aparecería una sola vez en la cadena. Por lo tanto, el número de arreglos diferentes que pueden formarse para esta 13-cadena es también $\frac{13!}{3!2!2!}$.

Observación. Al realizar cálculos combinatorios, sólo a veces es requerido enlistar todos los elementos que se piden, pero en realidad, éstas fórmulas no dan una manera de enlistar elementos. Si se pretende que un programa de música liste arreglos específicos basados en combinaciones o permutaciones, se necesita implementar un algoritmo de búsqueda, recursivo, que ordene elementos léxicográficamente.

Cómo contar collares

Contar collares es mucho más difícil que contar cadenas, y para lo cual es necesario recurrir al "Teorema de Pólya" de G. Pólya (1887-1985). Éste sirve para contar cadenas y collares, donde el conteo de collares es visto como un problema más general que el de contar cadenas.

> **Teorema de Pólya:** *Tenemos n urnas y m colores diferentes. El número de configuraciones con peso $k_1w_1 + \ldots + k_mw_m$, donde dos configuraciones son consideradas equivalentes bajo cualquier permutación en un grupo de permutación G, es igual al coeficiente de $w_1^{k_1}w_2^{k_2}\cdot\ldots\cdot w_m^{k_m}$ en el polinomio*
> $$\frac{1}{|G|}P(G; \sum w_i, \sum w_i^2, \ldots, \sum w_i^n).$$

El teorema de Pólya sirve como un auxiliar en las siguientes fórmulas, que son las que realmente se ocupan para contar collares:

$$N(n,k) = \frac{1}{n} \sum_{d|n,k} \phi(d) \cdot \binom{n \mid d}{k \mid d},$$

$$N(n,k_1,\ldots,k_m) = \frac{1}{n} \sum_{\substack{d|n \\ d|k_i (1\leq i\leq m)}} \phi(d) \cdot \binom{n \mid d}{k_1 \mid d, k_2 \mid d, \ldots, k_m \mid d}.$$

Cómo contar particiones

Una partición de un entero n es una expresión de n como una suma de k enteros positivos menores o iguales a n. Por ejemplo, $11 = 4 + 4 + 3$, es una partición de 11 en $k = 3$ sumandos.

Contar particiones es un problema mucho más difícil que el de contar collares. Su solución a manera de una fórmula explícita la dieron G.H. Hardy y S. Ramanujan (en 1918). Pero es la fórmula recursiva la que se suele utilizar en computación. Definimos $q(n,k)$ como "el número de particiones de n que son menores o iguales a k." La fórmula es la siguiente:

$$q(n,k) = q(n-k,k) + q(n,k-1).$$

Para mayores detalles respecto a la derivación de estas fórmulas que cuentan cadenas, collares y particiones se sugiere consultar [Keith, 1991].

Recursión

Las expresiones recursivas como la anterior para particiones son comunes en computación. Una fórmula recursiva es aquélla expresión $f(n)$ que se define en términos de sumas o productos de la misma función f, pero ésta siendo evaluada sobre términos anteriores. Es decir, primero se tiene que calcular la expresión f sobre el término o términos precedentes a n, para poder deducir $f(n)$. Un ejemplo de fórmula recursiva sencilla es la serie de Fibonacci, definida por

$$f(n) = f(n-2) + f(n-1), \quad \text{con} \quad f(0) = 0 \text{ y } \quad f(1) = 1.$$

Otro ejemplo (de fórmula recursiva) es "el" algoritmo que programa combinaciones en un lenguaje computacional. Es un árbol que comienza con un arreglo vacío y que va preguntándo en cada paso si un sucbonjunto tiene a cierto elemento. De un lado del árbol figuran todas las combinaciones que contienen al primer elemento ("a") y del otro lado están todas las combinaciones que no contienen al primer elemento ("a").

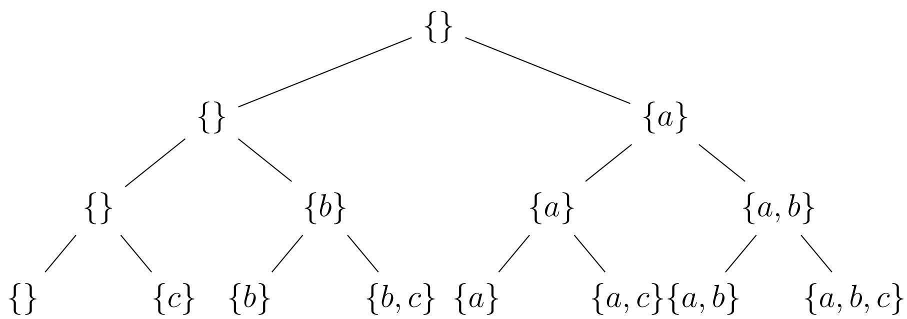


Figura 2.8: Árbol de combinaciones.

Enumeraciones en *SuperCollider*

El método `enum` de *SuperCollider* es una extensión de *SuperCollider*, desarrollada por D. Mayer, que implementa varios problemas de conteo (y de optimización), incluidos el cálculo de combinaciones, permutaciones y particiones. `enum` fue exitosamente utilizado para contar acordes atonales en la música de Clemens Nachtmann (en "*Forbidding Harmonies*", [Nierhaus et al., 2015], pp. 33 - 57).

Ejemplos sencillos del uso de `enum`:

```
/* Enlista todos los conjuntos de 3 elementos que pueden formarse con los numeros 1 y 2, admitiendo repeticiones. */

3.enum([1,2])

-> [ [ 1, 1, 1 ], [ 1, 1, 2 ], [ 1, 2, 1 ], [ 1, 2, 2 ],
   [ 2, 1, 1 ], [ 2, 1, 2 ], [ 2, 2, 1 ], [ 2, 2, 2 ] ]
```

```
/* Enlista todos los posibles arreglos de longitud 5 que tengan 3 elementos. */

f = { |x, i, col| (i == 4).if { (col[0..3] ++ [x]).as(Set).size == 3 }{ true } };
c = 5.enum((0..2), f);
```

```
// Enlista arreglos de 4 elementos sin admitir repeticiones.

m = [60, 65, 62, 69];
d = m.differentiate.sign;
f = { |x,i,col| col[(0..i-1)].includes(x).not && ((x - col[i-1]).sign == d[i]); };
c = m.size.enum(m,f);
```

2.2.3. El Tonnetz

El *Tonnetz*, o red de tonos, es un diagrama que relaciona acordes mayores y menores por medio de adyacencias. Un acorde de tres notas se representa por un triángulo. Para llevar un acorde a otro hay tres movimientos posibles (P, R y L), que se obtienen de dejar fijo un lado del triángulo e intercambiar la nota restante por otra.

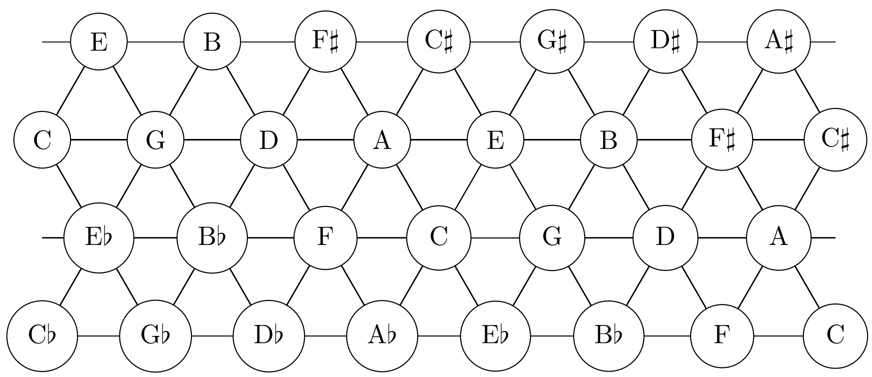


Figura 2.9: *Tonnetz.*

Las varias simetrías del *Tonnetz* garantizan que se pueden tomar distintos caminos cordales iniciando una secuencia en cierto acorde y terminándola en el mismo.

Tonalmente, los acordes más lejanos entre sí se distancian en cinco acordes (ver figura 2.10).

Figura 2.10: Distancia sonora máxima entre acordes.

Eligiendo recorrer el *Tonnetz* de izquierda a derecha (o de derecha a izquierda), éste regresará al mismo acorde en 23 pasos.

Otras posibilidades se esquematizan en la Figura 2.9.

La construcción del *Tonnetz* no está basada en la armonía funcional, sino en la cercanía que tienen dos acordes entre sí. Aunque es posible encontrar algunas transformaciones que permitan ubicar movimientos cadenciales sobre el diagrama. También resulta fácil ubicar qué notas se encuentran en cualquier tonalidad dada, así como se pueden encontrar los saltos por notas de séptimas y los acordes de séptimas (éstos son los que forman cuadrados).

Para programar un *Tonnetz* podemos representarlo por una gráfica:

```
/* C = 60 , C# = 61 , D = 62 ,  D # = 63 , E = 64 , F = 65 , F# = 66 , G =
   67 , G# = 68 , A = 69 ,
A# = 70 , B = 71 ,

*/

(
h = [  [65 , 69 , 66 , 71 , 67 , 70 ] , // D
     [65 , 60 , 64 , 61 , 66 , 62] , // A
     [60 , 67 , 71 , 63 , 61 , 69] , // E
     [67 , 62 , 66 , 63 , 68 , 64] , // B
     [62 , 69 , 61 , 70 , 63 , 71] , // F#
     [69 , 64 , 68 , 65 , 70 , 66] , // C#
     [64 , 71 , 63 , 60 , 65 , 61] , // G #
     [71 , 66 , 70 , 67 , 60 , 68] , // D#
     [66 , 61 , 65 , 62 , 67 , 63] , // A#
     [61 , 68 , 60 , 69 , 62 , 70] , // F
     [68 , 63 , 67 , 64 , 69 , 65] , // C
     [63 , 70 , 62 , 71 , 64 , 60] // G
];
)
```

Cada índice del arreglo representa un nodo, y los subarreglos indican las notas adyacentes a la nota x_i.

Sea $d(n)$ el número de pasos que hay entre dos acordes tocados en cierta secuencia. Un ejercicio consistiría en programar acordes para ser tocados por *SuperCollider*, moviéndose según la funcionalidad del *Tonnetz*. La idea básica para programar este algoritmo consiste en:

1. Tocar un primer acorde.

2. Para tocar un segundo acorde hay que elegir un nuevo acorde que comparta dos notas con el recién tocado y que difiera en una sola nota (sólo hay una posibilidad). No se puede elegir el acorde anterior.

3. Desde el paso $n = 2$, consideremos que para $n < d(5)$ no se admita tocar acordes anteriores.

4. Detener la secuencia cuando el primer acorde haya vuelto a sonar.

2.3. Síntesis de sonido

2.3.1. ¿Qué es la síntesis de sonido?

La síntesis de sonido se refiere a la generación artificial de varios sonidos y las maneras de combinarlos. Para realizar síntesis se emplea un dispositivo digital o electrónico llamado sintetizador.

2.3.2. Osciladores

El proceso de síntesis sonora comienza por crear cambios oscilantes en el voltaje eléctrico. Un dispositivo eléctrico llamado oscilador es el que crea estas variaciones del voltaje.

Un oscilador sólo puede generar un sonido a la vez. Por lo que, si se pretenden producir cuatro notas, se necesitan cuatro osciladores. Si se quieren producir tres tipos de sonidos, se necesitan tres osciladores. Si se quieren tocar acordes de cuatro notas usando tres tipos distintos de sonidos, se requerirán al menos doce osciladores.

En los primeros sintetizadores, los osciladores solían ubicarse en una sección separada del *hardware*. Con un *switch* se especificaba el tipo de oscilación y un *knob* controlaba la frecuencia de la oscilación. Estos tipos de osciladores solían ser demasiado grandes y caros, y eran conectados al resto del sintetizador vía cables. Debido a su precio y tamaño, la mayoría de los primeros sintetizadores contaba con muy pocos osciladores, que se combinaban para producir sonidos más complicados. Es decir, no era prioridad tocar múltiples notas simultáneas. Incluso en su evolución, en la que los osciladores pasaron a ser circuitos internos (y ya no externos), los manufacturadores continuaron enfocándose en usar los osciladores para potenciar el sonido, en vez de para incrementar la posibilidad de múltiples notas tocando al mismo tiempo. Esto no cambió sino hasta mediados de los años 80s, en que los circuitos de los osciladores eran ya suficientemente pequeños, potentes y baratos como para poder añadir polifonía.

Actualmente, los osciladores de los sintetizadores modernos son capaces de usar docenas — e incluso cientas — de voces simultáneamente. En la mayoría de los instrumentos, siguen existiendo un pequeño número de osciladores, a saber dos o tres. Sin embargo, es asumido que cada oscilador representa un gran número de ellos, todos configurados de manera idéntica, que posibilitan tocar varias notas con el mismo sonido.

Tipos de osciladores

Un oscilador es, en un sintetizador, la fuente generadora de sonido. Los osciladores más simples generan una onda repetitiva, que se desplaza de manera equidistante, es decir, positiva y negativamente, de un centro (donde el voltaje es igual a cero). La diferencia fundamental que hay entre los diferentes tipos de ondas fundamentales es: la proporción en que ocurren las transiciones en voltaje desde el polo positivo al negativo y la aceleración con que ocurren los cambios de dirección.

Los tipos de onda más elementales son:

- Onda sinusoidal

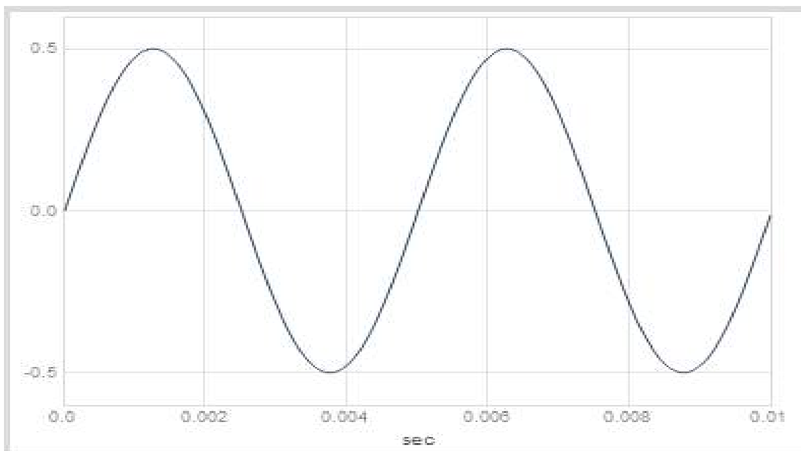


Figura 2.11: `{ SinOsc.ar(200, 0, 0.5) }.plot`

- Onda diente de sierra

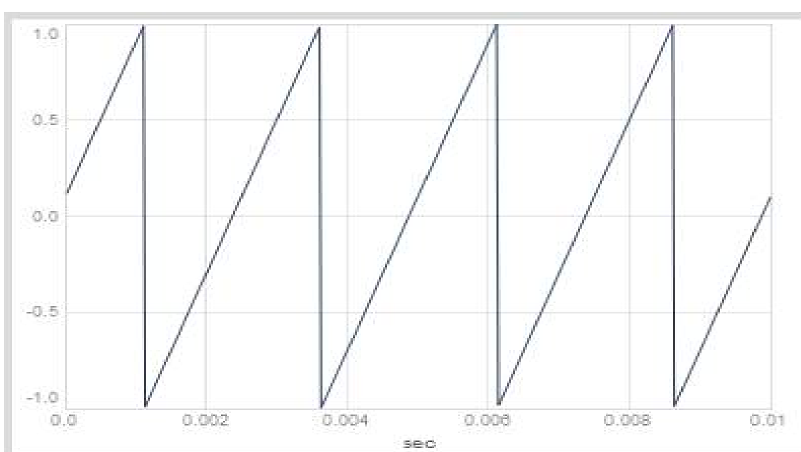


Figura 2.12: `{ LFSaw.ar(400, 0.1) }.plot`

- Onda cuadrada

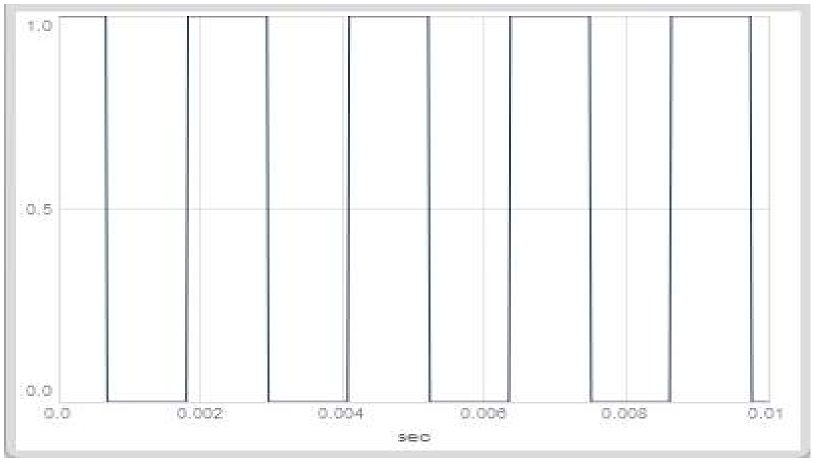


Figura 2.13: `{ LFPulse.ar(440, 0.2) }.plot`

- Onda triangular

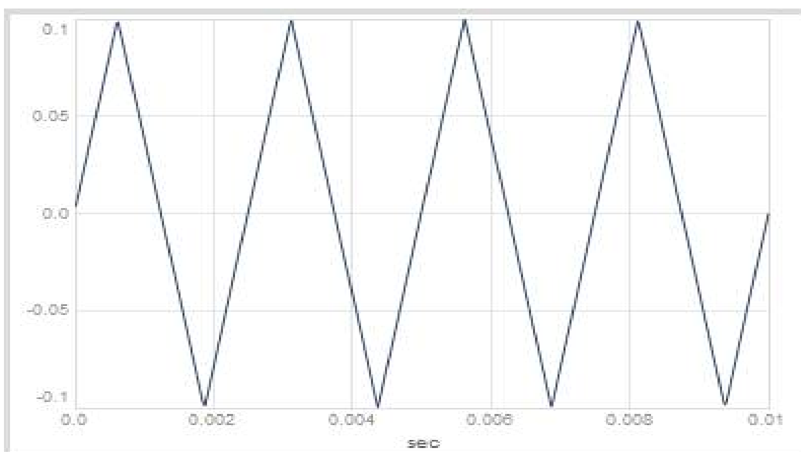


Figura 2.14: `{ LFTri.ar(400, 0, 0.1) }.plot`

Las ondas senoidales únicamente producen una frecuencia fundamental, pero ningún otro armónico. En contraste, los otros tipos de onda sí producen otros armónicos, lo que aporta una cualidad más brillante y filosa a su sonido.

Existen también osciladores para generar ruido. Se entiende por ruido a fluctuaciones aleatorias y rápidas de la frecuencia y amplitud de una onda oscilatoria. El resultado es un sonido que atraviesa todo el espectro de frecuencias, y lo hace tan rápidamente que, en el ruido se pierde toda noción de frecuencia definida y de afinación.

Tipos de ruido son: ruido blanco, ruido rojo, ruido rosa, ruido violeta, ruido azul o crackle.

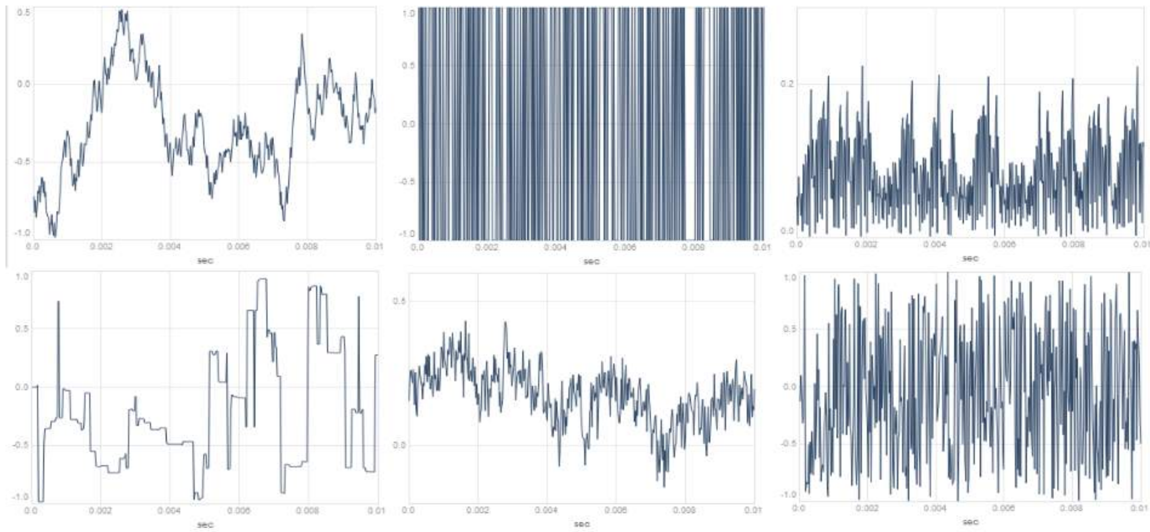


Figura 2.15: Distintos tipos de ruido en *SuperCollider*: `BrownNoise`, `ClipNoise`, `Crackle`, `GrayNoise`, `PinkNoise` y `WhiteNoise`.

2.3.3. Flujos de señales (combinaciones de osciladores y efectos)

En un proceso de síntesis, hay esencialmente dos maneras de combinar señales: la lineal y la no lineal (ver figura). La primera consiste en sumar (o apilar) señales y la segunda, en encadenarlas.

Sínteis lineal

El resultado de apilar señales conlleva que la onda de salida sea proporcional a la de entrada (de ahí el nombre de *lineal*).

La síntesis lineal es el proceso más simple para combinar osciladores, y consiste en hacer sonar osciladores simultánea o secuencialmente, creando un sonido compuesto.

En este método, sumando las funciones de las gráficas de las ondas asociadas a los osciladores, se puede determinar la forma de la onda compuesta. La palabra "sumar" puede significar restar, o lo que es mismo sumar números negativos, lo que depende de si las fases de dos ondas son positivas o negativas en un mismo punto (ver figura). Si dos ondas están en fase, éstas se suman, obteniéndose una onda de mayor amplitud. Si dos ondas están desfasadas, éstas producen una onda de menor amplitud, o silencio. Cuando dos ondas sonoras empiezan a una frecuencia parecida, puede que ambas comiencen en fase; pero en el transcurso del tiempo se desfasen (conforme a la onda de mayor frecuencia avanza). Llegará el momento en que ambas amplitudes comiencen a cancelarse, y eventualmente, las ondas vuelven a empatarse. Se repetirá el comportamiento permanentemente. Un efecto que tiene varias utilidades, según la magnitud de la diferencia de fase. Alrededor de los 200Hz, se genera una sensación rítmica, típica de la música electro-house. A más pequeñas distancias, un desfase permite afinar instrumentos.[5] Cuando la distancia $f_D = |f_1 - f_2|$ es aún más pequeña, el efecto audible percibido son clicks (alrededor del $f_D = 1Hz$), o aspereza (aproximadamente a los $f_D = 6Hz$).

Síntesis no lineal

En síntesis, encadenar señales significa "modificar una señal", o mejor dicho, "modularla". Se puede:

- Modular la frecuencia de un oscilador utilizando otro oscilador.
- Modular la amplitud de un oscilador con otro oscilador.
- Modular la fase de un oscilador empleando otro oscilador.

En cuanto a las variables que forman un sonido puro, éstas son las únicas tres posibilidades. Es posible que dos o más osciladores modulen al mismo sonido y que los osciladores moduladores estén siendo a su vez modulados por otros osciladores. Ya el proceso de síntesis en etapas no iniciales también podrá tener otras maneras de modulación: para los filtros, los envolventes, los efectos, y la secuenciación.

Al encadenar señales las modulaciones agregan frecuencias extra, por lo que la señal de salida puede llegar a ser muy diferente a la entrada y es no proporcional a ésta.

Cuando un oscilador modula a otro, el oscilador modulador no se escucha. Sólo se oye el que está siendo modulado.

La síntesis no lineal permite generar sonidos complicados usando pocos osciladores; mismos que, para conseguirlos emular linealmente, se requerirían demasiados.

2.3.4. Tipos de síntesis

Síntesis aditiva

La primera forma de síntesis que existió, históricamente, fue la implementada en el Telarmonio de Cahill; la síntesis aditiva.

[5] Al afinar dos cuerdas de una guitarra, si ambas frecuencias son muy parecidas, se escucha una vibración muy rápida, la cual desaparece cuando las dos cuerdas vibran a la misma frecuencia.

Es un tipo de síntesis que produce resultados artificiales, —no semejantes a los timbres de instrumentos musicales analógicos—, donde se suelen ocupar varios osciladores muy parecidos entre sí.

Un ejemplo de síntesis aditiva es: seleccionar "cinco" ondas senoidales para los osciladores, cuyas diferencias en frecuencias sean menores a 10 cents. En particular, se puede elegir un oscilador para la frecuencia central, dos osciladores a +3 cents y -3 cents de distancia de la frecuencia central, y otros dos osciladores a +7 cents y -7 cents de la frecuencia central.

En *SuperCollider*, ello se escribiría así:

```
{ [SinOsc.ar(300, 0, 0.1) + SinOsc.ar(303.0017334, 0, 0.1)
+ SinOsc.ar(296.9982666, 0, 0.1) + SinOsc.ar(307.0040446, 0, 0.1)
+ SinOsc.ar(292.9959554, 0, 0.1)]!2 }
```

La misma idea con 5 osciladores puede llevarse a 13 de ellos o a trescientos. Se produce entonces un efecto armónico más complejo al superponer muchos osciladores con mismas formas de onda pero en diferentes frecuecncias y fases.

El intervalo de cent está definido por $2^{\frac{1}{1200}}$. Es decir, al sumar este número a una frecuencia f_1, obtendremos una frecuencia f_2 que se encuentra a un cent de distancia de f_1.[6] El siguiente código, —que, obsérvese, es una generalización del anterior—, consta de 25 osciladores: el primer oscilador da la nota fundamental, los siguientes dos están situados a $|3|$ cents del primero y los otros veintidós distan en múltiplos de 4 cents del segundo y tercer oscilador:

```
    (
~synthCents = {
arg freq = 300, fAmp = 0.01;
var rand = exprand(0.01,1);
var rand1 = exprand(0.01,1);
var rand2 = exprand(0.01,1);
var numHarm = 12;
var sig = [SinOsc.ar(300, rand, fAmp), SinOsc.ar(300, rand, fAmp)];
for(0, numHarm, { |n|

    sig = sig + SinOsc.ar(( n * 4 * (2 **1200.reciprocal) - 1 ) +
   freq  ! 2, rand1, fAmp)
    + SinOsc.ar(((-1)*( n * 4 * (2 **1200.reciprocal) - 1 )) + freq
    ! 2, rand2, fAmp);

  });
sig;
}.play
```

[6] La ecuación es: $2^{\frac{1}{1200}} + f_1 = f_2$. Por lo tanto, al sumar $f_1 + \left(n(2^{\frac{1}{1200}})\right)$ encontramos la frecuencia f_2 que dista n cents de f_1.

```
17 )
```

La línea 8 (del código anterior) indica que el proceso comienza con un arreglo llamado `sig` que almacena una onda sinusoidal en el canal izquierdo y otra en el derecho (lo que simula una señal estéreo); ésta está afinada en una frecuencia fundamental de 300 Hz. La línea 9 indica que se realizarán doce iteraciones y en cada iteración se sumará una nueva onda al arreglo, afinada en una diferente frecuencia. Al finalizar las iteraciones, el arreglo tendrá una suma de 13 componentes sinusoidales por canal y el método `.play` permite escuchar el resultado.

El teorema de Fourier, demostrado por el matemático francés Joseph Fourier (1768-1830), afirma que: "siempre es posible descomponer una onda sonora periódica en una suma de componentes sinusoidales." Eso es el fundamento detrás de la síntesis aditiva.

Una consecuencia del teorema de Fourier es que todas las ondas básicas de la síntesis de sonido, —triangulares, sierras, cuadradas, etc— pueden construirse a partir de ondas senoidales.

Para obtener una onda diente de sierra, éste es el procedimiento:

Designar a un oscilador una frecuencia fundamental. Añadir más osciladores; cada uno debe afinarse en una frecuencia que sea múltiplo entero de la fundamental. Por ejemplo, si la frecuencia fundamental es 80 Hz, los armónicos sucesivos serán de 160 Hz (2×80 Hz), 240 Hz (3×80 Hz), 320 Hz (4×80 Hz), 400 Hz (5×80 Hz), etc.

Luego, se ajusta la amplitud correspondiente a cada oscilador tal que ésta sea el inverso del número de armónico: el segundo armónico tendrá amplitud $1/2$, el tercer armónico tendrá amplitud $1/3$, el cuarto armónico tendrá amplitud $1/4$, el quinto armónico tendrá amplitud $1/5$, etc. Después de realizar este cambio, se escuchará y verá graficado que la onda resultante es una sierra. En conclusión, la serie de Fourier que produce la onda diente de sierra es:

$$s(t) = \sum_{n=1}^{\infty} \frac{1}{n} sen(n\omega t + 0) = sen(\omega t) + \frac{1}{2} sen(2\omega t) + \frac{1}{3} sen(3\omega t) + \frac{1}{4} sen(4\omega t) + \dots,$$

donde ω es la frecuencia fundamental y t es el tiempo. Por simplicidad, la fase puede igualarse a cero y basta evaluar en $s(1)$.

En código:

```
    ~sierra = {
arg freq = 300, fAmp = 0.2;
```

```
var numHarm = 50;
var sig = [0, 0];
for(0, numHarm, { |n|

sig = sig + SinOsc.ar(freq * n ! 2, 0, fAmp/n)
});
sig
}
```

Si de la fórmula $\sum_{n=1}^{\infty} \frac{1}{n} sen(n\omega t)$ eliminamos todos los armónicos pares, construimos una onda cuadrada.

Es decir, la serie de Fourier que produce una onda cuadrada es:

$$s(t) = \sum_{n=1}^{\infty} \frac{1}{2n-1} \sin\Big((2n-1)\omega t\Big).$$

En código:

```
    ~cuadrada = {
arg freq = 300, fAmp = 0.3;
var numHarm = 30;
var sig = [0, 0];
for(1, numHarm, { |i|
var n = 2 * i - 1;
sig = sig + SinOsc.ar(freq * n, 0, fAmp/n);
});
sig
}
```

Para obtener una onda triangular, también hay que tomar exclusivamente los armónicos impares, invertir la fase de todas las componentes (de tal manera que las ondas se vean al revés), y establecer las amplitudes a $\frac{1}{n^2}$, siendo n el número de armónico.

Por lo tanto, la serie de Fourier asociada a una onda triangular es:

$$s(t) = \sum_{n=1}^{\infty} \frac{1}{(2n-1)^2} sen\Bigg(\Big((2n-1)\omega t\Big) + \pi\Bigg).$$

En código:

```
    ~triangular = { arg freq = 300, fAmp = 0.3;
var numHarm = 30;
var sig = [0, 0];
for(1, numHarm, { |i|
var n = 2 * i - 1;
var phase = if(i % 2 == 0, {0}, {pi});
```

```
sig = sig + SinOsc.ar(freq * n, phase, fundAmp * (1/n.squared));
});
sig
}
```

La cantidad de osciladores debe ser necesariamente finita, pero añadir cada vez más de ellos hace que éstos se vayan pareciendo cada vez más al sonido límite, que es el que se pretende obtener (respectivamente, una sierra, una onda cuadrada y una onda triangular).

Actualmente, la manera en que las computadoras generan ondas básicas es vía *wavetables* (o tablas de onda). Éstas son tablas que almacenan, en forma de datos, cada forma onda. Una tabla de onda es llamada cuando se requiere usar un oscilador y tiene la posibilidad de modificarse en su amplitud, frecuencia o fase, para producir síntesis. Es un método que consume muchos menos recursos a que si, cada vez que se requirieran crear nuevos osciladores, se tuviera que recurrir a la síntesis aditiva.

Conclusivamente, el método de síntesis aditiva consiste en combinar múltiples osciladores básicos de tal manera que al sumarse se obtengan nuevos colores y timbres.

La síntesis aditiva destaca resultados particularmente interesantes cuando los sinusoides son una mezcla de sonidos armónicos e inarmónicos (estos últimos no son múltiplos enteros de la frecuencia fundamental).

Síntesis AM

Como el nombre lo dice, la síntesis AM consiste en modular la amplitud de un oscilador usando otro oscilador. Al oscilador que está siendo modulado se le llama portador (del inglés, *carrier*), y al oscilador que le modifica se le llama modulador.

Modular es sinónimo de "cambiar". Al tocar guitarra eléctrica usando un pedal de efectos, suele haber un pedal que controla el volumen de la señal. Éste se usa para bajar o subir el volumen de la guitarra; es una manera de producir amplitud modulada.

Mover oscilatoriamente el volumen de una señal provoca un efecto llamado trémolo, que consiste en escuchar, intermitentemente, el subir y bajar del volumen de una señal de audio. Este es el efecto característico de la síntesis AM, y que se puede aplicar sobre el rango de todas las frecuencias audibles (más de 20Hz y menos de 20kHz).

Hay frecuencias añadidas al realizar la síntesis AM, a saber, las producidas por la suma $(f_c + f_m)$ y la resta $(f_c - f_m)$ de las frecuencias de ambos osciladores (el portador y el modulador). Se les llama a estas específicas frecuencias añadidas "bandas laterales" (en inglés, *sidebands*). En un contexto particular, si el sonido portador tiene una frecuencia de 1000 Hz, y la del modulador es 700 Hz, al final de la cadena de síntesis de escucharán sonidos a frecuencias de 1700 Hz (1000 Hz + 700 Hz) y 300 Hz (1000 Hz - 700 Hz).

En la práctica, la frecuencia moduladora cambia continuamente, por lo que las bandas laterales también lo hacen.

Síntesis FM

Como el nombre lo sugiere, síntesis FM consiste en modular la frecuencia de una onda sonora usando otra onda sonora. Al mover la amplitud del oscilador modulador, éste cambia la frecuencia del oscilador portador. Un tipo de modulación que, al igual que la síntesis AM, es aplicable a todo el espectro de frecuencias audibles.

En la síntesis FM, los cambios en frecuencia que va ejerciendo el modulador sobre el portador producen adicionales frecuencias en el sonido de salida. Éstas son considerablemente más complicadas que las producidas con síntesis AM. Adicional a la suma y diferencia de frecuencias de los osciladores, también se generan bandas en la suma y diferencia de las frecuencias que hay entre el portador y los múltiplos enteros de la frecuencia del modulador.[7]

Dependiendo de la amplitud del modulador, algunas de estas bandas de frecuencias llegan a ser tan fuertes que enmascaran a la frecuencia portadora.

Normalmente, las frecuencias de los osciladores modulador y portador son proporcionales entre sí; lo que se puede denotar por $\frac{f_c}{f_m}$ o $\frac{f_m}{f_c}$. Si ésta razón es armónica (es decir, produce un intervalo de octava, quinta justa, cuarta justa, tercera mayor, menor, etc), el portador producirá un sonido perfectamente entonado. Si en cambio, la proporción es inarmónica, el portador producirá un sonido metálico o ruidoso, apto para sintetizar campanas, metales o pianos electrónicos.

Los dos controles más importantes en la síntesis FM son su "proporción armónica" (la mencionada en el anterior párrafo) y su "índice de modulación". Ésta es la razón existente entre la amplitud del modulador y su frecuencia; lo que se denota por $\frac{a_m}{f_m}$.

Ejemplo. Casi todos los controladores MIDI incluyen una rueda de *pitch-bend*, la cual sirve para alterar la afinación de una nota al mover la rueda hacia arriba o hacia abajo; un efecto al que se le conoce como vibrato. La rapidez con que se va moviendo la rueda durante el vibrato es lo que identificaríamos como la razón armónica. La distancia que se desplaza la rueda con respecto al punto central es lo que, en este ejemplo, significaría el índice de modulación.

[7] Es decir, se agregan bandas de frecuencia en $|f_c+f_m|$, $|f_c-f_m|$, $|f_c+2f_m|$, $|f_c-2f_m|$, $|f_c+3f_m|$, $|f_c-3f_m|$. Se agregan valores absolutos a estas cantidades, pues las frecuencias negativas no existen. De modo que, valores negativos se vuelven valores positivos, pero invirtiéndose las fases.

La mayoría de sintetizadores FM permiten que el usuario, haciendo conexiones con cables, module a los osciladores moduladores. Estos sintetizadores suelen contar con un conjunto finito de bloques de construcción, llamados algoritmos, los cuáles permiten configurar múltiples osciladores para que se provoquen afectaciones al sonido. Por efecto, la de un modulador que afecta a otro modulador, el cual afecta a otro modulador, quien finalmente afecta a un portador. Algunos instrumentos incluso permiten loopear la señal de un modulador para que se afecte a sí mismo, creándose entonces una "modulación por *feedback*".

En términos del resultado sonoro, éste puede ser muy complicado e impredecible, dadas las tantas posibles variables a ser modificadas.

Síntesis sustractiva

La síntesis sustractiva parte de una onda sonora compleja, a la que se le aplican filtros para eliminar ciertas frecuencias y con ello modelar un nuevo sonido. Puede haber únicamente un filtro modificando a la señal o varios, y pueden algunos de ellos estar moviéndose dinámicamente en el tiempo.

Debido a los pocos recursos computacionales que este tipo de síntesis requiere, la síntesis sustractiva es muy popular. Se pudiera decir que todo sintetizador añade elementos sustractivos en su arquitectura, en forma de filtros de paso alto, de paso bajo, *knobs* de resonancias y de cortes de frecuencias (*cut-offs*).

Síntesis granular

La síntesis granular ya fue comentada en el Capítulo 1, Subsección 1.4.8.

Otros tipos de síntesis

Hay muchos tipos de síntesis y sus variantes. Como ejemplos están la síntesis por modelado físico, síntesis híbrida, síntesis Karplus-Strong, síntesis cruzada, síntesis vectorial, síntesis wavetable o síntesis RM. Para más información en el tema se puede consultar [Loy, 2007a], [Shepard, 2013], [Russ, 2004], o [Miranda, 1998].

2.4. Ritmos euclidianos y divisibilidad en la música

Godfried Tussaint (1944-2019), computólogo especializado en geometría computacional pero también percusionista, en el artículo "*The distance geometry of music*" demostró una interesante relación entre el algoritmo de Euclides y varios ritmos tradicionales del mundo (africanos, cubanos, hindúes, entre otros).[8] En resumen, este algoritmo es capaz de generar una colección de "ritmos equitativamente distribuidos" y que, según Tussaint, esta propiedad es característica de aquéllos ritmos muy agradables para el amplio público.[9] Un ritmo euclidiano puede ser generado para cualquier pareja de números enteros, pudiendo éstos ser o no primos relativos, aunque los números pequeños suelen ser musicalmente, de mayor interés.

Alex McLean realizó la programación exacta del algoritmo de Tussaint en *Tidalcycles* (lenguage de *live coding*), en [McLean, 2014], hecho que permite corroborar el poder musical que tiene este algoritmo en el *live coding*, primeramente porque pueden realizarse improvisaciones de música algorítmica que hagan uso de estos ritmos, encadenándolos, sobrelapándolos e intercambiándolos a grandes velocidades, lo que podría ser difícil y hasta imposible para algunos percusionistas.

A continuación explicaré qué es el algoritmo de Euclides y el algoritmo de Tussaint, mostraré ejemplos sonoros de ritmos euclidianos, y explicaré cómo emplearlos en composiciones musicales, usándolos como *timelines* y/o como método de orquestación.

2.4.1. El algoritmo de Euclides

El algoritmo de Euclides permite calcular el máximo común divisor de dos números, así como determinar si dos números son primos relativos. Para hablar del algoritmo de Euclides, primero hay que conocer el algoritmo de la división. El algoritmo de la división dice que, dados dos enteros a y b, siempre existen enteros q, r tales que $a = bq + r$, con $0 \leq r < b$. Por ejemplo, para $a = 14$ y $b = 8$,

$$14 = 8(1) + 6.$$

Geométricamente, el algoritmo de la división nos dice qué tan lejos está un número de dividir exactamente a otro número.

Ya con esto entendimos el algoritmo de la división. Recordemos ahora que el máximo común divisor de dos números a y b, es el mayor entero d que es divisor común de a y b.

Ejemplo. Ya que los divisores comunes de 14 y 8 son $\{1, 2\}$, el $MCD(14, 8) = 2$.

Cuando los números a y b son pequeños es posible enlistar todos los divisores y encontrar el divisor común. Alternativamente, se usa el algoritmo de Euclides para

[8] *The euclidean algorithm that generates traditional musical rhythms*, en [Toussaint, 2005].

[9] *The geometry of musical rhythm*, en [Toussaint, 2019], capítulo 1.

encontrar el MCD de 2 números. El algoritmo de Euclides consiste en calcular varios algoritmos de la división, (reduciéndose el par de números con cada iteración) y después de una cadena finita de operaciones se obtendrá un residuo igual a cero que termina el procedimiento. El último residuo no cero en la cadena de divisiones es el máximo común divisor.

Ejemplo. Usando el algoritmo de Euclides calcula el $MCD(14, 8)$,

$$14 = 8(1) + 6$$

$$8 = 6(1) + 2$$

$$6 = 6(1) + 0.$$

Como el último residuo distinto de cero es 2, $MCD(14, 8) = 2$.[10]

2.4.2. Versión musical de Tussaint del algoritmo de Euclides

Para generar ritmos, el algoritmo de Euclides implementa "paquetes" de cadenas binarias de números, y tras una serie de pasos, encuentra el ritmo euclidiano correspondiente. Veamos un ejemplo.

Dados dos números a y b, el número a representará "número de pulsos por unidad de tiempo", y el número b significará "número de golpes (u *onsets*) por unidad de tiempo."

Para $a = 14$ y $b = 8$, de todas las posibles maneras de distribuir 8 golpes en 14 pulsos, sólo hay una que resulta de emplear el algoritmo euclidiano.

Ejemplos de ritmos no euclidianos son:

00110110110011

10011001011101

10101010101101

Ejemplo. Encuentra el ritmo euclidiano $E(8, 14)$.

[10] En *Numbers and symmetry: an introduction to algebra*, [Johnston, 2020] se puede consultar una demostración de los algoritmos de la división y de Euclides.

Como $p = 14$, $o = 8$, habrán $14 \;-\; 8 = 6$ ceros y 8 unos. Al iniciar el algoritmo, comenzamos por colocar los unos y después los ceros:

$$[[1][1][1][1][1][1][1][1][0][0][0][0][0][0]]$$

Luego se distribuyen los ceros en los paquetes que ya tenían uno:

$$[[10][10][10][10][10][10][1][1]]$$

Esto se corresponde con que $8/6$ es $8 = 6(1) + 2$; se obtuvieron 6 paquetes de 2 bits, y sobraron 2 paquetes de 1 bit.

Esto termina el algoritmo de Euclides. Como en este caso los números no son primos relativos, continúa otro procedimiento para encontrar el ritmo correspondiente.

Ya que $8 = 2(4)$, se distribuirán los paquetes en cuatro cajas:

$$[[101][101][1010][1010]]$$

Luego se reordenarán para que queden alternados:

$$[[101][1010][101][1010]]$$

Por último se formará el patrón resultante al sacar los bits de sus cajas:

$$10110101011010$$

Escuchar el resultado en *TidalCycles*:

```
d1 $ s "bd:4(8,14)"
d2 $ s "hh*14"
d3 $ s "ht"
```

Ritmo euclidiano en el bombo electrónico.

Y con variación de tempo:

```
p "cpsfun" $ stack [ s "bd:4(8,14)" # gain "0.9",s "hh*14",
s "ht"] # cps (slow 8 $ 0.5 + saw)
```

Efecto de *TidalCycles* que automatiza la velocidad en que un ritmo es tocado.

Un ejemplo donde se puede ver qué ocurre cuando el par de números elegidos son primos relativos se puede consultar en "*Real World Algorithms. A begginer's guide*".[11]

Musicalmente, contar 14 tiempos por unidad ya es extraño, porque la cantidad de pulsos es grande. Es más común contar 7 tiempos y hacer subdivisiones. Lo que podría equivaler a un compás de 7/8; ya que en 7/8 se pueden agrupar hasta 14 dieciseisavos (o en 7/4 se pueden agrupar hasta 14 octavos por compás). La elección de compás realmente depende de cómo piensa la música un músico, sus habilidades y del contexto. 7/8 sigue siendo una métrica poco común para un compás, que se puede encontrar en piezas de rock progresivo ("*Frame by Frame*" – *King Crimson*), música celta ("*The fruit and the snoot*" – Liz Caroll), talas hindús, entre otros.

En el sentido de que es muy fácil automatizar ritmos euclidianos, esto presenta una ventaja para la música algorítmica, ya que los resultados producidos serán difícilmente emulables en las percusiones analógicas.

2.4.3. Timelines

El *timeline* es la pauta rítmica a la que se atiene una agrupación de músicos; suena durante toda o casi toda una pieza sin realmente desaparecer. Se diferencia del pulso en que acentúa únicamente ciertos golpes, pero la elección de éstos determina la base rítmica de una pieza o inclusive la de todo un género musical. En el clave-son, es el instrumento de la clave (dos percusiones de madera que se golpean entre sí para producir sonido) el que marca el *timeline* (ver [Toussaint, 2019], Capítulo 3 y Capítulo 6).

[11] En https://louridas.github.io/rwa/assignments/musical-rhythms/ .

3 Desarrollo de obras musicales

Cada vez que compongo música fijo mi atención en algo diferente. Si durante una sesión pienso estructurar una pieza por secciones, en otra ocasión me encontraré improvisando un arreglo de guitarra que querría se adecuara a esa estructura global. Y en otra sesión, trabajaría la síntesis de sonido, aún para mejorar la misma pieza. A veces estoy más en sintonía con los métodos tradicionales de composición, pero en otras ocasiones viro hacia la programación musical.

Las cuatro piezas que discuto en las siguientes páginas indagan cada una en un aspecto diferente del proceso de composición de piezas. Busco que, aislando las partes logre encontrar pequeños modelos matemáticos en cada una y así visibilizar estas líneas independientes del pensamiento que muchas veces, un compositor, más que buscar aislarlas, gusta trabajarlas conjuntamente como parte de ser original. Sin embargo, desglosar en elementos esenciales facilita la creación de una metodología que puede reutilizarse, y permite apreciar el alcance real de cada una de las partes. Así pues, depende de cómo se le vea, en la composición de una pieza hay información distinta y parte del reto consiste en identificar dónde va qué cosa.

La pieza 1 se enfoca en la mesoforma; la pieza 2 en la macroforma; la pieza 3 en la improvisación de guitarra con síntesis (donde interviene la microforma); la pieza 4, en una trama social que guía una composición.

3.1. Pieza 1: "*Ir*"

La pieza "*Ir*" comenzó con una melodía en guitarra, la cual puede apreciarse en la Figura 3.1.

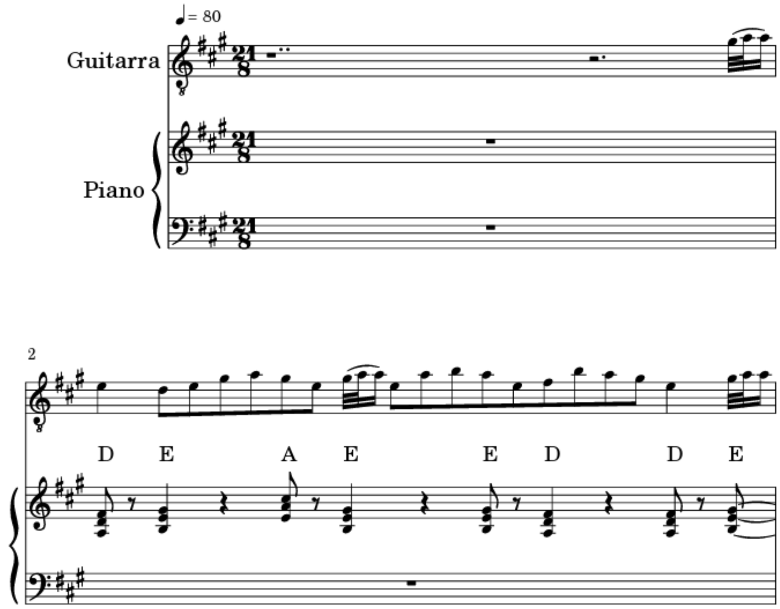


Figura 3.1: Primer Compás de “Ir”.

A pesar de que, para una más fácil lectura, la misma melodía podría escribirse en una métrica de $\frac{7}{8}$, el ciclo melódico realmente mide 3 compases de $\frac{7}{8}$; es decir, $3 * \frac{7}{8} = \frac{21}{8}$.

El segundo elemento que incorporé a la pieza fue la batería. Mi secuenciador *Ableton Live* me sugirió usar una métrica de $\frac{7}{8}$ para la parte percusiva; es decir, hay un contraste entre los ciclos largos que dirigen toda la pieza, contra los ciclos tres veces más rápidos que figuran para el ritmo, destacándose el beat principal (ver Figura 3.2).

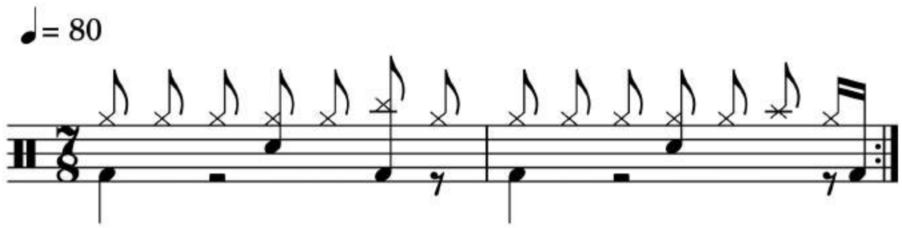


Figura 3.2: Rítmica en “Ir”.

Parte de la ingeniería de composición consiste en encontrar puntos de intersección entre los cíclos largos y los cortos, que están siendo tocados simultáneamente. Una manera de lograrlo es, tomar como únidad métrica de referencia a la subdivisión más pequeña de tiempo que sostenga la música. Ésta la da la figura de octavo en el sonido del *hi-hat*. Es cierto que en la melodía de guitarra hay subdivisiones de hasta treintaidosavos, pero sólo aparecen esporádicamente; no es necesario subdividir tanto para medir toda la pieza.

El tercer elemento agregado a la composición fue la armonía, que, como se aprecia en la Figura 3.3, en términos de rítmica, alterna 8 acordes siendo tocados dentro de un compás, seguidos de 6 acordes siendo tocados en el compás consecutivo. Sin embargo ésta elección de acordes no es fija. Es decir, la sucesión de acordes debe aprenderse de memoria o irse leyendo sin pretender predecir lo que viene a continuación. Me gustó que esto aporta la sensación de un espacio largo y sorpresivo, que juega con el hecho de que, una nota o sucesión de notas se puede armonizar de maneras diferentes. La situación de repetir la melodía guitarrística una y otra vez pide que se abra la música a partir de hacer más aleatoria la armonía.

Figura 3.3: Primer y segundo compases de "Ir".

Para toda la primera parte de la pieza sin contar el coro, los acordes son:

D E – A E – E D – D E

E A – D A – A E

E D – D E – E D – A E

E A – A B – A E

D A – E D – D A – A E

E D – D A – A A

3.2. Pieza 2: "*Tenemos que hablar de Sara*"

Normalmente las canciones se conforman por intro, estrofas, coros, puente y outro.

Inspirados en las matrices del juego de dados de Mozart y la pieza *Analogique B* de Xenakis (considerar la Figura 1.1 y la Figura 1.10, respectivamente), proponemos construir, para la composición de una pieza aleatoria, una matriz de 4 renglones y 16 columnas, que servirá para delimitar la macroforma de la misma, y donde cada una de sus entradas tiene duración de un compás. Todos los renglones de una misma columna representarán las posibles melodías que, para ese intervalo de tiempo podrían ocurrir, cada una con $\frac{1}{4}$ de probabilidad.

La pieza pretende ser, en su generalidad, una cadena de Markov, en tanto que cada compás actual siempre depende del que le precede, y a veces depende de los 2, 3 o cuatro compases anteriores (como se observa en la Figura 1.5). Para esto, convendremos en que los antecedentes y consecuentes musicales puedan ocurrir en tres dimensiones: melodía, armonía y ritmo.

Ejemplos:

Melodía

- Si una melodía es ascendente en un primer compás, una melodía descendente podría ser consecuente en el compás consecutivo.
- Si una melodía usa cuatro notas de una escala diatónica en un primer compás, otra melodía que usara el complemento de esas notas en el siguiente compás sería un buen consecuente.

Ritmo

- Si una melodía contiene muchas notas rápidas en un primer compás, una melodía con pocas notas y largas podría ser consecuente a la primera.
- El reflejado sobre el eje y, del ritmo establecido en un primer compás sería un buen consecuente a este.

Armonía

- A un primer grado armónico, le respondería bien un cuarto o quinto grado, mientras que a un quinto grado le respondería bien un primero.

- A un compás donde ocurre un acorde tocado al aire le puede responder una sucesión de notas sueltas o intervalos (y viceversa).

La combinación de las tres dimensiones (melodía, armonía y ritmo) genera fraseos, los cuáles pueden abarcar uno o varios compases.

Para mi pieza aleatoria, se considerará lo siguiente:

1. Primero, a cada columna se le asigna un grado armónico, de tal manera que los 16 compases sucesivos resultan en una unión de cadencias.
2. Si una melodía termina en cierta nota, las notas posibles con que iniciará el siguiente compás serán: la misma nota, la tercera mayor, la quinta justa, la séptima (mayor o menor, dependiendo del acorde) o la segunda (mayor o menor, dependiendo del acorde) ascendente o desendente.
3. Se enfatizará la misma nota objetivo en todos los renglones que pertenezcan a una misma columna.
4. Las frases de una misma columna serán variaciones entre ellas.
5. Las demás variables no mencionadas aquí (por ejemplo las dinámicas y articulaciones en el instrumento), serán libres y sujetas a la elección de la compositora.

Para la ejecución de la pieza, se lanzarán dos monedas al aire 16 veces, y cada combinación de las monedas indicará la melodía que deberá ser tocada:

- HH - Primer renglón de la columna i.
- HT - Segundo renglón de la columna i.
- TH - Tercer renglón de la columna i.
- TT - Cuarto renglón de la columna i.

La pieza será tocada sobre un colchón musical preestablecido.

Observación: Si se pretendiera alargar la pieza, bastaría duplicar algunas de las columnas de la matriz. Por ejemplo, las que correspondiesen a una estrofa. Análogamente, si se deseara incluir dos o tres coros, se alargaría la longitud de la matriz, de tal modo que ser respetaran las posibles realizaciones melódicas de esas columnas al igual que el orden de los grados armónicos.

Observación: Computacionalmente, para poder alinear alguna aplicación generadora de música electrónica con esta composición, tal aplicación requeriría poder organizarse por bloques, ya fuera visuales o temporales, para así poder definirse las secciones de la pieza que acompañarían a la guitarra. *TidalCycles*, *SuperCollider* y *Ableton Live* ciertamente cubren esta posibilidad.

3.3. Pieza 3: "*60s & 90s*"

La pieza "*60s & 90s*" fue presentada el jueves 22 de junio de 2023 durante el evento "*Flujos en emergencia*", como parte del concierto de clausura del XVIII Coloquio de la Comunidad Estudiantil del Posgrado en Música UNAM.

El objetivo de la pieza es experimentar la improvisación de guitarra paralelamente a la evaluación en tiempo real de código de *SuperCollider* de manera solista, procurando alternar estas tareas de un modo eficiente. Asímismo, se busca entender las limitaciones de esta configuración, ya que controlar dos instrumentos musicales quita cierta libertad de ejecución en ambos rubros, lo que no es malo necesariamente.

Esta pieza hace un énfasis en la microforma y se pregunta cuál es la influencia de ella en una improvisación guitarrística.

El uso de los *Patterns* conecta con la mesoforma, aunque no se tiene demasiado control de una macroforma.

Para la composición de la sección electrónica me centré exclusivamente en el problema de aplicar el método de combinaciones de la biblioteca `enum` (ver Capítulo 2, Subsección 2.3.2). El código de *SuperCollider*, en el contexto de mi pieza "*60s & 90s*" funciona como un acompañante de la guitarra, que ejecuta sintetizadores sobre la escala pentatónica mayor de Do.

El código se compone de tres partes:

1. La definición de los sintetizadores.
2. La definición de un método numérico encargado de calcular combinaciones sobre una escala musical y sobre sintetizadores.
3. La secuenciación de la música mediante *Patterns*.

Las combinaciones resultantes, que se muestran en un arreglo, pueden pasarse como parámetros del sonido secuenciado.

Ejemplo:

```
/* En cada arreglo hay un intervalo ascendente, seguido de un
   descendente, seguido de dos ascendentes. */

--> [ [ 60, 65, 62, 69, 71 ], [ 60, 69, 62, 65, 71 ], [ 60, 71, 62,
   65, 69 ],
     [ 65, 69, 60, 62, 71 ], [ 65, 71, 60, 62, 69 ], [ 62, 65, 60,
   69, 71 ],
     [ 62, 69, 60, 65, 71 ], [ 62, 71, 60, 65, 69 ], [ 69, 71, 60,
   62, 65 ] ]
```

```
/* Se pueden pasar las combinaciones obtenidas como parametros en
   los Patterns. */

\midinote, Pshuf(c, 1).flatten
        + Pstutter(c[0].size, Pseq([-12, 0, 12], inf))
        + Pseq(~phrases.scramble.flat, inf).collect { |x| ~octaves[x
   ] }
```

La interfaz analógica que media entre la guitarra y la música electrónica se conforma por dos computadoras. La primera genera sonido (en *SuperCollider*) y la segunda procesa esa señal entrante añadiendo efectos extra. La segunda computadora, a través de *Ableton Live*, también gestiona la señal de guitarra y la balancea con la de *SuperCollider*.

Los post-efectos consiguen una música más estructurada en el nivel de la mesoforma a que si sólo le relegásemos esta tarea a *SuperCollider*. También, distribuir visualmente los controles que se han de modificar durante una pieza vuelve más fácil la ejecución para la solista.

Músicos de actualidad como *FKJ*, en sus canales de *YouTube*, demuestran que un buen manejo de interfaces e instrumentos musicales, bien puede producir resultados increíbles, y donde la interacción entre ambas cosas debe ser algorítmica.

Un ejemplo de ello sería: primero tocar una secuencia pregrabada de batería; luego grabar un *loop* en los teclados, a continuación grabar otro *loop* de acordes en guitarra. Finalmente, improvisar figuras en guitarra. El reto consiste en mantener consciencia de este nivel de las cosas durante toda una pieza y no perderlo de vista a pesar de las varias líneas instrumentales.

Mi propuesta contempla la programación musical con código, a diferencia de lo que a la fecha es el *set-up* de *FKJ*, u otros artistas populares, que más se enfocan en la composición musical empleando *hardware* y *software* de marca. Es comprensible que ellos lo hagan así, pues éstos facilitan la automatización de procesos y están fabricados para que el músico se enfoque en tocar. Sin embargo, es mi opinión que se pueden encontrar equivalentes y extendidas herramientas en la programación, inclusive gratuitas, permitiéndose incorporar el diseño de interfaces y de sonidos a un proceso compositivo y performático. Aunque puede que la indagación en estos utensilios modifique el tipo de música generada, su sonido y estilo musical.

3.4. Pieza 4: "*La mujer que quiso actuar como hombre*"

3.4.1. Objetivo

Continuaré utilizando los métodos compositivos previamente descritos, explorando aquéllos rubros que antes dejé de lado. Esto es, creando una trama, que es la de una mujer que quiere actuar como hombre. Para ello empleamos un dron, describimos un escenario conceptual en donde a ciertos rasgos de lo que la mujer quiere proyectar se les asignan correspondencias con elementos, acciones o variables del espacio físico o sonoro y se hará una comparación entre la realidad visual con la identidad psicológica que la persona busca expresar. Este proyecto con perspectiva de género puede servir como una práctica y análisis sobre la autoimagen, también inclusiva con la diversidad sexual.

Hacer de esto una obra de arte, contrariamente a un estudio experimental o cognitivo puramente científico, es que el arte da la posibilidad de inventar una trama única, no necesariamente estadísticamente correcta. Se busca con ello expresar un punto de vista; una manera de hacer las cosas que venga de una sola persona, y quien ciertamente puede tener una visión sesgada o parcial con respecto a una realidad.

3.4.2. Marco teórico

Música algorítmica

Para la presente tesis, de entre los muchos modelos matemáticos musicales que existen seleccioné las cribas, ritmos euclidianos, las cadenas de Markov musicales, la síntesis *wavetable* o la granular. Una característica de un modelo matemático musical es que se le puede analizar independientemente del lenguaje de programación o instrumento con el que el sonido sea producido. También, un modelo matemático musical se puede mapear hacia las diversas escalas temporales (tales como el microsonido, mesosonido o macrosonido) obteniéndose resultados diferentes.[1]

El *live coding*, que es improvisación en tiempo real usando un ordenador, está relacionado directamente con la programación musical y nos proporciona maneras de conectar varias computadoras, lenguajes de programación, o de hacer interactuar visuales o *hardware* de manera reactiva.

[1] *Microsound*, Capítulo 1, en [Roads, 2001].

Algunas de las piezas que compuse también usaron guitarra, lo que ya implica la consideración de un cuerpo humano en la música. Así, pude reflexionar acerca de aquéllos mensajes que sí se conseguían proyectar desde ese instrumento hacia una audiencia y cuáles de mis quereres se quedaron más bien en la añoranza. También, la ejecución de un instrumento puede complicarse tanto como se quiera, pero el agregar más elementos a un set musical necesariamente restringe las posibilidades con el instrumento (ej. computadora más guitarra). Sin embargo, a pesar de que se añaden restricciones de movimiento, algunos mensajes quedan mejor determinados bajo un contexto multi-sonoro.

El sonido, aunque muchas veces significa la canalización de las emociones de la persona que le produce, no siempre induce una emoción en el escucha. A veces ni siquiera el sonido debe ser el encargado de transmitir tal emoción para una audiencia, sino que la vestimenta del artista o sus movimientos lo logran más fácilmente. Aquí corremos el riesgo de abandonar el mundo musical por un argumento de que "no sirve en este expresar ciertas cosas"; no es esa la intención. Más bien, queremos encontrar un lugar adecuado para la música dentro de una situación social en la que intervienen tanto la creación musical como la necesidad de actuar como hombre.

Clichés y sesgos cognitivos

Daniel Kahneman, en su libro "*Thinking Fast and Slow*" da una lista de fenómenos de percepción que todos los humanos cometemos; entre ellos, la disonancia cognitiva.[2]

Supongamos que un zorro hambriento está en un bosque y ve unas uvas que cuelgan de un alto árbol. El zorro salta para alcanzarlas pero están muy altas. El zorro sigue saltando en su empeño por llegar a las uvas a pesar de que su estrategia nunca le permitirá alcanzarlas. Cuando se empieza el zorro a cansar, en automático su cerebro comienza a interpretar la situación de manera retrospectiva, concluyendo:

"¡Bah! ¡En realidad nunca quise esas uvas!"

Así, el deseo del zorro por obtener las uvas empieza a luchar con el querer del zorro por abandonar el problema. Es decir, el zorro, inconscientemente ha generado una disonancia cognitiva entre ambas opciones, pasando entre una y otra sin objetivo fijo.

En realidad, si el zorro conscientiza que ha llegado a una disonancia cognitiva, puede elegir entre tres opciones: 1. Alcanzar las uvas usando un método diferente al de saltar. 2. Admitir que tiene una habilidad insuficiente para alcanzar las uvas. 3. Reinterpretar la situación retrospectivamente.

[2] *Thinking, fast and slow*, en [Kahneman, 2017].

Es fácil que cada quién adapte esta historia a su situación particular; a “aquella vez que ella soñó con ganar la Medalla Fields, pero cada día salía al cine con sus amigos.” “Aquélla vez que yo deseaba tener una novia pero me la pasaba estudiando en solitario desde la mañana hasta la noche.” Etcétera.

Visto así, con unos pocos principios ya justificados por Kahneman, es posible describir cierta situación, a continuación explorarla de manera práctica y finalmente filmarla. El enfoque propone una interacción entre la psicología de un personaje con su ambiente. Como los sesgos cognitivos aplican para cualquier persona, pueden adaptarse para una lesbiana, bisexual o transexual y considerar si las acciones de ellos exhaltan ciertos sesgos, y cómo eso se ve desde afuera. Yo simplemente tomaré el caso de una mujer que quiso actuar como hombre, buscando ser atractiva para otras mujeres.

A continuación se busca contrastar el plano de la psicología con el de la realidad. Se parte del supuesto de que a una mujer le gustan otras mujeres, por lo que quiere adaptar sus comportamientos a algo que favorezca este objetivo. Cuenta ella con un robot y asistente psicológico que le proporciona maneras de mejorar y justificar ese querer. Su condición de mujer por otra parte, parece entorpecer la realización de su tarea; primeramente están las variables genéticas y luego las sociales, en las que está compitiendo contra otros hombres (u otras mujeres) para seducir a cierta mujer, y asímismo compite contra la mujer que le interesa.

La realidad en que vivimos es heteronormada; favorece pues, a las parejas heterosexuales y las pinta de bellas. Si al azar elegimos a una mujer joven de la Ciudad de México y le preguntamos qué espera del amor, ella seguramente responderá que quiere encontrar a un hombre que la quiera, tal vez tener hijos con él, o tal vez sólo compartir con él su pasión por el deporte.

La respuesta anterior no nos habla de la posibilidad de que a esa mujer le interesen otras mujeres. La mujer sólo se pronunció en positivo para los hombres pero no dijo ser heterosexual. Así que pudiera ser heterosexual o bisexual (o en el peor de los casos, lesbiana de clóset). Y una mujer lesbiana podría interesarse por la mujer entrevistada a pesar de que ella se hubiera pronunciado por los hombres en un inicio.

Algo que la respuesta sí sugiere, es que es nueva en el tema LGBT, por lo que en su respuesta, ese factor no se le ocurrió.

Se le pudiera a ella preguntar a continuación: “¿Qué opina usted de las relaciones homosexuales? ¿Usted tendría una?”

Algunas de las respuestas posibles serían:

- Son un pecado y no tendría una.
- Respeto a los homosexuales, pero a mí no me interesan las mujeres.

- Nunca había pensado en eso.
- Respeto a los homosexuales y tal vez tendría una pareja de mi mismo sexo.
- Ya he tenido una historia homosexual pero no me gustó.
- Ya he tenido una historia homosexual y tendría otra.
- Ya he tenido otras historias homosexuales y tendría otra.
- Respeto a los homosexuales y tal vez tendría una pareja de mi mismo sexo, si a ella también le gustara el deporte.
- Respeto a los homosexuales pero es mucha la presión social para ellos y qué flojera me daría ser una.

Ante todas estas respuestas, alguna mujer homosexual continuaría interesada por la entrevistada, ya que incluso la primera de ellas podría tener algo de falso. De aquí, una conclusión es: la diversidad va más allá de cualquier respuesta general.

Una rápida inspección a la respuesta "Ya he tenido una historia homosexual pero no me gustó" puede ser resultado de una decisión inconsciente de la mujer, una interpretación retrospectiva consecuencia de una no procesada disonancia cognitiva. A un hombre interesado esa respuesta le alegraría, mientras que una mujer homosexual pensaría: "tal vez era una pareja poco adecuada, pero hay otras," o "tal vez esta mujer desea que un caballero la transporte de un lado para otro porque ella no tiene coche. Si de esto se trata, también yo tengo coche y pudiera recogerla a ella cada vez que quiera." Una mujer homosexual aún más experimentada pensaría: "Pero si además de querer ser transportada en coche quiere que le ayude a realizar sus rutas de escalada, también me tendría que comprometer a ir al gimnasio y ahí hay hombres que son muy buenos escalando, y ¿qué voy a hacer para seguir siendo atractiva para ella?"

Bien un sesgo de la mujer homosexual es inventarse un mundo gay tipo serie de televisión, teniendo amigas lesbianas, pero gustarle alguien fuera de ese círculo quien lleva una vida ocupada. Bien podría plantearse convertirla al lesbianismo (que en mi opinión es mala estrategia, pues si acaso ello llegara a depender de dos personas, primordialmente la decisión sería de quien portara determinada orientación sexual), aunque podría también reducirse el problema a una noche de sexo casual. Esta situación ya ha sido previamente retratada en la película "*Habitación en Roma*". Un *cliché* que se convirtió en película.

Yo con mi pieza audiovisual pretendo ser más breve. Bien usando clichés o asociaciones inventadas entre los elementos involucrados.

El planteamiento de "*Habitación en Roma*" es realista. Es más plausible de suceder que una mujer felizmente casada y una lesbiana que conoce en Roma tengan un *affaire* de una noche pero se despidan al día siguiente, a que terminen viviendo juntas y felices para siempre después de esa noche. Sigue siendo criticable que esto suceda una y otra vez, lo que nos lleva a preguntarnos por qué sucede, y la explicación detrás muchas veces es la dependencia que las mujeres presentan hacia los hombres: "Él era muy rico y la consentía, por lo que ella vivía muy bien (y también lo amaba)"; o la inseguridad de las mujeres para elegir irse con la recién conocida. En la misma película hay varios momentos en que las chicas se plantean la posibilidad de escapar juntas, pero se decepcionan al imaginar las consecuencias.

Un planteamiento más a favor de las relaciones lésbicas lo da la película "*Kyss Mig*" en donde, a pesar de que una de las mujeres recién se comprometió, se enamora de Frida y al finalizar la historia, son ambas quienes se quedan juntas. Es una película que muestra lo que una romántica lesbiana quisiera, pero esa situación, dadas esas circunstancias, es prácticamente imposible de suceder, ya que los intereses suelen ir más allá del amor. Si a él le va bien, si el hombre es útil, o si es mucho más controlador, suele él lograr boicotear el amor entre mujeres, y una vía para ello es haciéndola cambiar de opinión. Suele ella tener unos sentimientos más encontrados que lo que la anterior película ilustra, por lo que nunca el amor femenino por el masculino queda totalmente anulado. Por lo general, es la mujer la que se cansa de batallar tanto con el hombre, y por lo tanto él gana.

Paralelamente a esta lucha homosexual, hay mujeres que también luchan por lograr mejores relaciones heterosexuales.[3] Citando a Mona Chollet,

> "*Codificada en nuestras comedias románticas y modelos de pareja ideal, existe una forma de inferioridad femenina que sugiere que las mujeres deben elegir entre la realización personal y la romántica.*"

Cada quien tiene un ideal romántico que a veces no se corresponde con la realidad, o que lleva a la realidad del patriarcado. Pero hay otras posibilidades. Ejemplos podemos encontrarlos en la realidad LGBT.

Si el individuo trae una lucha interna entre lo que quiere y lo que puede, y una línea de aprendizaje LGBT que lleva tiempo para ser asimilada, tendrá un conflicto con otras realidades que se le opongan. Por ejemplo, cuando un hombre sólo ve lo que quiere de una mujer (algo que sucede frecuentemente) pero esta mujer es homosexual, se presenta también la inferioridad femenina, pudiendo sabotear él lo que ella realmente desea, y donde, por ejemplo, a esa mujer no se le permitirá figurar en el camaradismo entre hombres (los amigos de este hombre), ya que está siendo vista como "una mujer" donde no se toman en cuenta sus intereses y de quien no se cree que esté a la altura del pensamiento masculino. Esto nos lleva a tener que diferenciar al género masculino del femenino, que al menos desde la perspectiva de la genética, esta clasificación es

[3] *Reinventar el amor: cómo el patriarcado sabotea las relaciones heterosexuales*, en [Chollet, 2023].

necesaria, al igual que desde la de la ley. Pero la consciencia humana sobre lo que se entiende por masculino y femenino permite reinventar el género y expresar un querer.

Una mujer homosexual podrá manifestarse en clichés masculinos y conservar algunas cualidades femeninas. Una mujer siempre es juzgada por ser mujer, y esto es cierto incluso cuando interactúa con otras mujeres. Si a nosotras se nos tacha de inseguras, quizá ello explique que si dos mujeres se gustan, aún así ninguna dé el primer paso. O si se nos cataloga de vanidosas, el amor femenino siempre se esfumará cuando prefiramos competir por quién está mejor vestida y los hombres serán los mejores colegas para defendernos de la otra mujer.

En resumen, y retomando el hilo de los sesgos cognitivos, estoy segura que si el lector lo desea, puede encontrar algunos de ellos en las situaciones mencionadas y analizarlos a profundidad.

La pieza y su relación con la tecnología

Uso la programación musical, la guitarra, un dron y una red inalámbrica (que también incluye conexiones inalámbricas para guitarra y *EcoFlow*) para producir la música y los visuales. Continuaré enfatizando algunos modelos matemáticos musicales en específico.

La presente pieza está más abierta a la colaboración que la creación de las obras individuales, proponiendo que otros actores, músicos o programadores contribuyan a la visibilización de esta situación social específica, la cual, de no ser el eje temático, podría quedar completamente anulada.

3.4.3. Planteamiento del problema

> Érase una vez una mujer que intentaba actuar como hombre para ser más atractiva hacia otras mujeres. En el intento de actuar como hombre se cohibía ella misma. Otras mujeres sólo serían sus amigas si replicaba sus comportamientos; el de tener un novio para ponerse celosas entre ellas, por ejemplo; pero ella no quería uno. Los hombres cuando la veían, la percibían como una simple mujer. Cuando la mujer intentaba copiarles a los hombres, ella sentía que se quedaba corta. Los hombres duplicaban el sentimiento de inferioridad que ella misma tenía. Una mujer heterosexual que le gustaba esperaba que ella se comportara como caballero. Su amiga lesbiana esperaba que le acompañase a todos lados. Otra mujer heterosexual la aceptaría siempre y cuando se pusiera más

```
corpulenta. La mujer lo intentaba, pero a otros, ni a ella misma,
les parecía que eso era suficiente. La mujer terminó por odiar
a todo mundo. Se sentía tan masculina que no concebía reconvertirse
a heterosexual. Si lo hacía, su amiga lesbiana le dejaría de hablar.
Las mujeres que le gustaban la tacharían de insegura. Los hombres
le quitarían el espacio en aquéllo que ella misma había aprendido.
Toda su experiencia se iría a la basura. Y finalmente, a ella
le seguían gustando las mujeres e imaginaba que algo debía hacer.
   Sólo tenía 4 pistas sobre lo que quería:
   1. Ser rápida.
   2. Ser hábil.
   3. Ver las cosas como son sin necesidad de su intervención.
   4. Saber tomar buenas decisiones.
```

La anterior historia, primeramente nos permite observar que es una mujer a la que le gustan las mujeres, y aunque no se nos dice algo sobre que le gusten o no los hombres, su objetivo en mente es el de ser atractiva hacia otras mujeres.

En segundo lugar, es una persona pudorosa. Pero esta es una característica de personalidad que específicamente ella presenta, ya que no todas las mujeres u hombres lo son. Existirá en este mundo una actriz que haya interpretado a un hombre y para tener éxito en ello, debió omitir la vergüenza de hacerse pasar por un miembro del sexo opuesto. Un bailarín puede moverse libremente por un escenario. Si acaso el pudor del que habláramos fuera la pena de moverse mucho, el bailarín sería buen complemento. O tal vez el pudor del que se tratara fuera el miedo a ser juzgada negativamente. Se nos pueden ocurrir situaciones en donde el pudor es minimizado o incluso ignorado y que contrarresten, por lo tanto un pudor exhacervado.

Tan sólo esos dos párrafos proponen analizar el asunto para pensar cómo retratarlo. Algunas frases, a primera leída son ambiguas. Por ejemplo: “Los hombres cuando la veían, la percibían como una simple mujer.” Pero para hacer eso visible, deberemos pensar, primero que nada, qué significa ser percibida como una simple mujer. Lo que nos llevará a indagar en las situaciones específicas que detonan que esa mujer crea que es percibida como una simple mujer. O incluso, cuáles de esas situaciones demuestran que no es sólo ella la que cree que los hombres la ven así, sino que otras personas también son capaces de ver el mismo fenómeno, quizá en actitudes o comportamientos frecuentes.

Para ciertas mujeres, ella les sería atractiva si la vieran con un novio, pero ella no quería uno. Así que nuestra protagonista no haría cualquier cosa para lograr ser atractiva hacia otras mujeres. Si una mujer homosexual tuviera que andar con hombres para seducir a una mujer, ya no sería por definición, homosexual. Si andar con un hombre fuera obligado, se estaría violando el derecho a la diversidad sexual. Estaríamos también diciendo, mediante acciones, que las relaciones de pareja lésbicas necesitan un agente masculino y que por lo tanto, son tríos, y no relaciones de pareja. Hay quienes ven este cliché del trío, pero quienes no, fácilmente pueden caer en ello, por lo que mi interés es visibilizarlo. Visibilizar esta y otras situaciones pertinentes a la historia de una mujer que quiso actuar como hombre. Utilizaré para ello, el medio audiovisual.

3.4.4. Hipótesis

Es posible construir un escenario audiovisual en donde a ciertos rasgos de lo que una mujer quiere proyectar se les asignen correspondencias con elementos, acciones o variables del espacio físico o sonoro, y de tal modo que se contrasten la realidad visual con la identidad psicológica que la persona busca expresar. Este proyecto busca incidir en la práctica y análisis de la autoimagen, proporcionando el fortalecimiento de una identidad.

3.4.5. Justificación

En primer lugar, la equidad de género se encuentra enlistada entre las 17 metas de la ONU para el desarrollo sustentable. Algunas de las subproblemáticas que ya planteé tienen que ver con una desigualdad asumida en las relaciones de pareja o en lo que se entiende socialmente que debe ser una mujer.

La comunidad LGBT es una minoría dentro de nuestra sociedad, que muchas veces es odiada, ignorada, burlada, otras veces es tomada como una moda o etiqueta. Estas visiones generales y a veces egoístas de lo que debe ser o no debe ser una persona LGBT o su comunidad, impiden profundizar en el significado e implicaciones de la diversidad. Planteo entonces una trama que parte desde el interior (desde una persona) de la comunidad LGBT. Así visibilizaremos su complejidad, al igual que haremos partícipe a esta comunidad.

Por otro lado, los drones comerciales actualmente son muy populares, proporcionando imágenes panorámicas de alta calidad, movimientos de cámara únicos para la fotografía aérea, y se operan mediante control remoto. En esto se ve involucrada información satelital, conexiones WI-FI, reconocimiento de objetos, planeación de rutas, o la destreza en el joystick, entre otros. Pueden servir como un asistente autónomo para retratar al músico y su escenario.

La manera de producir la música ya se trabajó anteriormente y se seguirá perfeccionando sobre esa línea.

En resumen, este proyecto busca:

- Incidir en la práctica y análisis de la autoimagen, proporcionando el fortalecimiento de una identidad.
- Visibilizar la problemática de una mujer que quiere actuar como hombre para ser atractiva hacia otras mujeres.
- Fomentar la creación musical con perspectiva de género.
- Fomentar la programación musical y los modelos matemáticos musicales.
- Fomentar el gusto por la creación artística.
- Explorar la tecnología con drones.
- Generar una escenografía audiovisual para una trama.

3.4.6. Metodología

Separadamente se considerarán:

1. La creación musical.
2. La exploración de imágenes a partir del dron.
3. La selección de aquéllas situaciones que tienen que ver con la trama.

Ello permitirá desglosar cada línea de investigación en elementos básicos, hasta llegar a aquéllos simbolismos que independientemente de las otras líneas de investigación son importantes de mencionar. Por ejemplo, dentro de la creación musical podemos dar forma al total de una pieza tomando como punto de referencia a los contrastes entre las partes. Éstos pudieran ser: consonancia contra disonancia, tonalidad contra atonalidad, melodía contra armonía, síntesis sonora contra instrumentación, arreglos preescritos contra improvisación, formalismo contra intuicionismo, coherencia contra inventiva, espontaneidad contra reflexión, intervalos contra morfologías, suavidad contra aspereza, atracción contra repulsión, variación de parámetros contra variación de estrategias, etcétera. El contraste, pues, es una manera de pensar en una lucha interna, o una lucha entre dos personas, de modo que la música misma puede contarnos, si así lo preferimos, todo. La música a veces puede apreciarse positiva o negativa, oscura o brillante. Y por ejemplo, ser muy negativo repele a la otra gente. Podemos asociar la actitud de una persona con un sintetizador, pero a otra gente con algo fuera de él. Y tal sintetizador puede verse modulado en el tiempo por otras actitudes; es decir, por otros sonidos. Todo esto en el nivel de la pura música. Más aún, este escenario descrito puede interpretarse como un juego de estrategias mixtas entre n personajes.

Actuar como hombre puede manifestarse en la velocidad al caminar, al realizar varias acciones concatenadamente, o al tocar un solo de guitarra. Y eso tiene una manifestación también visual. Una imagen es capaz de resumir muchas cosas. Por ejemplo, se reduce a un minuto lo que a una persona le tomó un año (o más) aprender. O cuando se rueda la escena de una película, de todo ese tiempo filmado se elige una mínima parte que forme parte de la película; dos minutos por hora, en promedio, según Daniel Tubau.[4]

Lo visual nos ayudará a sintetizar mucha información, pero con el análisis de las situaciones específicas (inciso 3) buscaremos ser aún más precisos acerca de lo que se quiere decir y cómo es mejor hacerlo.

3.4.7. Fuentes de investigación

Música algorítmica por un lado, drones por otro, sesgos congitivos por otro, algunas referencias sobre la comunidad LGBT y proyectos de actualidad relacionados al protocolo. El guión cinematográfico sirve como una inspiración para desarrollar escenas preestablecidas que consigan más impacto y coherencia.

3.4.8. Realización

Se parte de una improvisación base de *live coding* en *TidalCycles*. Sobre esto, que son beats percusivos con algunos efectos FX, se realizó una improvisación en guitarra. Las sesiones de grabación de la música percusiva y la guitarrística fueron totalmente independientes entre sí.

La improvisación de guitarra se practicó previo a realizar el video musical, durante una misma sesión. Estuve yo sentada en un sillón en la planta baja de la casa, mientras que la rola sonaba sin interrupción a manera de *loop*. La improvisación en guitarra se grabó al mismo tiempo que se reproducía la música (el beat de *TidalCycles*) y yo era capaz de escuchar la guitarra con claridad desde la planta baja, a pesar de que su amplificación sucedía en un primer piso.

[4]Ver capítulo *Entender el medio audiovisual* en *Las paradojas del guionista* de Daniel Tubau, en [Tubau, 2013].

3.4.9. Resultados experimentales

Video

La más grande dificultad del videoarte es tomar buenos cuadros de video al momento de una grabación. En el caso de videoarte solista con drones hay que considerar posibles desenfoques (tanto en objetos cercanos como en objetos a distancia), grabar secuencias de videos lo suficientemente largas para ser útiles, grabar atractivos cuadros cinematográficos y tener buena iluminación. Realizar todas estas cosas en solitario conlleva errores, ya que al momento de la grabación algunas imperfecciones son indistinguibles.

La ventaja de los videos con dron son las vistas panorámicas, las secuencias en movimiento y el posicionamiento de la cámara en sitios poco accesibles para otros tipos de cámaras.

En este caso "*La mujer que quiso actuar como hombre*" aprovecha la posibilidad de una escena vista desde arriba donde Ofelia se recuesta en un suelo cubierto por una colcha afelpada y comienza a realizar movimientos, alternando el tiempo de interpretación para la cámara con el manejo del dron. Fue una grabación que consistió de dos sesiones: una con duración de cinco minutos que fue tomada a las 3 de la tarde, y otra con duración de veinte minutos que fue tomada a las siete de la mañana.

La primera sesión reluce por tener una sutil mejor iluminación y un mejor enfoque en las secuencias tomadas de cerca, donde Ofelia aparece de medio cuerpo. Ofelia está vestida con una playera negra y shorts.

Como era la primera grabación, Ofelia quedó medio desprevenida en cuanto a los gestos faciales que debía realizar, pero se contrasta con la elección de las posiciones corporales, las cuáles fueron únicas con respecto a todo el material videográfico.

La segunda sesión luce por varias cosas: la incorporación de una pantalla verde en los bordes de la colcha, la sudadera 3D que protagoniza la primera parte de la grabación, el uso de la guitarra en una sección consecutiva de la grabación, mayor tiempo de grabación, gestos y movimientos algo más relajados. Sin embargo, la segunda sesión muestra un leve desenfoque en las escenas tomadas desde cerca, lo que arruina bastante del material disponible.

En ambas sesiones se experimentó con dos alturas de cámara diferentes, de tal forma que en total, obtuve cuatro puntos de referencia en cuanto a posibilidades para tomar cuadros cinematográficos. Los que más me gustaron fueron el close-up del primer video (que con respecto al close-up del segundo video está más alejado del sujeto a ser fotografiado) y la vista panorámica del segundo video (que con respecto a la vista

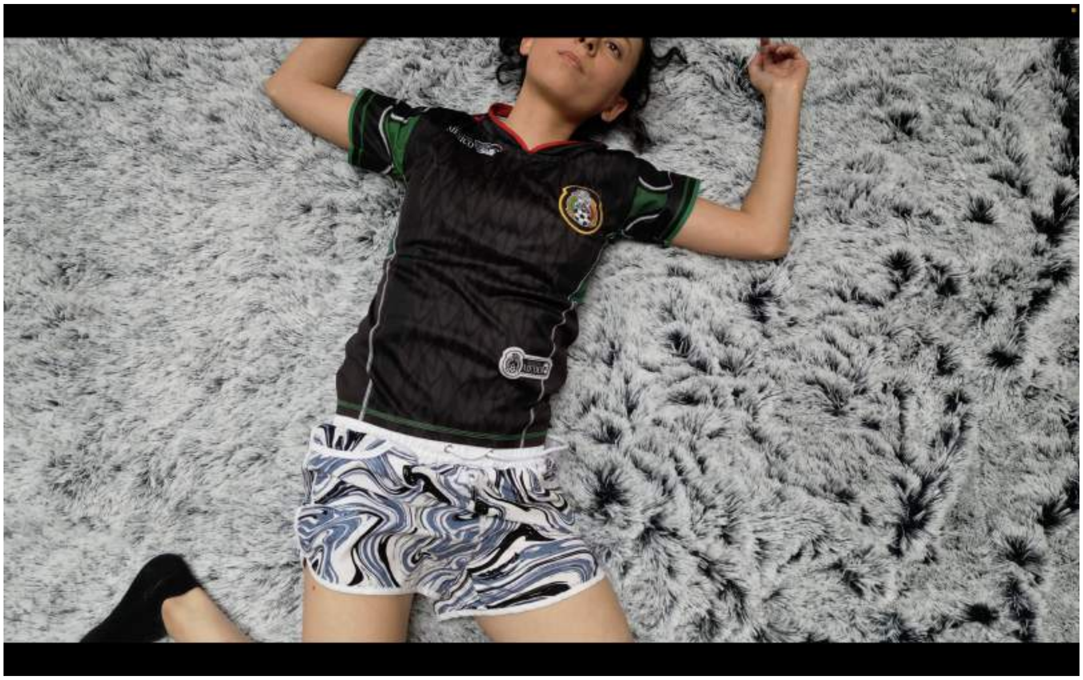


Figura 3.4: Encuadre desde el dron en plano cenital.

Figura 3.5: Encuadre medianamente alejado del sujeto.

panorámica del primer video, es más cercana al sujeto a ser fotografiado). Habría resultado útil anotar la distancia exacta a la que voló el dron cada vez, sin embargo fue un dato que al momento de la grabación no tomé en cuenta pues estuve más preocupada por "improvisar movimientos con respecto al tiempo de batería."

En resumen, hay tecnicismos que considerar respecto a la interpretación para el video y a la manipulación del dron.

Interpretación

Aquí distinguimos: gestos faciales, posiciones corporales, tiempo, consciencia de cuadro fotográfico y optativamente, ejecución en guitarra.

GESTOS FACIALES: A la cámara le gusta que los gestos sean definidos y queden fijos durante cierto tiempo sin variarlos. Si se elige cerrar los ojos, la intérprete debe contar al menos siete segundos para re-abrirlos. Lo mismo si decide reír, mantener la boca abierta, o poner un gesto triste. Hay pues, gestos faciales más fáciles que otros, pues unos son relajados y otros tensos. La transición entre gestos debe ser lo menos forzada posible. Específicamente, el nerviosismo provoca el arruinar de varias tomas.

POSICIONES CORPORALES: Aquí se distinguen las posiciones estáticas y las en movimiento. Las posiciones estáticas consideran: sentarse, pararse, recostarse, realizar alguna posición de yoga, o tomar una guitarra y fijar esa posición por al menos siete segundos (ya que ese lapso de tiempo permitirá seleccionar intervalos de esa secuencia en la edición). Las posiciones en movimiento consideran: rodar de una esquina a otra, de un lado a otro, cambiar rápida o lentamente de posición, bailar, o realizar espirales.

CONSCIENCIA DE CUADRO CINEMATOGRÁFICO: Es lo que le pide a la intérprete considerar que está grabando para una cámara, por lo que, antes de interpretar, debe colocarse en un cuadro cinematográfico y evaluar que éste sea suficientemente atractivo. Tomar en cuenta el espacio que tiene para realizar movimientos y la tendencia que muestra al situarse en cierto lado del cuadro (por ejemplo, si ha dejado el controlador del dron en su lado izquierdo, eso puede manifestarle la fijación de quedarse cerca de éste). También, con respecto al tiempo, debe la intérprete considerar qué tipos de movimientos ya ha realizado, e intentar nuevas cosas con respecto al tiempo que le queda.

TIEMPO: Con ello nos referimos a que la intérprete lleve una medida (precisa o aproximada) de sus tiempos de realización frente a la cámara. Es decir: considerar que tiene un total de 20 minutos para efectuar todo el ejercicio, y que alternará el manejo del dron con la interpretación. Debe posicionar el dron de tal modo que la imagen quede enfocada; elegir buenos cuadros fotográficos; situarlo de tal forma que se eviten posibles colisiones con el objeto volador, y que ella evalúe lo que puede

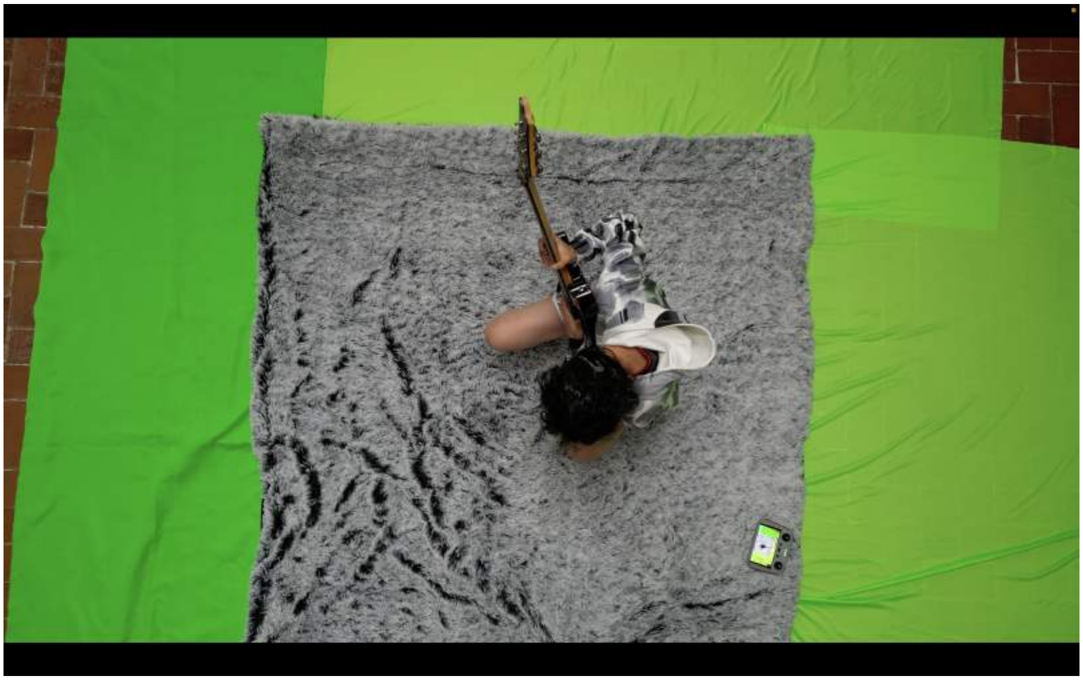


Figura 3.6: Encuadre desde la altura más lejana en el interior de una casa, donde se aprecian la guitarra, guitarrista y escenario con pantalla verde.

realizar cuando el dron está volando muy cerca de ella. Como medida de precaución, también deberá ella tener el controlador del dron lo suficientemente cerca de ella (para poder re-posicionarlo o aterrizarlo), y lo suficientemente lejos para que no se cuele en la grabación. Una vez que Ofelia haya posicionado el dron, debe concentrarse en actuar para la cámara: relajándose, haciendo movimientos precisos, planear hacia dónde se moverá y ella misma notar cuándo ha sido suficiente tiempo de esa escena, de tal modo que hay que regresar al dron para re-posicionarlo.

Música

Una post-inspección a la improvisación de 28 minutos de duración me permite verificar qué tan buena fue mi ejecución del instrumento. En el análisis resaltan:

El tempo. En unas partes es acertado y son rescatables los fragmentos de la grabación. En otras secciones toco con retraso, y en promedio esa es una constante equivocación que pide más práctica.

Las notas de la escala. En una generalidad, casi todas las notas tocadas son correctas y se pueden contar todas las veces en que ejecuto una nota incorrecta.

La calidad de la grabación. Quedó un poco distorsionada, hecho que no es corregible en una postproducción, aunque no arruina totalmente el resultado. La distorsión

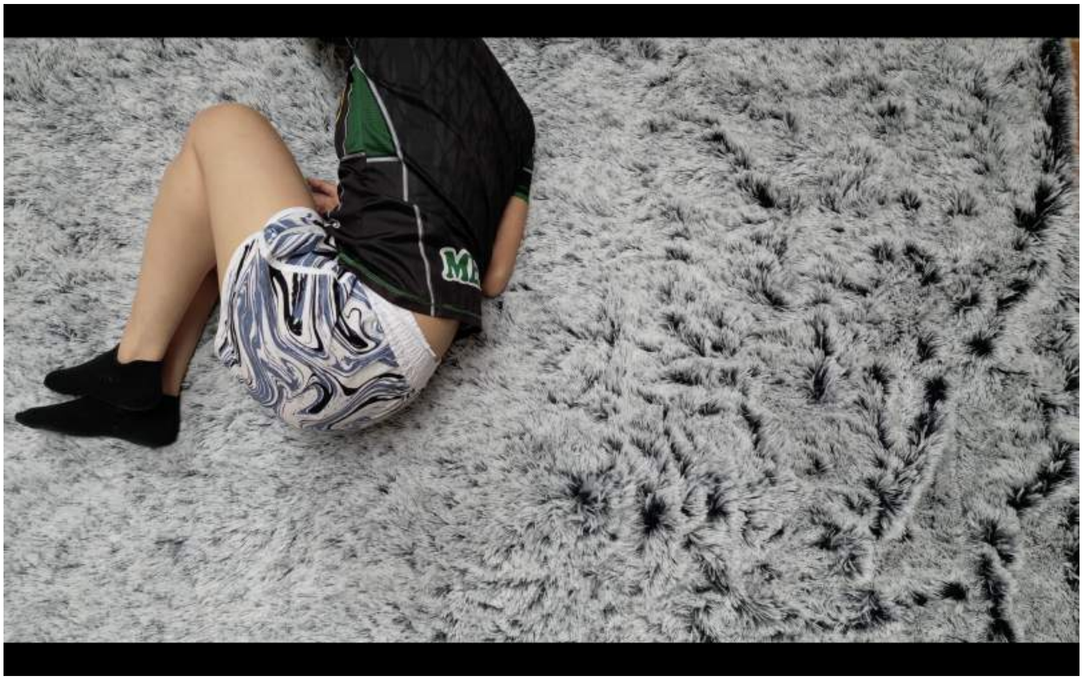


Figura 3.7: Fotografía de una posición corporal tomada en plano cenital, cerca del sujeto.

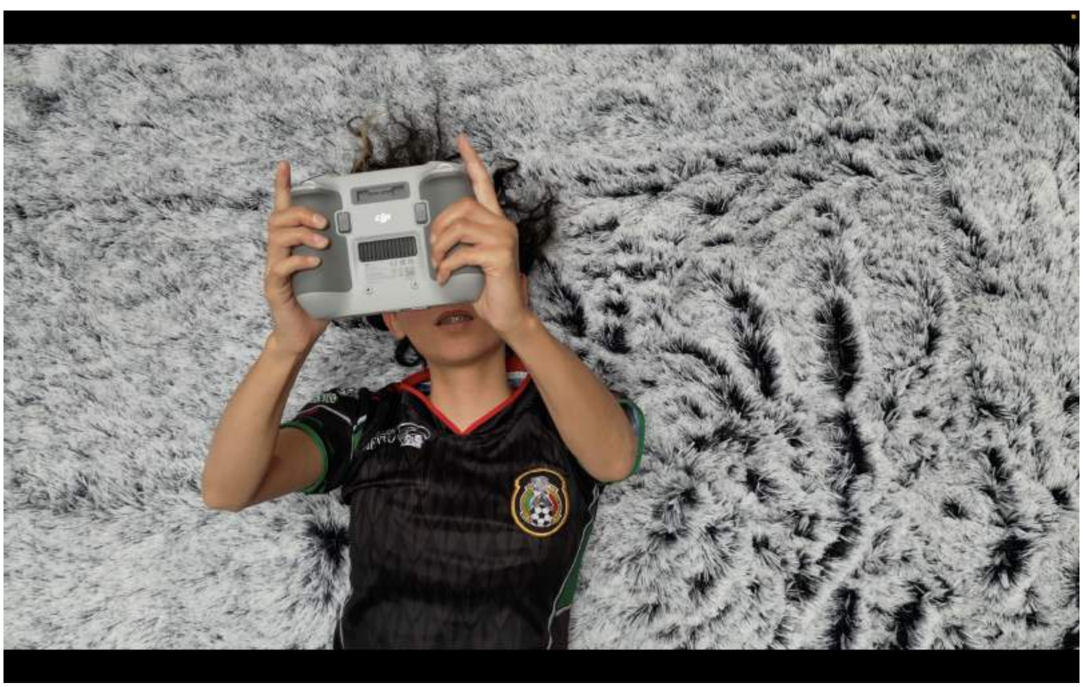


Figura 3.8: Ofelia controlando el dron mientras elige un encuadre.

se debió a la necesidad de tocar con conexiones inalámbricas, ya que, durante una manipulación libre de mi instrumento, perdí de vista la calidad del audio que estaba ingresando a la interfaz.

Conclusivamente, el tempo sí arruina el que consideremos a la grabación completa como una improvisación válida, ya que estos errores los capta el oyente y son molestos. En la postproducción, el tempo se corrige mediante el *warping*; de este modo, la grabación tiene una calidad suficientemente buena como para incluir esos arreglos de guitarra en la pieza musical final. Lo que sí es, por tiempo, imposible de corregir, es el *warping* contra el video grabado de la sesión. Es decir, si se corrige el tempo para el audio, los visuales se desincronizan. Ello sería muy evidente al fijarnos en la digitación por notas que hace la guitarrista en el video. Así pues, tanto la música como el video piden ser editados. Se pudiera exigir de la guitarrista que mejorara su ejecución para ocasiones posteriores, sin embargo, la perfección es prácticamente imposible, de ello que la edición sea considerada una solución eficaz en muchos ámbitos de la producción musical.

Discusión de resultados

La pieza se presta para experimentar qué música de fondo queda mejor; en virtud de que durante la supuesta ejecución de ésta en vivo ocurrieron algunos errores técnicos. Además, una post-inspección al material videográfico permitió dar cuenta que otros tipos de piezas musicales le acompañaban mejor, como son la música clásica, la música *drone*, *jazz*, o un beat más acelerado. Así pues, lo que se puede concluir que debe quedar fijo para desarrollar bien esta pieza es la trama. Añadido a esto, tocar con una preestablecida pista de *live coding* al grabar las escenas de video (aunque al final tal pista no figurase tal cual en el resultado final), y la puesta en práctica varias veces de la idea a ser filmada.

A partir de la trama se ha de buscar un apropiado prospecto de música. Si ésta termina siendo sólo improvisación, sólo precomposición o algo de ambas, ello ha de verificarse en la marcha. La música no necesariamente debe ir primero.

Iniciar este proyecto de manera solista tiene la ventaja de que se puede plantear y perfeccionar mejor una idea que se tiene en mente, respetándose mejor lo que se quiere decir y la forma en que se quiere decir, usando el tiempo y precisión necesarias. Asímismo, las prácticas llegan a ser más relajadas a falta de las presiones de otros. De manera solista, también es fácil notar las restricciones que hay bajo este escenario cuando son manejadas por una única persona, y permite envisionar quién o quiénes podrían hacer falta para mejorar la pieza, o extender el proyecto.

Esto es crucial, pues de no pre-planearse la idea, si sólo juntáramos a un grupo de gente con el pretexto de decidir cómo retratar a una mujer que quiere actuar como hombre, cada quien tendría varias interpretaciones de lo que ello significa; cada quién propondría diferentes cuadros cinematográficos y cada quién elegiría música diferente. Ello seguramente cambiaría la idea original que la autora tenía y se eliminaría mucho del contenido previsto.

4 Conclusiones

Una inspección al qué es la música algorítmica permite aseverar que algoritmos podemos encontrarlos tanto en la música puramente instrumental como en la música programada con ordenador. De este modo, la propuesta de componer para un instrumento como lo es la guitarra eléctrica, tiene validez. En algunos pasajes musicales, el ordenador se usó únicamente como un recurso de acompañamiento, donde quedó invisible en el track final lo que éste tocaba.

El punto fuerte de la presente investigación, en cuanto a composición se refiere, es el de mediar las interacciones entre la música tradicional y la música que usa un ordenador, lo cual presenta un reto al momento de seleccionar qué parámetros de cada una de la partes figurarán en la música. Es decir, necesariamente hay que referir una representación de los algoritmos elegidos para la guitarra y los algoritmos elegidos para el ordenador. Algunos coinciden con nociones comunes. Los conceptos de "partitura", "estándar de jazz", "*timeline*", "macroforma", "mesoforma" y "microforma" han resultado elecciones apropiadas, y por lo tanto se les puso un especial énfasis durante la investigación. Aunado a ello, mis piezas buscan adecuarse al género electrónico pop, cuyo nivel es simple y donde es permisible la síntesis, armonía tonal y guitarra.

Durante la composición de piezas resultó demasiado ambicioso improvisar en forma de *live coding* al mismo tiempo que tocar guitarra. De ello que yo prefiera hablar sobre haber elaborado el "diseño de una plataforma y de un método compositivo para música que emplea programación e instrumentación." Aún así, la pieza "*Tú, Autómata*" muestra lo que la configuración *live coding* vs guitarra en su momento logró.[1] Fue un motivo de perfeccionismo el que me llevó a ir un paso más allá. Busqué estructurar mejor una pieza musical a partir de la sencillez, reduciendo las variables a controlar y enfocándome en explotar aquéllas seleccionadas.

Una solución fue la de considerar solamente *loops* de *live coding* (al estilo del DJ), o en su caso, usar un único canal para el *live coding* al momento de una presentación en vivo, ya que la ejecución en guitarra es muy complicada, y limitar la cantidad de *live coding* ayuda a estructurar una pieza.

[1] "*Tú Autómata*", video de *YouTube*

Un punto básico de comunicación entre un lenguaje de *live coding* y un instrumento musical es el tempo. Intenté exigirme lo suficiente como ejecutante de guitarra para poder establecer una sincronización certera, y sirve aquí también incluir una predefinida estructura tipo "ABABCB". De este modo, la macroforma queda definida, tal cual sucede con los estándares de jazz.

Entonces, aunque se habla de usar *live coding*, el cómo estamos utilizándolo en mi música resulta más relevante. Es un generador rítmico de sonido que fue elegido por su versatilidad, y por las sensaciones y timbres que se le puede asociar. Sin embargo, sacrificamos algo de su significado al mezclarlo con el resto de la música, quitándole cierta importancia al "código" para adecuarlo a mi contexto de creación de piezas originales. Aunque el *live coding* es una novedosa herramienta contemporánea, lo que por música se entiende ya existía desde antes, y tal bagaje previo fue lo que me guió para el cómo incluirlo en mi música. Asimismo mis influencias musicales juegan aquí un papel importante.

Se rescató el concepto de *timeline*, y más específicamente el de ritmo euclidiano para con éste proponer música que girase a su alrededor. Se trata de un algoritmo fuertemente asociado al lenguaje de programación *TidalCycles*. Por lo tanto, es una manera de asegurar que este lenguaje intervenga desde un momento en el proceso de composición de piezas. Ello invita a construir esquemas musicales que puedan en teoría complejizarse en ambos ámbitos, tanto en el instrumental como el programático. Y como ya varias veces se ha mencionado, el ritmo euclidiano basta para caracterizar algunas formas y estilos musicales bailables.

Finalmente, quiero destacar que la bailabilidad, siendo entendida como un algoritmo o conjunto de algoritmos reactivos, ha sido subyacente a la creación de mi material. Es decir, hay una cierta dificultad para crear una música bailable usando lenguajes de programación musicales. Una precisa selección de figuras y sonoridades musicales implica que alguien comience o no a bailar (y así permanezca). Quise entonces encontrar estos algoritmos causantes de euforia y esencialmente aislarlos de aquéllos otros discursos y contextos con los que se les relaciona en el ámbito de lo popular. Ello a su vez permite enfocarnos en el fenómeno cognitivo por sobre el fenómeno social de la publicidad, los estereotipos, la industria, etc.

La presente tesis sin duda ahonda en cuestiones que difícilmente llegan a reflejarse en una presentación musical o discografía. Específicamente, las influencias matemáticas han quedado aquí documentadas, pero el resultado sonoro puede entenderse autónomamente y llegar a cambiársele el significado. Una audiencia que escucha la obra puede considerar que lo que escucha es únicamente música, y esto en realidad sucede frecuentemente en el mundo musical, en particular lo que termina siendo observado desde el punto de vista de un producto finalizado. Así pues, es parte del artista explicar su música a partir de otros formatos como lo sería la prosa o la entrevista.

A Solución al problema matemático

En el Capítulo 1, Sección 1.3.5, se menciona a grandes rasgos la solución a un problema de cadenas de Markov en el largo plazo. En este apéndice el mismo problema se comenta más a detalle.

A.1. Matrices de transición regulares

A una matriz de transición $\mathbf{P} = (P_{ij})$ con un número finito de estados $1, \ldots, N$, se le llama **regular** si existe algún número k tal que las entradas de $\mathbf{P}^k$ sean estrictamente positivas. Es decir, $P_{ij} > 0 \quad \forall\ i, j$.

Cuando una cadena de Markov es regular, hay un importante resultado que establece que "siempre existe para ella una **distribución de probabilidad límite** $\pi = (\pi_1, \ldots, \pi_N)$ tal que $\pi_j > 0\ \forall j \in \{1, \ldots, N\}$ y $\sum_j \pi_j = 1$. La distribución es independiente del estado inicial."

Más formalmente, para una matriz de transición de probabilidades regular se tiene la convergencia:

$$\lim_{n\to\infty} P_{ij}^n = \pi_j > 0 \quad \forall j \in \{1, \ldots, N\}.$$

Lo que significa que, en el largo plazo (cuando n tiende a infinito), la probabilidad de encontrar una cadena de Markov en el estado j se aproxima a π_j, y no importa en qué estado de la cadena se empezó.

A.2. Matrices doblemente estocásticas

A una matriz de probabilidades de transición se le llama doblemente estocástica si cada una de sus entradas es mayor o igual a cero, y si la suma de cada renglón y columna es igual a 1. Formalmente, $\mathbf{P} = (P_{ij})$ es doblemente estocástica si

$$P_{ij} \geq 0 \qquad \text{y} \qquad \sum_k P_{ik} = \sum_k P_{kj} = 1 \qquad \forall \ i, j.$$

Considere una matriz de probabilidades de transición doblemente estocástica de N estados $1, \ldots, N$. Si la matriz es regular, entonces la única distribución límite es la distribución uniforme $\pi = (\frac{1}{N}, \frac{1}{N}, \ldots, \frac{1}{N})$. Ya que sólo hay una solución a $\pi_j = \sum_k \pi_k P_{kj}$ y $\sigma_k \pi_k = 1$ cuando P es regular, únicamente debemos verificar que $\pi = (\frac{1}{N}, \frac{1}{N}, \ldots, \frac{1}{N})$ es una solución con $\mathbf{P}$ doblemente estocástica. Pero ya que una matriz doblemente estocástica siempre cumple que $\sum_j P_{jk} = 1$, se cumple que

$$\frac{1}{N} = \sum_j \frac{1}{N} P_{jk} = \frac{1}{N}.$$

Para ejemplificar, consideremos el siguiente problema:

Sea Y_n la suma de N tiradas de dado independientes. Determina, en el largo plazo, con qué probabilidad Y_n resulta ser múltiplo de 7, considerando que el estado inicial es 1.

Solución

Primero propongamos una matriz de transición que sirva:

Sea X_n el residuo cuando Y_n es dividido por 7. Entonces X_n es una cadena de Markov en los estados $0, 1, \ldots, 6$ cuya matriz de probabilidades de transición es

$$\mathbf{P} = \begin{pmatrix} 0 & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\ \frac{1}{6} & 0 & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{6} & 0 & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & 0 & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & 0 & \frac{1}{6} & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & 0 & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & 0 \end{pmatrix}. \tag{A.1}$$

Analicemos más detalladamente por qué la elección de la matriz.

Observamos que, ya que un dado tiene seis caras, hay seis resultados posibles de obtenerse y es igualmente probable sacar $1, 2, 3, 4, 5$ o 6. Así, al lanzar un dado, la probabilidad de que ocurra cualquiera de estos eventos es $\frac{1}{6}$.

Sea Y_n la suma de las N tiradas de dado que se han realizado. Y_n es multiplo de 7 o no lo es. Es decir, al dividir Y_n entre 7 obtendremos un residuo, que es un entero $X_n \in \{\overline{0}, \overline{1}, \overline{2}, \overline{3}, \overline{4}, \overline{5}, \overline{6}\}$. De este modo, podemos preguntarnos cuál es la probabilidad de que si estoy en $X_n = \overline{0}$, se obtenga un múltiplo de 7 en la siguiente tirada. Y cuál es la probabilidad de que si estoy en $X_n = \overline{1}$, se obtenga un múltiplo de 7 en la siguiente tirada. Y así sucesivamente con $X_n = \overline{2}$, $X_n = \overline{3}$, $X_n = \overline{4}$, $X_n = \overline{5}$ y $X_n = \overline{6}$. Cada renglón en la matriz representará lo que puede pasar en la siguiente tirada del dado cuando me encuentro en cierto X_n específico.

Por ejemplo, si $Y_n = 52$, entonces $X_n = \overline{3}$ (pues $52 = 7 \cdot 7 + 3$). Y es así que, si en la $(N+1)$-ésima tirada de los dados se obtuviera un 4, Y_{N+1} sería múltiplo de 7. Por lo tanto, hay un sexto de probabilidad de que en el juego siguiente, del estado $\overline{3}$ se vaya al estado $\overline{0}$.

Notemos que es imposible que en juegos consecutivos, el estado $\overline{3}$ vaya a sí mismo (pues eso requeriría obtener un 7 en el lanzamiento del dado); y lo mismo pasa en general, de ahí que, en la matriz de transición, haya ceros en la diagonal.

Ahora, para entender el resto del problema, nos dan 1 como estado inicial. Esto se puede interpretar como que "se lanzó una vez el dado y se obtuvo un uno", o como "antes de iniciar la partida, se coloca el dado mostrando el 1." Formalmente, denotaremos este estado inicial por el vector $X_1 = (0, 1, 0, 0, 0, 0, 0)$. Esto simplemente significa que, como al iniciar el juego, obtener 1 siempre ocurre, entonces la probabilidad de que el dado muestre un 1 siempre es 1, y en todos los demás casos es cero.

Ahora, sean $\mathbf{P}$ la matriz de transición y X_1 el vector inicial. Podemos obtener el estado X_2 al multiplicar $\mathbf{P}X_1$; de este modo, $X_2 = (\frac{1}{6}, 0, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6})$.

Si ahora multiplicamos $\mathbf{P}X_2$, obtenemos que $X_3 = (\frac{1}{9}, \frac{5}{36}, \frac{1}{9}, \frac{1}{9}, \frac{1}{9}, \frac{1}{9}, \frac{1}{9})$. Con esto sabemos que la probabilidad de obtener un múltiplo de 7 en la tercer tirada del dado (sabiendo que la primera vez se obtuvo uno) es $\frac{1}{9}$.

De modo que siempre podemos encontrar el vector de estados X_n en el tiempo N, si conocemos el estado anterior X_{n-1}, pues $X_n = \mathbf{P}X_{n-1}$.

Pero si N es muy grande y sólo nos dan el vector inicial, sería costoso (a veces imposible) calcular cada X_{n-i}. En este caso mejor sería utilizar el criterio de convergencia dado, y lo que éste nos dice es que si la matriz es doblemente estocástica y regular (esto es, $\mathbf{P}^2$ tiene entradas estrictamente positivas), cuando hacemos N tender a infinito, X_n se aproxima a una distribución límite $\pi = (\frac{1}{N}, \frac{1}{N}, \ldots, \frac{1}{N})$, y esto ocurre indepndientemente del estado inicial. Así, aplicando directamente el resultado al problema, obtenemos $\pi = (\frac{1}{7}, \frac{1}{7}, \ldots, \frac{1}{7})$, ya que la matriz de transición sí es doblemente estocástica y regular. El vector π también nos dice lo que sucede entrada a entrada. De este modo, como Y_n es múltiplo de 7 si y sólo si $X_n = \overline{0}$, la primera entrada de π verifica que, en el largo plazo (es decir, cuando N es muy grande), la probabilidad de ser múltiplo de 7 es $\frac{1}{7}$.

Bibliografía

[Ames, 1989] Ames, C. (1989). The Markov Process as a compositional model: A Survey and Tutorial. *Leonardo Music Journal*, 22(2):175–187.

[Arlandi, 1996] Arlandi, G. F. (1996). Symmetry-Canon: Music and Mathematics, Painting and Graphicization "Perpetuum Mobile". In *Katachiu Symmetry*, pages 117–124. Springer.

[Ashton, 2003] Ashton, A. (2003). *Harmonograph: A visual guide to the mathematics of music*. Bloomsbury Publishing USA.

[Benson, 2006] Benson, D. (2006). *Music: a mathematical offering*. Cambridge University Press.

[Boolos et al., 2002] Boolos, G. S., Burgess, J. P., and Jeffrey, R. C. (2002). *Computability and logic*. Cambridge university press.

[Bravo et al., 2006] Bravo, A., Rincón, H., and Rincón, C. (2006). Álgebra superior.

[Burkholder et al., 2019] Burkholder, J. P., Grout, D. J., and Palisca, C. V. (2019). *A history of western music: Tenth international student edition*. WW Norton & Company.

[Chadabe, 1997] Chadabe, J. (1997). *Electric Sound: The Past and Promise of Electronic Music*. Prentice Hall.

[Chollet, 2023] Chollet, M. (2023). *Reinventar el amor: Cómo el patriarcado sabotea las relaciones heterosexuales*. Paidós.

[Collins and McLean, 2014] Collins, N. and McLean, A. (2014). Algorave: Live performance of algorithmic electronic dance music. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 355–358.

[Corona and de los Ángeles Ancona, 2011] Corona, M. A. and de los Ángeles Ancona, M. (2011). *Diseño de algoritmos y su codificación en lenguaje C*. McGraw Hill.

[DuBreuil, 2020] DuBreuil, A. (2020). *Hands-On Music Generation with Magenta: Explore the role of deep learning in music generation and assisted music composition*. Packt Publishing.

[Frenkel, 2013] Frenkel, E. (2013). *Love and math: The heart of hidden reality*. Basic Books.

[Frenkel, 2014] Frenkel, E. (2014). Los Angeles Times. Accessed on March 2, 2014.

[Galilei, 2010] Galilei, G. (2010). *Diálogos sobre los sistemas del mundo*. Editorial Maxtor.

[Haack, 1991] Haack, J. K. (1991). Clapping music—a combinatorial problem. *The College Mathematics Journal*, 22(3):224–227.

[Hagan, 2005] Hagan, K. (2005). Genetic Analysis of Analogique B. *EMS : Electroacoustic Music Studies Network*.

[Hollings et al., 2018] Hollings, C., Martin, U., and Rice, A. C. (2018). *Ada Lovelace: The making of a computer scientist*. Bodleian Library Oxford.

[Johnston, 2020] Johnston, B. L. (2020). *Numbers and symmetry: an introduction to algebra*. CRC Press.

[Kahneman, 2017] Kahneman, D. (2017). *Thinking, fast and slow*.

[Keith, 1991] Keith, M. (1991). *From Polychords to Polya: Adventures in Musical Combinatorics*. Vinculum Press.

[Lazzarini et al., 2016] Lazzarini, V., Yi, S., et al. (2016). *Csound: A sound and music computing system*. Springer.

[Louridas, 2017] Louridas, P. (2017). *Real-world Algorithms: A Beginner's Guide*. MIT Press.

[Loy, 2007a] Loy, G. (2007a). *Musimathics, Volume 1: The Mathematical Foundations of Music*. MIT Press.

[Loy, 2007b] Loy, G. (2007b). *Musimathics, Volume 2: The Mathematical Foundations of Music*. MIT Press.

[Luque, 2009] Luque, S. (2009). The Stochastic Synthesis of Iannis Xenakis. *Leonardo Music Journal*, 19:77–84.

[MacLane, 2012] MacLane, S. (2012). *Mathematics form and function*. Springer Science & Business Media.

[Mannone et al., 2018] Mannone, M., Kitamura, E., Huang, J., Sugawara, R., and Kitamura, Y. (2018). Musical combinatorics, tonnetz, and the cubeharmonic. In

Collected Papers of the Academy of Arts of Novi Sad, pages 104–116. University of Novi Sad.

[Mayer, 2019] Mayer, D. (2019). PbindFX: an interface for sequencing effect graphs in the Supercollider audio programming language. *AM'19: Proceedings of the 14th International Audio Mostly Conference: A Journey in Sound*, page 287–291.

[Mayer, 2022] Mayer, D. (2022). Algorithms in Sound Synthesis, Processing, and Composition: a Dialectic Game. *Almat 2020 - symposium on algorithmic agency in artistic practice, Research Catalogue.*

[McLean, 2014] McLean, A. (2014). Making programming languages to dance to: live coding with tidal. In *Proceedings of the 2nd ACM SIGPLAN international workshop on Functional art, music, modeling & design*, pages 63–70.

[McLean et al., 2018] McLean, A., Dean, R., et al. (2018). *The Oxford handbook of algorithmic music.* Oxford University Press.

[Miranda, 1998] Miranda, E. R. (1998). *Computer sound synthesis for the electronic musician.* Butterworth-Heinemann.

[Morales et al., 2011] Morales, E. O., Reinoso, J. S. S., Maldonado, M. M. S., Haro, C. E. R., and Iñiguez, J. D. B. (2011). Análisis del Efecto Mozart en el desarrollo intelectual de las personas adultas y niños. *Ingenius*, (5):45–54.

[Nierhaus, 2009] Nierhaus, G. (2009). *Algorithmic Composition. Paradigms of Automated Music Generation.* Springer-Verlag, Germany.

[Nierhaus et al., 2015] Nierhaus, G. et al. (2015). *Patterns of Intuition. Musical Creativity in the Light of Algorithmic Composition.* Springer, Germany.

[Noll et al., 2006] Noll, T., Andreatta, M., and Agon, C. (2006). Computer-aided transformational analysis with tone sieves. *SMC 06.*

[Parviainen, 2017] Parviainen, T. (2017). How Generative Music Works: A Perspective.

[Petzold, 2008] Petzold, C. (2008). *The annotated Turing: a guided tour through Alan Turing's historic paper on computability and the Turing machine.* Wiley Publishing.

[Pinsky, 2011] Pinsky, K. (2011). *An Introduction to Stochastic Modeling.* Elsevier, USA, Fourth Edition edition.

[Ramírez-Galarza and Seade-Kuri, 2002] Ramírez-Galarza, A. I. and Seade-Kuri, J. (2002). *Introducción a la geometría avanzada.* Las prensas de ciencias.

[Roads, 1996] Roads, C. (1996). *The computer music tutorial.* MIT press.

[Roads, 2001] Roads, C. (2001). *Microsound.* MIT Press, Cambridge.

[Ruskey and Sawada, 1999] Ruskey, F. and Sawada, J. (1999). An efficient algorithm for generating necklaces with fixed density. *SIAM Journal on Computing*, 29(2):671–684.

[Russ, 2004] Russ, M. (2004). *Sound synthesis and sampling.* Taylor & Francis.

[Seeger, 1958] Seeger, C. (1958). Prescriptive and descriptive music-writing. *The Musical Quarterly*, 44(2):184–195.

[Serra, 1993] Serra, M.-H. (1993). Stochastic Composition and Stochastic Timbre: GENDY3 by Iannis Xenakis. *Perspectives of New Music*, 31(1):236–257.

[Shepard, 2013] Shepard, B. K. (2013). *Refining sound: A practical guide to synthesis and synthesizers.* Oxford University Press.

[Slonimsky, 1946] Slonimsky, N. (1946). The Schillinger system of musical composition.

[Stewart, 2022] Stewart, I. (2022). *Galois theory.* CRC press.

[Toussaint, 2005] Toussaint, G. (2005). The euclidean algorithm generates traditional musical rhythms. *Renaissance Banff: Mathematics, Music, Art, Culture*, pages 47–56.

[Toussaint, 2019] Toussaint, G. T. (2019). *The geometry of musical rhythm: what makes a good rhythm good?* CRC Press.

[Tubau, 2013] Tubau, D. (2013). *Las paradojas del guionista: reglas y excepciones en la práctica del guión.* Alba Editorial.

[Volchenkov and Dawin, 2012] Volchenkov, D. and Dawin, J. R. (2012). Musical Markov Chains. *International Journal of Modern Physics: Conference Series*, 16:116–135.

[Xenakis, 1992] Xenakis, I. (1992). *Formalized music: thought and mathematics in composition.* Pendragon Press.