



UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH



PROCESADO DE PARES DE SEÑALES EN COORDENADAS POLARES: APLICACIÓN MUSICAL

A Degree Thesis

Submitted to the Faculty of the

Escola Tècnica d'Enginyeria de Telecomunicació de Barcelona

Universitat Politècnica de Catalunya

by

Clara Vila Brustenga

In partial fulfilment

of the requirements for the degree in

**TELECOMMUNICATION AUDIOVISUAL SYSTEMS
ENGINEERING**

Advisor: Philippe Salembier

Barcelona, June 2020

Abstract

In this work we present the result of applying signal processing techniques to music. The whole development is based on modular synthesis, for which we have used a free code software called VCV Rack that simulates a hardware synthesizer and allows the user not only to manipulate it but also to implement new modules. More specifically we study what effect and what utility can have, the fact of treating the signals in pairs and filtering or distorting its module or its phase. It is an experimental work that aims to explore a type of processing that is currently not very deep in the world of music synthesis.



Resum

En aquest treball es presenta el resultat d'aplicar tècniques de processat de senyal dins del món de la música. Tot el desenvolupament es basa en la síntesi musical, per el qual hem utilitzat un Software de codi lliure, el VCV Rack que simula un sintetitzador Hardware i permet al usuari tant manipular-lo com implementar nous mòduls. Més concretament, estudiem quin efecte i quina utilitat pot tenir el fet de tractar els senyals per parells i filtrar o distorsionar el seu mòdul o la seva fase. Es un treball experimental que pretén explorar un tipus de processament poc estudiat actualment en el món de la síntesi musical.



Resumen

En este trabajo se presenta el resultado de aplicar técnicas de procesado de señal a la música. Todo el desarrollo se basa en la síntesis modular, para lo cual hemos utilizado un Software de código libre llamado VCV Rack que simula un sintetizador Hardware y permite al usuario no solo manipularlo sino también implementar nuevos módulos. Más concretamente estudiamos qué efecto y que utilidad puede tener, el hecho de tratar las señales por pares y filtrar o distorsionar su módulo o su fase. Es un trabajo experimental que pretende explorar un tipo de procesado poco profundizado actualmente en el mundo de la síntesis musical.

Agradecimientos

En primer lugar, me gustaría agradecer a mi familia y amigos por darme su apoyo incondicional a lo largo de toda la carrera. No ha sido siempre fácil, y en los momentos donde me he querido rendir, su cariño y fuerza me ha ayudado a seguir con una sonrisa. Me gustaría mencionar especialmente a mi madre, quien ha celebrado mis éxitos como si fueran suyos, y nunca ha dejado que los momentos complicados me derrumbaran. Ha sido siempre un ejemplo de fortaleza y superación y quiero agradecer el gran ejemplo con el que he tenido la suerte de crecer.

Por otro lado, me gustaría agradecer a mi tutor Philippe Salembier, quién ha tenido una enorme paciencia conmigo y me ha ayudado siempre que lo he necesitado. Nunca me he sentido abandonada frente a este trabajo y ha dado lo mejor de él para que pudiésemos avanzar.

Historial de revisiones y registro de aprobación

Revisión	Fecha	Propósito
0	23/05/2020	Creación del documento
1	22/06/2020	Revisión del documento
2	26/06/2020	Revisión del documento

LISTA DE DISTRIBUCIÓN DEL DOCUMENTO

Nombre	e-mail
Clara Vila	cvilabrustenga@gmail.com
Philippe Salembier	philippe.salembier@upc.edu

Escrito por:		Revisado y aprobado por:	
Date	25/06/2020	Date	26/06/2020
Name	Clara Vila Brustenga	Name	Philippe Salembier
Position	Autora del proyecto	Position	Supervisor del proyecto

Tabla de contenido

Abstract	1
Resum	2
Resumen	3
Agradecimientos	4
Historial de revisiones y registro de aprobación	5
Tabla de contenido	6
Lista de figuras	7
Lista de tablas	10
1. Introducción	11
1.1 Propósito de la declaración	11
1.2 Especificaciones y requisitos	11
1.3 Plan de trabajo	12
1.4 Incidencias respecto al plan inicial.....	12
2. Uso de tecnologías utilizadas o aplicadas en esta tesis	13
2.1 Primeros pasos hacia el sintetizador.....	13
2.2 El primer sintetizador.....	14
2.3 VCV Rack.....	14
3. Metodología y desarrollo:	16
3.0 Contexto	16
3.1 Oscilador CLAMAR.....	18
3.2 Módulos conversores de coordenadas	20
4. Resultados	22
4.1 Caso 1: Escalamos la fase de un coseno y su señal desfasada	22
4.2 Caso 2: Filtramos la fase de una señal diente de sierra y un triángulo.....	26
4.3 Caso 3: Distorsión módulo de un triángulo y un triángulo desfasado.....	30
5. Budget	34
6. Conclusiones y desarrollo futuro:	35
Bibliography:	36

Lista de figuras

<i>Figura 1.3.1 Diagrama de Gantt.....</i>	<i>13</i>
<i>Figura 2.1.1 Telharmonium.....</i>	<i>14</i>
<i>Figura 2.2.1 Moog Modular.....</i>	<i>15</i>
<i>Figura 2.2.2 Mini-Moog.....</i>	<i>15</i>
<i>Figura 2.3.1 Ejemplo de patch en el software VCV Rack.....</i>	<i>16</i>
<i>Figura 3.0.1 Esquema del sistema a estudiar cogiendo como ejemplo como el filtrado..</i>	<i>17</i>
<i>Figura 3.0.2 Señales X e Y mostradas en el display del osciloscopio en formato temporal.....</i>	<i>18</i>
<i>Figura 3.0.3 Señales X e Y mostradas en el display del osciloscopio en formato vectorial</i>	<i>18</i>
<i>Figura 3.0.4 Espectro de la señal X.....</i>	<i>18</i>
<i>Figura 3.1.1 Esquemmatización del método para generar las formas de onda del oscilador CLAMAR. Vemos a la izquierda la señal diente de sierra, en el centro la señal cuadrada y a la derecha la señal triangular.....</i>	<i>19</i>
<i>Figura 3.1.2 Señal Coseno en temporal.....</i>	<i>20</i>
<i>Figura 3.1.3 Señal Triangular en temporal.....</i>	<i>20</i>
<i>Figura 3.1.4 Señal Cuadrada en temporal.....</i>	<i>20</i>
<i>Figura 3.1.5 Señal Diente de Sierra en temporal.....</i>	<i>20</i>
<i>Figura 3.1.6 Señal Coseno y su señal desfasada en tempora.....</i>	<i>20</i>
<i>Figura 3.1.7 Señal Cuadrada y su señal desfasada en temporal.....</i>	<i>21</i>
<i>Figura 3.1.8 Señal Triangular y su señal desfasada en temporal.....</i>	<i>21</i>
<i>Figura 3.1.9 Señal Diente de Sierra y su señal desfasada en temporal.....</i>	<i>21</i>
<i>Figura 3.1.10 Señal Coseno y su señal desfasada en vectorial.....</i>	<i>21</i>
<i>Figura 3.1.11 Señal Cuadrada y su señal desfasada en vectorial.....</i>	<i>21</i>
<i>Figura 3.1.12 Señal Triangular y su señal desfasada en vectorial.....</i>	<i>21</i>
<i>Figura 3.1.13 Señal Diente de Sierra y su señal desfasada en vectorial.....</i>	<i>21</i>
<i>Figura 4.0.1 Diagrama de bloques que ilustra como se modifican las diferentes características de un sonido.....</i>	<i>23</i>

<i>Figura 4.1.1 Señales reconstruidas a la salida del sistema escalando el módulo.....</i>	<i>24</i>
<i>Figura 4.1.2 Esquema básico del caso 1.....</i>	<i>24</i>
<i>Figura 4.1.3 Señales de entrada X e Y con desfase de 90°.....</i>	<i>24</i>
<i>Figura 4.1.4 Módulo (azul) y fase (rojo) de las señales de entrada.....</i>	<i>24</i>
<i>Figura 4.1.5 Señales de entrada X e Y en vectorial.....</i>	<i>25</i>
<i>Figura 4.1.6 Fase original F (azul) y fase escalada F' (rojo)</i>	<i>25</i>
<i>Figura 4.1.7 Señales reconstruidas. En azul tenemos la señal X' y en rojo la señal Y' ...</i>	<i>25</i>
<i>Figura 4.1.8 Espectro frecuencial la señal X'.....</i>	<i>25</i>
<i>Figura 4.1.9 Señales de entrada, X e Y en formato vectorial.....</i>	<i>26</i>
<i>Figura 4.1.10 Señales de salida X' e Y' en formato vectorial.....</i>	<i>26</i>
<i>Figura 4.1.11 Señal fase F (azul) y señal fase escalada F' (rojo)</i>	<i>26</i>
<i>Figura 4.1.12 Señales reconstruidas a la salida. En azul tenemos la señal X' y en rojo la señal Y'.....</i>	<i>26</i>
<i>Figura 4.1.13 Espectro frecuencial de la señal X' a la salida.....</i>	<i>26</i>
<i>Figura 4.2.1 Espectros del módulo (en verde) y de la fase (en lila) de las señales diente de sierra y triangular desfasadas.....</i>	<i>27</i>
<i>Figura 4.2.2 Salidas X' e Y' al filtrar el módulo M con un filtro paso alto.....</i>	<i>27</i>
<i>Figura 4.2.3 Salidas X' e Y' al filtrar el módulo M con un filtro paso bajo</i>	<i>27</i>
<i>Figura 4.2.4 Esquema básico del sistema del caso 2.....</i>	<i>28</i>
<i>Figura 4.2.5 Señales de salida con desfase = 0.1. En rojo tenemos la señal X' y en azul la señal Y'.....</i>	<i>28</i>
<i>Figura 4.2.6 Señales de salida con desfase = 0.25. En rojo tenemos la señal X' y en azul la señal Y'.....</i>	<i>28</i>
<i>Figura 4.2.7 Señales de salida con desfase = 0.8. En rojo tenemos la señal X' y en azul la señal Y'.....</i>	<i>28</i>
<i>Figura 4.2.8 Espectros frecuenciales de las señales diente de sierra X en rosa y triangular Y en verde.....</i>	<i>29</i>
<i>Figura 4.2.9 Espectro de la fase.....</i>	<i>29</i>
<i>Figura 4.2.10 Forma de onda de la fase.....</i>	<i>29</i>
<i>Figura 4.2.11 Espectro frecuencial de las salidas X' (verde) e Y' (rojo) con una frecuencia de corte de 1,5 kHz i desfase de 36°.....</i>	<i>30</i>

<i>Figura 4.2.12 Espectro frecuencial de las salidas X' (verde) e Y' (rojo) con una frecuencia de corte de 1,5 kHz i desfase de 90°</i>	30
<i>Figura 4.3.1 Ilustración del wavefolding en una senoide</i>	31
<i>Figura 4.3.2 Coseno (rojo) y coseno distorsionado por wavefolding (azul)</i>	31
<i>Figura 4.3.3 Espectro de un coseno (verde) y del coseno distorsionado por wavefolding (rojo)</i>	31
<i>Figura 4.3.4 Modulo M (azul) y fase F (azul) de las señales X e Y</i>	32
<i>Figura 4.3.5 Esquema básico del caso 3</i>	32
<i>Figura 4.3.6 Formas de onda de la señal X después de ser distorsionada por wavefolding. En la imagen de la izquierda con una ganancia mínima y en la imagen de la derecha con una ganancia de 3.15</i>	32
<i>Figura 4.3.7 Señal del módulo M (rojo) y la señal M' (azul)</i>	33
<i>Figura 4.3.8 Señales a la salida X' (azul) e Y' (rojo)</i>	33
<i>Figura 4.3.9 Espectro frecuencial de la señal a la salida X'</i>	33
<i>Figura 4.3.10 Señales X e Y en formato vectorial a la izquierda y señales de salida X' e Y' en formato vectorial a la derecha</i>	33
<i>Figura 4.3.11 Modulo M (azul) y modulo M' (rojo)</i>	34
<i>Figura 4.3.12 Señales a la salida X' (azul) e Y' (rojo) en formato temporal</i>	34
<i>Figura 4.3.13 Señales a la salida X' e Y' en formato vectorial</i>	34
<i>Figura 4.3.14 Espectro de la señal X' a la salida</i>	34

Lista de tablas

Tabla 3.1.1 Tabla del libro Developing Virtual Synthesizer with VCV Rack de Leonardo Gabrielli que muestra la media del tiempo de ejecución de la señal diente de sierra con y sin sobremuestro.....20

1. Introducción

Un sintetizador modular es un sintetizador físico o virtual que crea sonidos musicales gracias a módulos independientes que sintetizan el sonido pasando la fuente por los distintos módulos.

El usuario los conecta a través de matrices o interruptores para crear algún tipo de sonido existente o no y a este tipo de combinaciones les llamaremos “patch”.

Los sintetizadores modulares son tanto originadores como manipuladores de sonido, que se disponen como un conjunto de módulos donde cada uno realiza un trabajo único, similar a una función matemática.

Los módulos individuales se conectan a través de cables, conmutadores y sliders. De esta forma es posible acceder a un gran número de opciones de conexión y permutación.

1.1 Propósito de la declaración

Este trabajo tiene como objetivo final explorar áreas de la síntesis musical en las que no se ha profundizado hasta ahora. Más concretamente, estudiaremos el efecto que puede tener el hecho de manipular pares de señales en coordenadas polares. La síntesis de sonido es el método por el cual se consiguen sonidos “musicales” a partir de señales analógicas y posibles variaciones en dichas señales (adición, sustracción, modulación, filtrado...).

En este proyecto se ha trabajado mediante un simulador software del sintetizador modular clásico (hardware), a través del programa VCV Rack. Este programa no solo permite descubrir el mundo de la síntesis musical, sino que también permite al usuario desarrollar sus propios módulos e implantarlos para que puedan formar parte de los diferentes patches.

1.2 Especificaciones y requisitos

Debido al contenido y características de este proyecto no hay medidas numéricas que permitan comprobar la calidad del funcionamiento de nuestras implementaciones. De todos modos, este proyecto sí exige terminar con módulos que funcionen y de fácil uso para un usuario externo al proyecto. Para facilitar dicha tarea, cada módulo viene acompañado de un manual de instrucciones.

Dichos módulos permiten explorar el procesado por pares de señales en coordenadas polares en un contexto musical. El resultado final consta de tres módulos:

-Generador de señal: Oscilador que consta de 4 salidas con diferente forma de onda y sus respectivas salidas desfasadas. Se inicia con un desfase de 90° pero se puede modificar.

-Módulos conversores: Un módulo que pasa de entrada con coordenadas cartesianas a salidas en coordenadas polares y otro módulo que hace el proceso inverso.

1.3 Plan de trabajo

Las tareas planteadas desde el inicio:

- Introducirme al Software que voy a estar utilizando durante el proyecto, VCV Rack
- Gracias a esta introducción y a la documentación leída, definir el alcance del proyecto y los objetivos.
- Documentarme de manera más exhaustiva sobre la teoría que hay detrás de la síntesis modular.
- Seguir adentrándome en el mundo de VCVRack, haciendo mis propios patches y aprendiendo cómo funcionan los módulos ya existentes.
- Crear un entorno de programación y probar con un primer módulo sencillo
- Asentar las bases teóricas sobre Procesado de Señal que necesitaré para crear mis módulos
- Programar y utilizar los módulos que voy a crear: Generador de sonido y conversores de coordenadas
- Escribir mi tesis y hacer la presentación final.

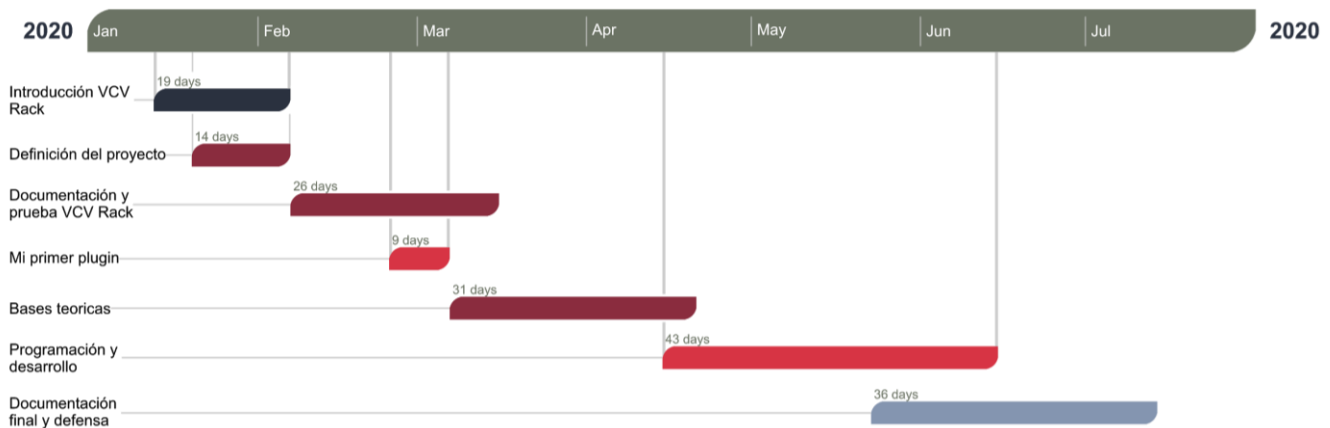


Figura 1.3.1 Diagrama de Gantt

1.4 Incidencias respecto al plan inicial

No ha habido grandes desviaciones o incidencias que nos hayan hecho hacer un replanteamiento completo del trabajo. De todos modos, cabe destacar la gran crisis mundial en la que nos encontramos inmersos. Ha tenido una gran influencia no solo porque nos ha obligado a reinventar la manera de trabajar sino también por el gran peso emocional que ha podido tener.

También cabe destacar ciertos problemas con el programa y mi sistema operativo que me han atrasado en ciertas tareas en cuanto empecé con el bloque de programación y desarrollo.

2. Uso de tecnologías utilizadas o aplicadas en esta tesis

Para entender mejor el sujeto del proyecto, es importante hacer un breve viaje al pasado. Cuando hablamos de sonido sintético, nos referimos a una alteración que se crea gracias a un voltaje, que permitirá alterar la membrana de un altavoz formando dichas alteraciones en la presión.

Desde su invención en los años 60, el sintetizador ha sido un instrumento básico para generar cualquier sonido de manera electrónica, como bases musicales, efectos especiales o sonidos para videojuegos. Debido a la revolución digital de nuestra era, el primer sintetizador analógico que usa técnicas de síntesis sustractivas ha pasado a un segundo plano, abriendo camino al mundo del software que permite crear nuevos matices sonoros e infinidad de combinaciones de sonido al alcance de cualquier usuario amateur.

2.1 Primeros pasos hacia el sintetizador

Para remontarnos al primer antecedente histórico de generación de sonido por medios electrónicos debemos viajar al año 1897, cuando Thaddeus Cahill diseñó y construyó el primer instrumento musical electrónico, el Telharmonium que podemos ver en la figura 2.1.1. El Telharmonium generaba un tono puro para cada frecuencia gracias a cientos de alternadores que giraban mediante motores. Sumando dichos tonos, conseguíamos el sonido final esperado. Por contrapartida, eso implicaba que este generador de sonidos fuera de gran envergadura. Por otro lado, el ruido generado



Figura 2.1.1 Telharmonium

por todos los motores era tal, que el Telharmonium requería que dichos motores se mantuviesen en una sala separada en un edificio aparte. El Telharmonium fue el primer caso de una retransmisión musical, pero pesaba aproximadamente 200 toneladas y para transportarse necesitó 30 vagones de un ferrocarril. Por este motivo, y por su inusual característica electrónica, avanzada para la época, el instrumento murió en 1914.

Alrededor de 1935, el ingeniero Laurens Hammond produjo el primer órgano eléctrico, basándose en parte en los discos giratorios del Telharmonium. El órgano Hammond tuvo mucho éxito, siendo el primer instrumento musical electrónico que se produjo en masa hasta tal punto que hoy en día es una marca de referencia.

2.2 El primer sintetizador



Figura 2.2.1 Moog Modular

El primer sintetizador llamado Moog Modular nació cerca del año 1965 gracias a un encuentro casual entre su creador, Robert Moog, y su profesor de música HerbDeutsh. Moog decidió desmontar los amplificadores de su guitarra para aplicar los osciladores a posibles aplicaciones musicales. Después de mejorar su experimento, nació el revolucionario Moog Modular, que vemos en la Figura 2.2.1 junto a su creador HerbDeutsh Mood, que empleaba el control por tensión para variar el tono del oscilador o el volumen del amplificador por ejemplo. Esto permitía obtener muchas posibilidades musicales con pocos elementos.

Después de varios años perfeccionando esta gran revolución de la música, Robert Moog consiguió crear el Mini-Moog ilustrado en la Figura 2.2.2, una versión mucho más manejable y más asequible a nivel económico, que permitió a muchos músicos poder hacer uso de este nuevo instrumento. Prácticamente todos los sintetizadores controlados por tensión siguen basados en gran parte a este Mini-Moog.



Figura 2.2.2 Mini-Moog

2.3 VCV Rack

Desde el primer Mini-Moog hasta ahora, la síntesis modular no ha dejado de reinventarse y perfeccionarse. En la actualidad, existen infinidad de marcas y módulos que permiten a los usuarios crear todo tipo de sonidos para múltiples aplicaciones (musicales, videojuegos etc).

Con la revolución tecnológica en la que estamos inmersos, la síntesis modular ha pasado también al mundo software. Más concretamente el software que hemos utilizado en este proyecto, VCV Rack. VCV Rack es un software libre de código abierto, multiplataforma para la síntesis modular.

Este programa tiene dos vertientes muy interesantes. La primera es la aplicación en sí, que te permite jugar con los módulos creando infinidad de patches diferentes con todo tipo de sonidos. Tiene muchos módulos distintos, tanto de marcas famosas como Befaco o Murable Instruments, como de usuarios libres que han decidido desarrollar sus propios módulos y ofrecerlos al resto de usuarios. En la Figura 2.3.1 mostramos un ejemplo de patch complejo del programa VCV Rack.

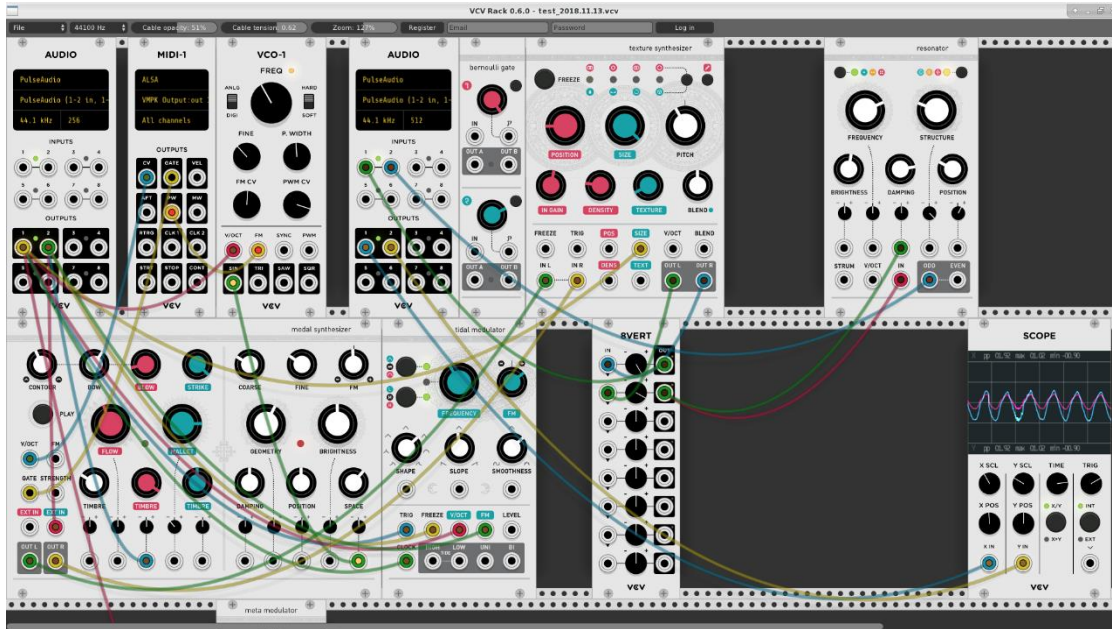


Figura 2.3.1 Ejemplo de patch en el software VCV Rack

Por otro lado, el software tiene una vertiente compilable que permite programar módulos en c++. Esto beneficia a todos los usuarios puesto que cualquier persona puede crear nuevos módulos nuevos y de esta manera generar infinidad de ellos. Gracias a esta vertiente tan potente de VCV Rack, he podido experimentar mi propia creación de módulos.

Para ello necesitamos primero configurar el entorno de desarrollo para mac, en mi caso. El primer paso es instalar las dependencias de compilación proporcionadas por el administrador de paquetes de su sistema gracias a Homebrew. A partir de aquí creamos nuestra carpeta de rack con su makefile y podemos ir programando y probando nuestro módulo en la versión de VCV Rack compilable.

También nos dan la opción de añadir a nuestra aplicación “oficial” el módulo creado sin necesidad de pasar por la librería oficial.

3. Metodología y desarrollo:

Esta tesis gira en torno a dos grandes pilares aprendidos en este grado. El primero es el procesado de señal digital, teoría fundamental para entender la base de los módulos y sobre todo para poder desarrollar el mío propio. En segundo lugar y el otro gran pilar es la programación, que en este caso se efectúa en el lenguaje c++. En este apartado vamos a desarrollar como hemos procedido para crear los módulos fusionando estos dos sujetos.

3.0 Contexto

Como ya hemos mencionado anteriormente, este proyecto pretende explorar el procesado de señales por pares en coordenadas polares. Es un campo en el cual no se ha profundizado ya que en la síntesis modular que conocemos, procesamos las señales simples para transformarlas en sonidos complejos que nos interesen. En la Figura 3.0.1 ilustramos el tipo de sistema que pretendemos realizar cogiendo como ejemplo el filtrado.

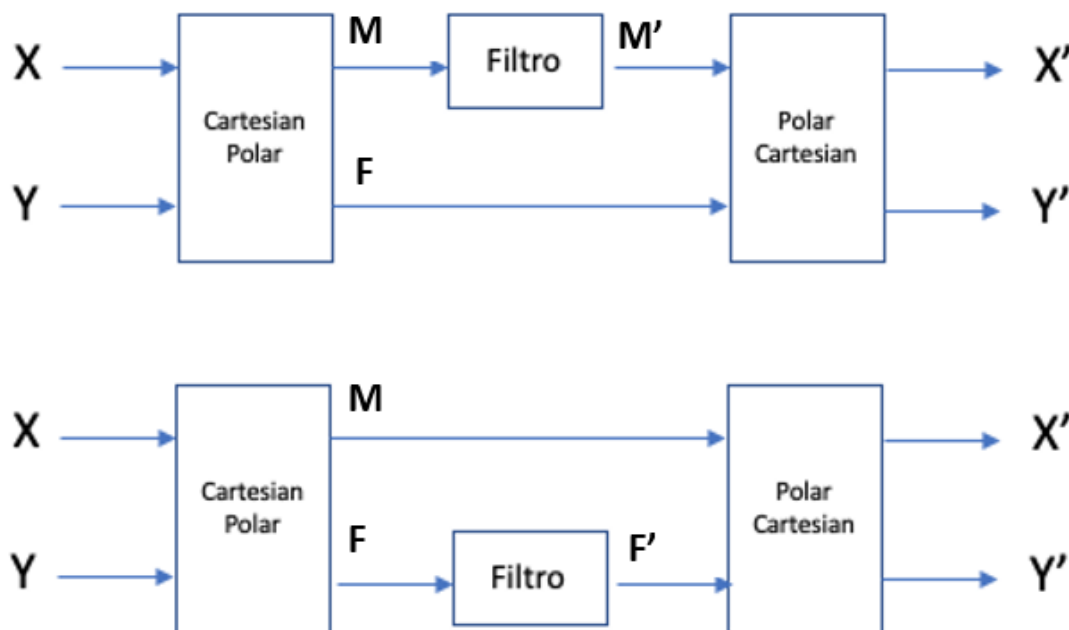


Figura 3.0.1 Esquema del sistema a estudiar cogiendo como ejemplo como el filtrado

Las señales X e Y serán a partir de ahora nuestras señales de entrada al sistema donde siempre tendremos la Y desfasada con respecto a la X, con un desfase escogido por el usuario. Cabe destacar y aclarar las distintas fases y módulos que encontraremos a lo largo de esta memoria. En la síntesis modular, todo sistema en el que tenemos entradas y salidas o que generan señales se les llama módulos. Por otro lado, también hablaremos con frecuencia de la señal módulo (M), la cual refiere al módulo de las dos señales de entrada de nuestro sistema X e Y, con la que trabajaremos constantemente. En cuanto a las fases, tenemos la señal fase (F) que la obtendremos junto a la señal

módulo gracias a las señales X e Y; y la variable fase, que es la que definirá el desfase que hay entre las dos señales de entrada.

Al pasar las señales X e Y por el módulo Cartesian Polar, obtenemos las señales M y F. Escogeremos una de estas dos últimas señales para procesarla bien sea escalándola, filtrándola o distorsionándola. Si hemos alterado la señal M, llamaremos a nuestra nueva señal M' y de la misma manera con F que llamaremos F'. Una vez procesado volveremos a pasar a mundo cartesiano para poder comparar la entrada con la salida. En la salida del sistema tendremos pues las señales X' e Y', reconstrucciones alteradas de las señales originales X e Y.

Para poder implementar este sistema, la primera cosa que necesitamos es un módulo que genere señales y señales con desfase, es decir un oscilador. En el mundo VCV Rack podemos encontrar infinidad de osciladores, pero ninguno con la opción de generar señales con desfase y es por ese motivo que el primer paso del trabajo ha sido desarrollar este módulo concreto.

Otro tema a tener en cuenta es el tipo de imágenes de los resultados que vamos a mostrar. En VCV Rack, encontraremos dos módulos esenciales para ilustrar los resultados.

El primero es un osciloscopio que puede mostrarnos las señales de dos maneras diferentes. En primer lugar y como estamos más acostumbrados, podemos ver las señales en 2D en función del tiempo. A este tipo de ilustraciones las llamaremos “temporales”. En segundo lugar, tenemos la opción de ver nuestras señales dibujadas en forma de vectores teniendo como ejes las señales X e Y. A este segundo tipo de ilustraciones las llamaremos “vectoriales”. En las figuras 3.0.2 y 3.0.3 mostramos estos dos casos respectivamente cogiendo como señales de entrada X una señal de forma de onda triangular e Y la misma señal desfasada. En este caso escogemos un desfase de 90°.

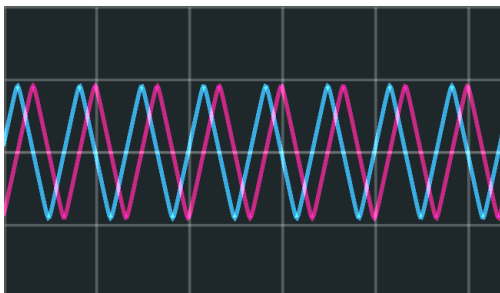


Figura 3.0.2 Señales X e Y mostradas en el display del osciloscopio en formato temporal



Figura 3.0.3 Señales X e Y mostradas en el display del osciloscopio en formato vectorial

En segundo lugar, tenemos el analizador de espectros llamado ANALYZER de la marca Bogaudio, que nos muestra el espectro frecuencial de hasta cuatro señales de entrada. En la figura 3.0.4 mostramos el espectro de las señales X del párrafo anterior.

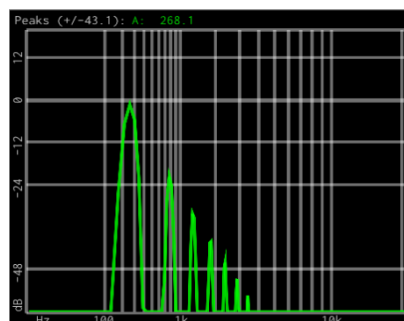


Figura 3.0.4 Espectro de la señal X

3.1 Oscilador CLAMAR

Un sintetizador modular se compone de varios módulos principales. Por un lado, tenemos módulos que generan sonidos, los osciladores, y por otro lado tenemos los módulos que alteran este sonido de muchas maneras distintas. Cabe destacar que podemos encontrar infinidad de módulos con funciones muy distintas, pero en este caso nos centraremos en estos dos tipos principales comentados anteriormente.

Como hemos expuesto anteriormente, es un trabajo experimental puesto que aún no se ha explorado la manipulación de sonidos por pares, en coordenadas polares. Por ese motivo hemos tenido que crear este oscilador CLAMAR que tiene la peculiaridad de poder generar salidas desfasadas respecto a las señales originales, oscilador que por ahora no podemos encontrar en el programa.

3.1.1 Generar señales

En un oscilador, la tarea principal es definir las salidas, que serán las señales con la forma de onda que queremos generar. Existen varias maneras de hacerlo según la utilidad que buscamos. En una primera versión experimental del oscilador CLAMAR utilizamos las fórmulas trigonométricas para definir las salidas.

Seguidamente y puesto que íbamos a manipular y filtrar las señales, tuvimos que redefinirlas de manera que tuvieran cierto control de aliasing para que no surgieran problemas adicionales. Al definir las a través de las fórmulas trigonométricas, estamos creando señales con un número infinito de armónicos que nos generarán mucho aliasing que no nos interesa.

Para ello nos apoyamos enormemente en el libro: *Developing Virtual Synthesizer with VCV Rack de Leonardo Gabrielli [1]* que explica dos técnicas posibles para tener este control sobre el aliasing. La primera lo hace a través del sobremuestreo y la segunda, definiendo las señales con funciones parabólicas.

El primer paso es redefinir las señales. Gracias al pitch y a una frecuencia definimos un incremento que se irá sumando al valor anterior hasta llegar a 1, nuestro threshold, y a continuación forzamos la caída al 0. Así es como generamos la señal diente de sierra que sirve como referencia para generar las señales de forma de onda triangular y cuadrada. En la Figura 3.1.1 ilustramos esto último que hemos explicado. Por contrapartida, la señal coseno sí la definimos con su fórmula trigonométrica ya que, al ser un tono puro, no tendrá problemas de aliasing.

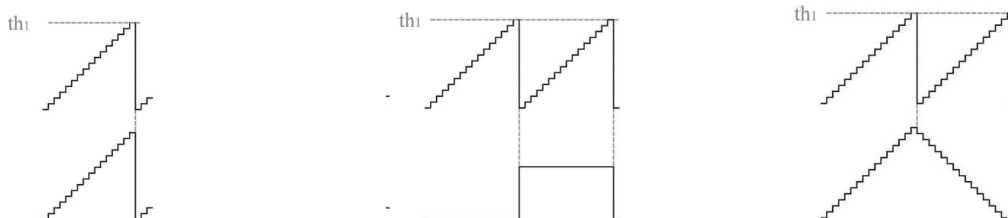


Figura 3.1.1 Esquemática del método para generar las formas de onda del oscilador CLAMAR. Vemos a la izquierda la señal diente de sierra, en el centro la señal cuadrada y a la derecha la señal triangular pero en los tres casos cogiendo siempre como referencia la señal diente de sierra.

Para continuar, debemos proceder a sobremuestrear estas señales. Este proceso consiste en aumentar la frecuencia de muestreo en un factor N, respecto a la que ya tenemos en el sistema y después diezmamos la señal para volver a la frecuencia de muestreo inicial. La frecuencia de muestreo del sistema es de 44,1 kHz y nosotros la multiplicaremos por N y de esta manera, conseguimos cumplir el criterio de Nyquist que nos asegura la reconstrucción de la señal y un espectro sin aliasing.

$$\rightarrow F_s > 2 F_{max}$$

siendo F_s la frecuencia de muestreo (sobremuestreada) y F_{max} la frecuencia máxima.

Después tenemos que diezmar para poder recuperar la señal original. En este caso, la función *decimator* lo hace por nosotros. En esta función pasara la señal por un filtro paso bajo para eliminar las frecuencias mayores a $F_s/2$ y poder diezmarla en factor N.

Gracias al estudio de Leonardo Gabrielli[1] del cual mostramos una tabla a continuación (Tabla 3.1.1), escogemos $N = 8$ ya que da una buena respuesta en cuanto al aliasing y tiene un peso computacional soportable.

	1x	2x	4x	8x
Only output connected, knob at C4	0.09 μs	0.11 μs	0.14 μs	0.25 μs
All ports connected OR knob not at C4	0.24 μs	0.26 μs	0.3 μs	0.4 μs

Tabla 3.1.2 Tabla del libro *Developing Virtual Synthesizer with VCV Rack* de Leonardo Gabrielli que muestra la media del tiempo de ejecución de la señal diente de sierra con y sin sobremuestro

Finalmente obtenemos las siguientes salidas que ilustramos en las Figuras 3.1.5, 3.1.6, 3.1.7 y 3.1.8

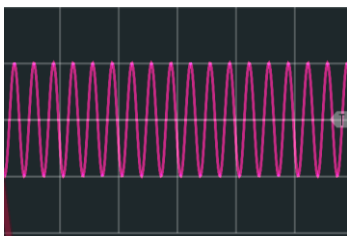


Figura 3.1.3 Señal Coseno en temporal

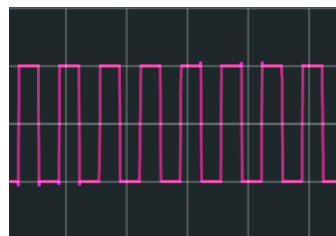


Figura 3.1.4 Señal Cuadrada en temporal

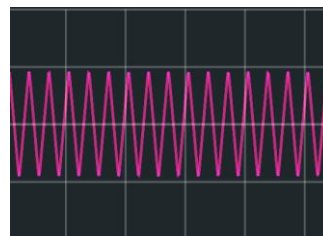


Figura 3.1.5 Señal Triangular en temporal

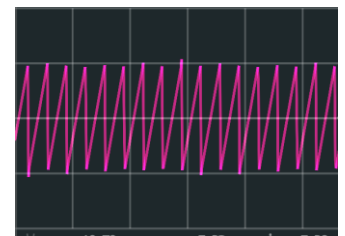


Figura 3.1.6 Señal Diente de Sierra en temporal

3.1.2 Desfase

En este párrafo llegamos al punto clave de nuestro proyecto. Este trabajo se basa en trabajar y procesar el módulo (M) y la fase (F) de señales por pares, en lugar de trabajar con señales simples. Es por eso por lo que, para cada señal generada explicada en el apartado anterior, crearemos su señal con un desfase que introducirá el usuario. Esta variable desfase que abarca una fase entera, se iniciará en 0.25, que corresponde a 90° .

Después de crear estos desfases, conseguimos las salidas que mostramos en temporal en las Figuras 3.2.1, 3.2.2, 3.2.3, 3.2.4 a continuación.

También ilustramos en las figuras 3.2.5, 3.2.6, 3.2.7 y 3.2.8 la forma que tienen cada una de las señales por pares en formato vectorial.

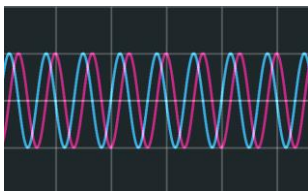


Figura 3.1.7 Señal Coseno y su señal desfasada en temporal

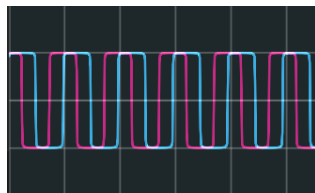


Figura 3.1.8 Señal Cuadrada y su señal desfasada en temporal

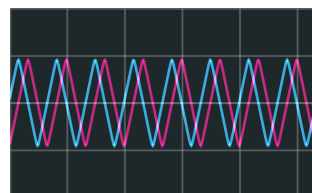


Figura 3.1.9 Señal Triangular y su señal desfasada en temporal

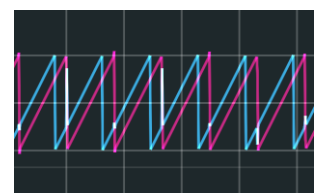


Figura 3.1.10 Señal Diente de Sierra y su señal desfasada en temporal

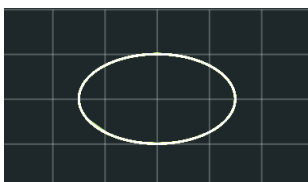


Figura 3.1.11 Señal Coseno y su señal desfasada en vectorial

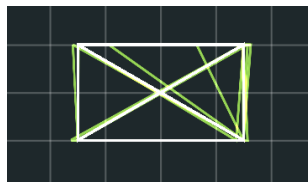


Figura 3.1.12 Señal Cuadrada y su señal desfasada en vectorial



Figura 3.1.13 Señal Triangular y su señal desfasada en vectorial

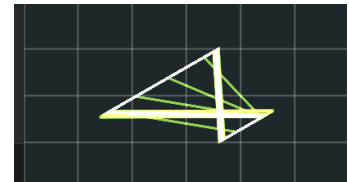


Figura 3.1.14 Señal Diente de Sierra y su señal desfasada en vectorial

3.2 Módulos conversores de coordenadas

Con tal de estudiar el efecto acústico que puede tener el hecho de procesar las señales por pares en coordenadas polares, no solo necesitamos diseñar un módulo que genere señales con un desfase específico si no también crear módulos conversores que nos permitan trabajar con el módulo (M) y la fase (F) de estas señales. Para ello generamos los módulos conversores de coordenadas cartesianas a polares y viceversa.

3.2.1 Conversor cartesianas a polares

Este es un módulo muy sencillo donde cogemos dos señales de entrada x e y , aplicamos las fórmulas matemáticas de conversión:

$$\rightarrow mod = \sqrt{x^2 + y^2}$$

$$\rightarrow fase = arctg\left(\frac{y}{x}\right)$$

En el caso de c++, utilizamos la función `atan2` puesto que esta función tiene en cuenta los signos de las señales x e y . De otra manera, a la hora de pasar de polares a cartesianas, la reconstrucción no es coherente.

3.2.2 Conversor polares a cartesianas

Para este segundo módulo, solo tenemos que coger el módulo y la fase anteriores como entrada, y generar las señales de salida x e y gracias a las fórmulas:

$$\rightarrow x = mod * \cos(fase)$$

$$\rightarrow y = mod * \sin(fase)$$

4. Resultados

Todos los sonidos del universo se componen fundamentalmente de cuatro parámetros. Tono, Timbre, Duración y Volumen. En la síntesis sustractiva, cualquier sonido nuevo viene de alterar alguna de las tres últimas características. En el caso del sintetizador modular tenemos un equivalente a cada uno de estos parámetros que ilustramos en la Figura 4.0.1. El oscilador (VCO) es el encargado de generar el tono. Para nuestro proyecto, hemos desarrollado un oscilador con la peculiaridad de tener 4 señales de salidas básicas y una señal desfasada para cada una de ellas que puede modular su desfase con el “knob”. El filtro (VCF) es uno de los módulos que se encarga del timbre y en este caso que ilustramos el amplificador (VCA) se encarga de modificar la envolvente que altera el volumen y la amplitud.

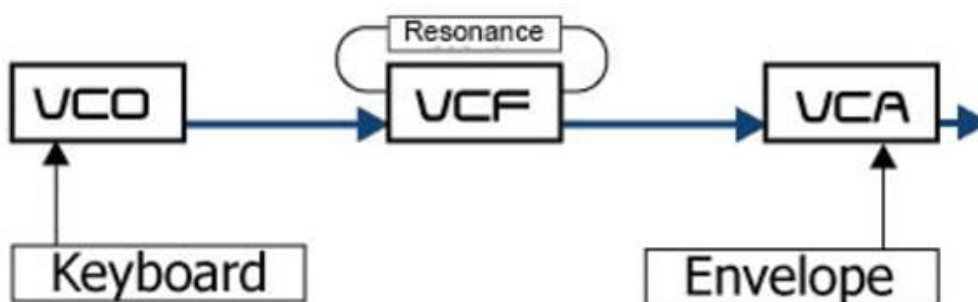


Figura 4.0.1 Diagrama de bloques que ilustra como se modifican las diferentes características de un sonido

Desde un principio, el objetivo ha sido explorar las opciones y posibilidades auditivas que tiene el hecho de trabajar con dos señales en vez de una, transformándolas en módulo (M) y fase (F) y manipulándolas en estas coordenadas polares antes de volver a transformarlo en coordenadas cartesianas.

Para ilustrar los resultados, hemos estudiado tres casos concretos. Cada uno de ellos corresponde a un tipo de manipulación y para cada caso hemos buscado la opción más interesante puesto que siempre tienes infinidad de opciones a manipular y no todas tienen el mismo interés auditivo.

Dicho interés auditivo es de criterio totalmente subjetivo.

Nota a tener en cuenta: A lo largo de este apartado vamos a ilustrar algunos ejemplos con audios muy sencillos, adjuntos en los archivos anexos. Dichos audios son únicamente orientativos puesto que la calidad del sistema de grabación no permite apreciar con claridad los matices de los cuales hablamos.

4.1 Caso 1: Escalamos la fase de un coseno y su señal desfasada

Para este primer caso, hemos escogido el tipo de manipulación más sencillo dentro del mundo del procesado, el escalado. Con este caso, pretendemos ilustrar que con un sistema tan sencillo, ya apreciamos la creación de nuevos sonidos y vemos el interés que puede tener el hecho de tratar las señales por pares en coordenadas polares.

En primer lugar, analizamos auditivamente qué resultado tenemos al escalar el módulo y la fase para escoger uno de ellos y en caso de ser ambas opciones interesantes, analizarlas las dos. Para el caso del módulo, como observamos en la Figura 4.1.1, la salida es siempre una senoide con la peculiaridad que escalando el módulo cambiamos la amplitud. Es por este motivo que la señal de salida no es interesante a nivel auditivo. Tendremos una senoide a la entrada y otra a la salida, no estamos generando nada diferente y nos da muy poca variación de sonido.

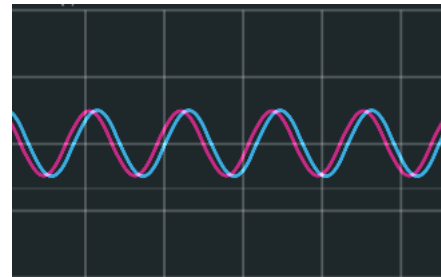


Figura 4.1.1 Señales reconstruidas a la salida del sistema escalando el módulo.

Nos centramos pues en escalar la fase.

Gracias al siguiente esquema de la Figura 4.1.2 ilustramos el sistema que estamos estudiando en este primer caso.

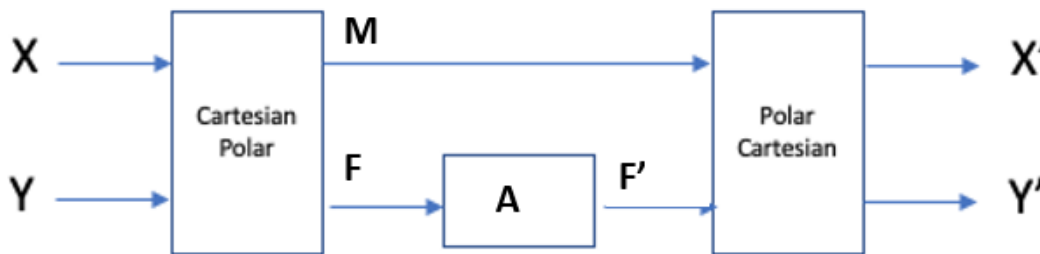


Figura 4.1.2 Esquema básico del caso 1

A continuación, ilustramos paso a paso los procedimientos. Mostramos en la Figura 4.1.3 nuestras señales de entrada en temporal donde en azul tenemos la señal X que es un coseno y en rojo la señal Y con un desfase de 90°. En la Figura 4.1.4 encontramos la forma de onda del módulo (M) en azul y de la fase (F) en rojo de estas dos señales X e Y. También observamos en las Figura 4.1.5 la entrada del sistema en formato vectorial y en la Figura 4.1.6 la transformación que le hemos hecho a la fase.

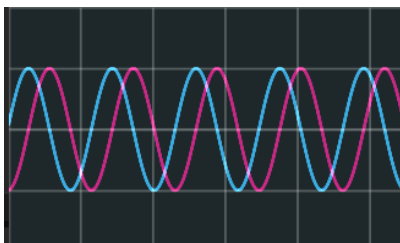


Figura 4.1.3 Señales de entrada X e Y con desfase de 90°

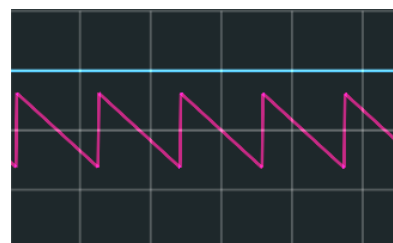


Figura 4.1.4 Módulo (azul) y fase (rojo) de las señales de entrada

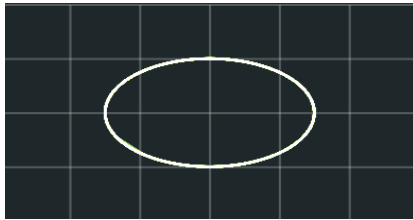


Figura 4.1.5 Señales de entrada X e Y en vectorial

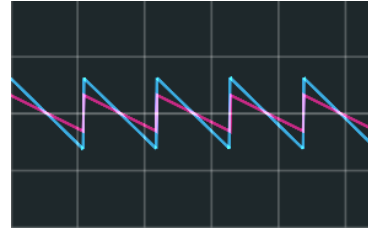


Figura 4.1.6 Fase original F (azul) y fase escalada F' (rojo)

En primer lugar, observamos la forma de onda de la fase cuando tenemos el desfase inicial, 90° . Es una señal diente de sierra que va de $-\pi$ a π . Escalamos la fase para que tenga un rango dinámico aproximadamente de la mitad y observamos muchos cambios tanto en la forma de onda de la salida, X' e Y' como a nivel auditivo.

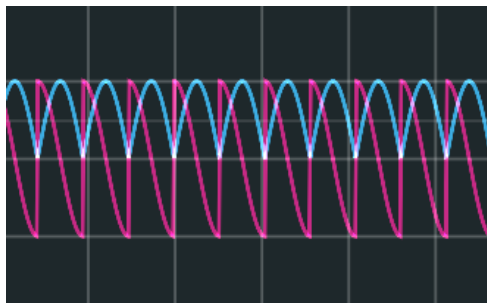


Figura 4.1.7 Señales reconstruidas. En azul tenemos la señal X' y en rojo la señal Y'

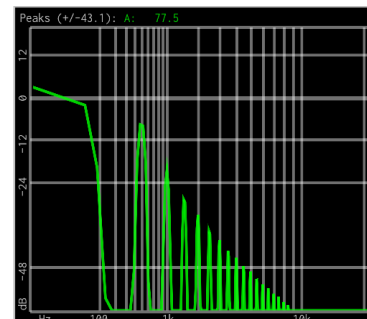


Figura 4.1.8 Espectro frecuencial la señal X' .

En la figura 4.1.7 vemos las señales X' e Y' a la salida del sistema. Podemos ver que hemos transformado considerablemente la forma de onda. Tenemos una señal reconstruida X' que es únicamente positiva, y por otro lado tenemos una señal Y' , con un rango dinámico parecido a la señal de entrada, pero con saltos. Por otro lado, hemos pasado de tener un tono puro, a una señal frecuencial rica en armónicos que ilustramos en la Figura 4.1.8. Antes incluso de escuchar cómo suena ya podríamos haber deducido que hemos transformado notoriamente el sonido.

Para entender este resultado miramos que está pasando a nivel teórico. Estas formas de onda son debido a la transformación de coordenadas.

Para recuperar la señal X utilizamos la fórmula $X' = M * \cos(F')$. Al tener una fase de entrada que va de $-\frac{\pi}{2}$ a $\frac{\pi}{2}$, nos encontramos con una X' que será siempre positiva ya que el módulo es siempre positivo, y el coseno también en este rango de valores.

Para el caso de la señal Y' tenemos $Y' = M * \sin(F')$ y en este caso vemos como se nos elimina la parte izquierda de cada onda que es la que corresponde a los valores de la fase fuera del rango en el que nos encontramos.

Escuchando y valorando las distintas opciones, vemos que es muy interesante estudiar este caso puesto que algo tan simple como escalar la fase nos genera sonidos muy distintos con muchas posibilidades cambiando el escalar, el desfase y el pitch.

→ En el Audio 4.1.7 podemos escuchar el tono puro en los primeros segundos y a continuación la salida de nuestro sistema y como varía cuando modificamos el escalar.

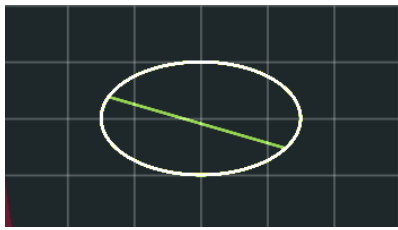


Figura 4.1.9 Señales de entrada, X e Y en formato vectorial

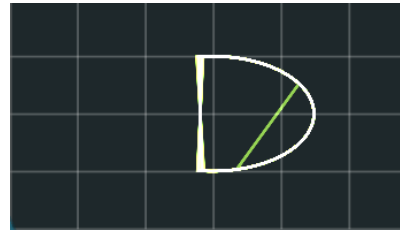


Figura 4.1.10 Señales de salida X' e Y' en formato vectorial

En estas dos imágenes, Figura 4.1.9 y 4.1.10, vemos las señales de entrada y las de salidas en formato vectorial y vemos claramente como hemos eliminado las fases de $-\frac{\pi}{2}$ a $\frac{\pi}{2}$.

Para seguir ilustrando la idea que hemos planteado, cambiamos el parámetro de desfase a 180° y dejamos el rango dinámico de la fase a la mitad. Obtenemos la siguiente salida

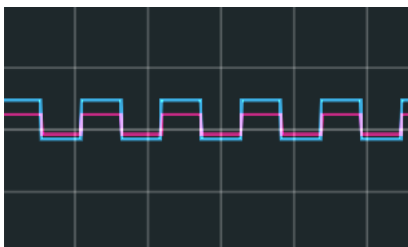


Figura 4.1.11 Señal fase F (azul) y señal fase escalada F' (rojo)

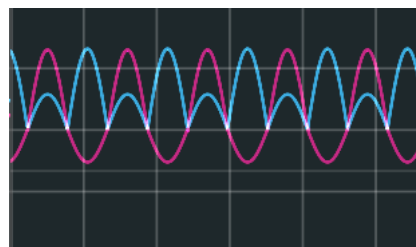


Figura 4.1.12 Señales reconstruidas a la salida. En azul tenemos la señal X' y en rojo la señal Y'.

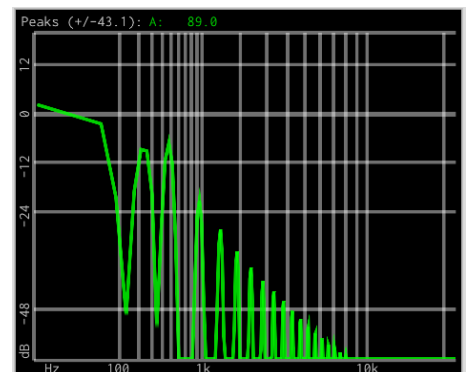


Figura 4.1.13 Espectro frecuencial de la señal X' a la salida.

→ En el Audio 4.1.12 podemos escuchar el tono puro en los primeros segundos y a continuación la salida de nuestro sistema con un desfase de 180° y como varía cuando modificamos el escalar.

Después de ver estos casos, concluimos que el rango dinámico de la fase tiene mucha influencia sobre la reconstrucción de las señales y por tanto sobre el sonido de salida. El rango dinámico de la fase afecta directamente a lo que oímos y es un dato importante a tener en cuenta a la hora de buscar nuevos sonidos. Haciendo una transformación tan simple como escalar la fase, encontramos una salida muy interesante con opciones de sonido muy variadas según las variables disponibles.

4.2 Caso 2: Filtramos la fase de una señal diente de sierra y un triángulo

Cuando pasamos una señal por un filtro buscamos eliminar cierta banda de frecuencias. A nivel auditivo, al filtrar estamos alterando el timbre de la señal y por lo tanto afecta directamente a lo que oímos.

De igual manera que en el caso anterior, el primer paso es decidir si es más interesante manipular el módulo o la fase de estas dos señales. Para ello analizamos el espectro, la forma de onda a la salida y el sonido que obtenemos tanto del módulo como de la fase. En la siguiente Figura 4.2.1 vemos el espectro frecuencial de la fase y el módulo y constatamos que la fase es más rica armónicamente y tiene un rango de frecuencias muy elevado en cambio con el módulo, da la sensación que en las altas frecuencias nos encontramos ruido.

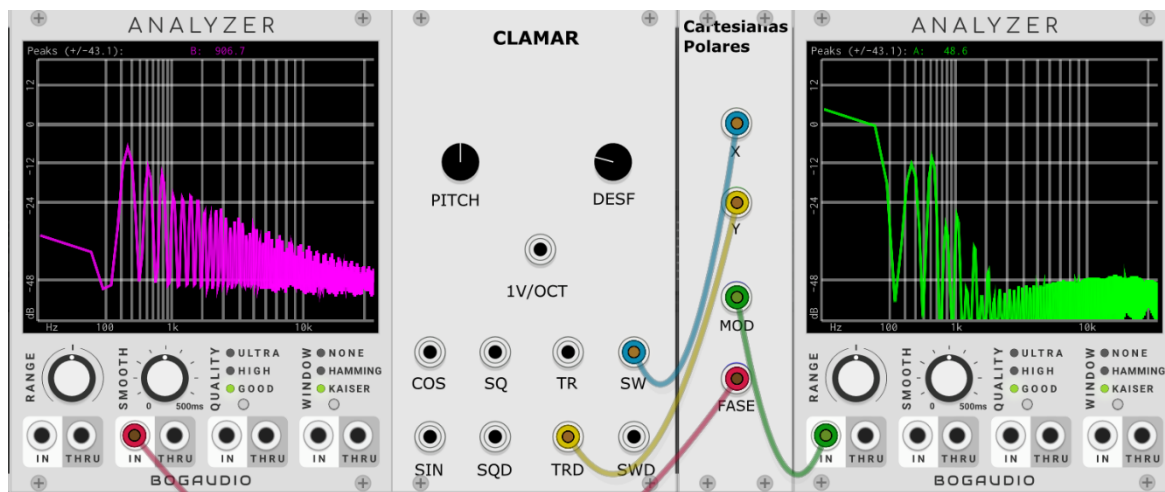


Figura 4.2.1 Espectros del módulo (en verde) y de la fase (en lila) de las señales diente de sierra y triangular desfasadas.

Cuando pasamos a valorar el sonido, con el módulo nos encontramos con una señal que auditivamente no tiene mucho interés puesto que prácticamente no varía su sonido al alterar las variables (desfase y frecuencia de corte). En el caso de utilizar un filtro paso alto, nos encontramos con dos señales de salida de media prácticamente nula, con poca variación y poco volumen. En el caso del filtro paso bajo, encontramos señales cercanas a la sinusoidal y con poca o nula influencia de la frecuencia de corte y por tanto manipular la frecuencia del módulo no tiene interés. En las Figuras 4.2.2 y 4.3.3 podemos ver la idea que acabamos de exponer.

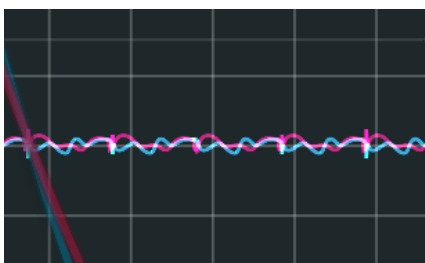


Figura 4.2.2 Salidas X' e Y' al filtrar el módulo M con un filtro paso alto.

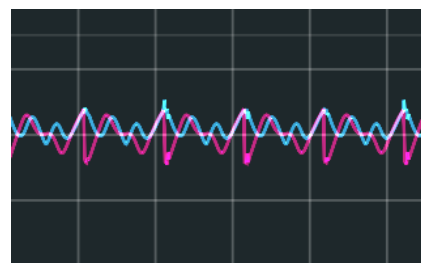


Figura 4.2.3 Salidas X' e Y' al filtrar el módulo M con un filtro paso bajo.

- En los audios 4.2.2 y 4.2.3 escuchamos las salidas correspondientes a las figuras anteriores variando las frecuencias de corte y el desfase. En ambos casos variamos la frecuencia de corte, pasando de las frecuencias de corte que eliminan todas las bandas de frecuencia hasta la frecuencia de corte que deja pasar todas las bandas de frecuencia.

Decidimos manipular la fase con un filtro paso bajo puesto que, de manera subjetiva, encuentro el sonido de salida más interesante, pero ambos dos pueden utilizarse a gusto del usuario. En la Figura 4.2.4 mostramos un esquema básico de cómo es nuestro sistema, donde X será la señal con forma de onda diente de sierra y la señal Y será la señal triangular.



Figura 4.2.4 Esquema básico del sistema del caso 2.

En las siguientes Figuras 4.2.5, 4.2.6 y 4.2.7 vemos las señales de salida X' e Y' con diferentes desfases. La primera cosa que observamos es que, en la salida, sea cual sea el desfase, la señal X' nos recuerda a la forma de onda diente de sierra que es la señal de entrada X, mientras que la señal triangular es irreconocible.

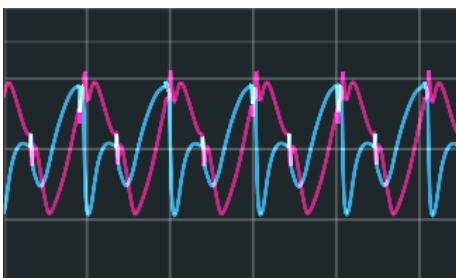


Figura 4.2.5 Señales de salida con **desfase = 0.1**. En rojo tenemos la señal X' y en azul la señal Y'

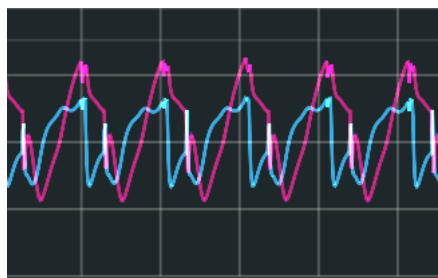


Figura 4.2.6 Señales de salida con **desfase = 0.25**. En rojo tenemos la señal X' y en azul la señal Y'

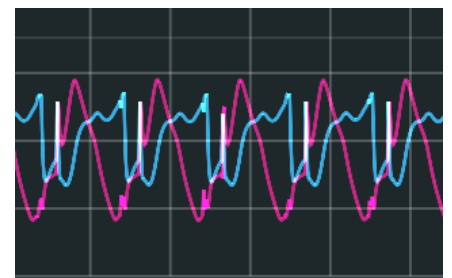


Figura 4.2.7 Señales de salida con **desfase = 0.8**. En rojo tenemos la señal X' y en azul la señal Y'

Esta observación nos lleva a preguntarnos: ¿Por qué este filtrado afecta más a la señal triangular que a la señal diente de sierra?

Para tratar de resolver esta duda, analizamos las señales originales. Sabemos que el espectro de una señal triangular tiene únicamente los armónicos de orden impar y decrecen al ritmo de $\frac{1}{n^2}$ mientras que la onda en diente de sierra tiene todos los armónicos y decrecen en $\frac{1}{n}$. Lo ilustramos a continuación gracias a la figura 4.2.8.

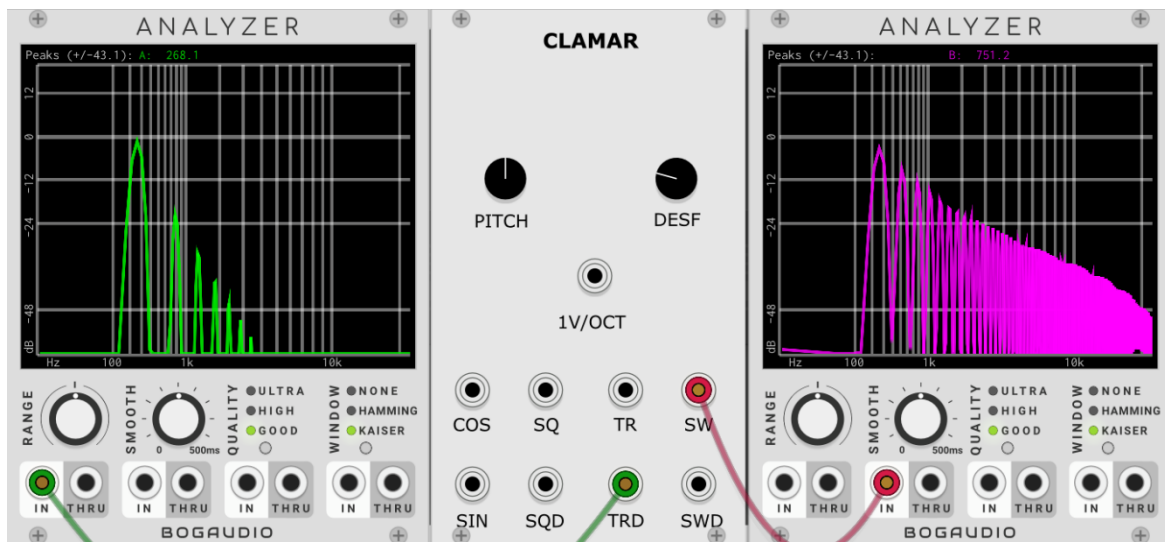


Figura 4.2.8 Espectros frecuenciales de las señales diente de sierra X en rosa y triangular Y en verde

Pero lo que pasamos por el filtro es la fase y no las señales X e Y. En las siguientes Figuras 4.2.9 y 4.2.10 vemos el espectro de la fase F y su forma de onda cuando el desfase es igual 90° .

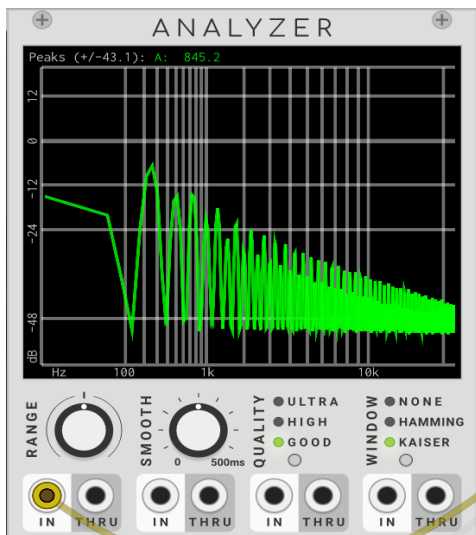


Figura 4.2.9 Espectro de la fase

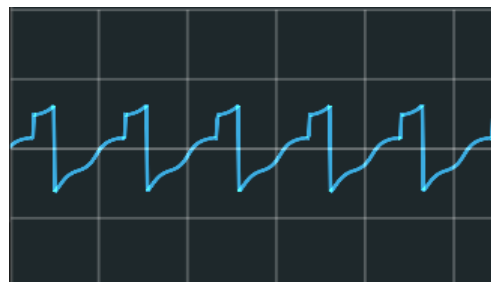


Figura 4.2.10 Forma de onda de la fase

Vemos cierta similitud del espectro de la fase con el de la diente de sierra, pero con los primeros armónicos más definidos.

Si intentamos hacer una comparativa con las señales antes de ser transformadas en coordenadas polares, nos damos cuenta de que en este caso tenemos más bien una densidad espectral que decrece a un ritmo más parecido a la señal diente de sierra y al transformarla de nuevo encontraremos una señal reconstruida X' con este ligero parecido a la entrada X . Por lo contrario el espectro se aleja del de la señal triangular generando una señal Y' completamente diferente a la entrada Y .

Aun y así, es un sistema que no da un resultado demasiado interesante a nivel acústico puesto que tanto el desfase como la frecuencia de corte apenas alteran el sonido a la salida.

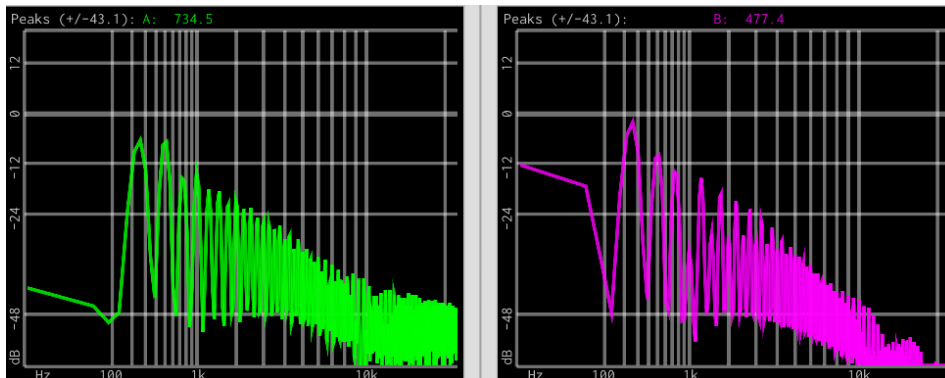


Figura 4.2.11 Espectro frecuencial de las salidas X' (verde) e Y' (rojo) con una frecuencia de corte de 1,5 kHz i **desfase de 36°**.

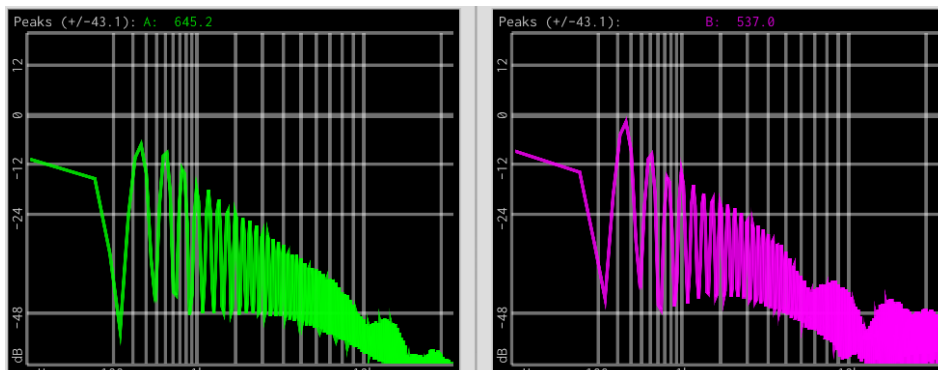


Figura 4.2.12 Espectro frecuencial de las salidas X' (verde) e Y' (rojo) con una frecuencia de corte de 1,5 kHz i **desfase de 90°**.

En las Figuras 4.2.11 y 4.2.12 vemos el espectro de las señales reconstruidas con desfases diferentes y observamos la poca variabilidad de las señales reconstruidas. Esto confirma una vez más el poco interés que tiene manipular las frecuencias en el contexto de las variables fase y modulo.

Como conclusión general y en comparación con el resto de los casos podemos deducir que manipular la frecuencia mediante el filtro, tanto del módulo como de la fase, no tiene demasiado interés acústico. Hemos estudiado el filtro paso bajo y sí que generamos sonidos nuevos con algo de juego cambiando las variables, pero sigue sin ser el foco de interés de este estudio.

4.3 Caso 3: Distorsión módulo de un triángulo y un triángulo desfasado

Para este último caso, lo que haremos es una distorsión del módulo del estilo wavefolding. Puesto que esta manipulación no resulta tan obvia de entender como las otras dos haremos una breve explicación del wavefolding.

Es un tipo de distorsión de síntesis en el que cuando la amplitud de entrada supera cierto umbral, se crea un espejo de la imagen y se refleja una porción de ella misma en donde podríamos hacer un símil visual a “pliegues”. Estos pliegues producen espectros complejos a partir de señales sencillas como la senoide, mientras que otras como la señal cuadrada no se verá afectada. Mientras en el caso 2 eliminábamos bandas de frecuencia, en este nuevo sistema estaremos añadiendo componentes frecuenciales para estudiar el caso opuesto del filtrado. En la Figura 4.3.1 ilustramos esta idea de pliegues que hemos explicado.

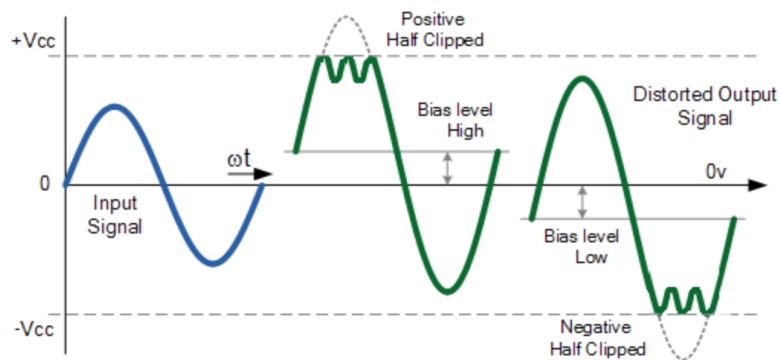


Figura 4.3.1 Ilustración del wavefolding en una senoide

En las Figuras 4.3.2 y 4.3.3 ilustramos el ejemplo de wavefolding con nuestro coseno visualizando tanto la forma de onda en formato temporal como su espectro.

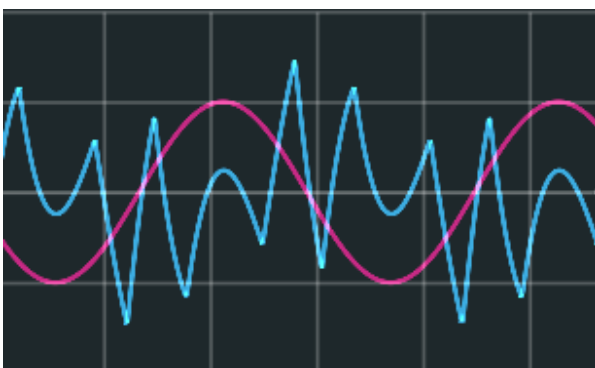


Figura 4.3.2 Coseno (rojo) y coseno distorsionado por wavefolding (azul)

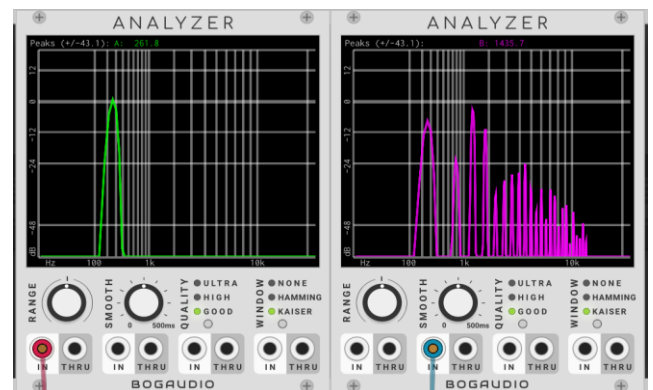


Figura 4.3.3 Espectro de un coseno (verde) y del coseno distorsionado por wavefolding (rojo)

Para hacer nuestro estudio, escogemos el Shaper de la marca Squinky Labs por ser un Wave folder con control de aliasing. Además, hace el control de aliasing con el método sobremuestreo, de la misma manera que hemos procedido para el oscilador CLAMAR.

Puesto que la distorsión wavefolding añade componentes armónicas, decidimos trabajar con una señal con pocos armónicos como es la señal triangular X y su señal desfasada Y.

Viendo las formas de onda de la fase (F) y el módulo (M) que ilustramos en la Figura 4.3.4 decidimos distorsionar el módulo (M) puesto que es una señal parecida a una senoide y, como explicado anteriormente, resulta más interesante de distorsionar.

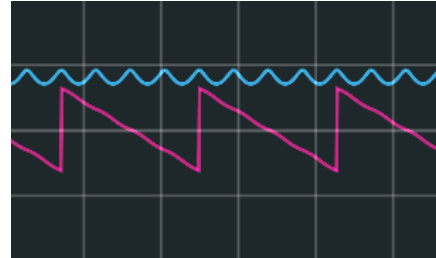


Figura 4.3.4 Modulo M (azul) y fase F (rojo) de las señales X e Y.

De la misma manera que en los casos anteriores, ilustramos en la Figura 4.3.5 con un esquema básico la manipulación de las señales triangular X y su desfasada Y que vamos a exponer.

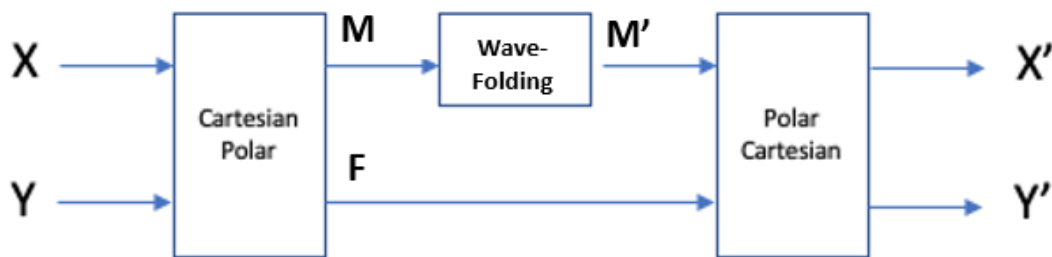


Figura 4.3.5 Esquema básico del caso 3

Antes de ver qué salida obtenemos, miramos qué implica hacer un wavefolding de una señal triangular simple. Vemos las salidas en la Figura 4.3.6 donde en la imagen de la izquierda tiene una ganancia baja y la de la derecha aumentando la ganancia a 3,15.

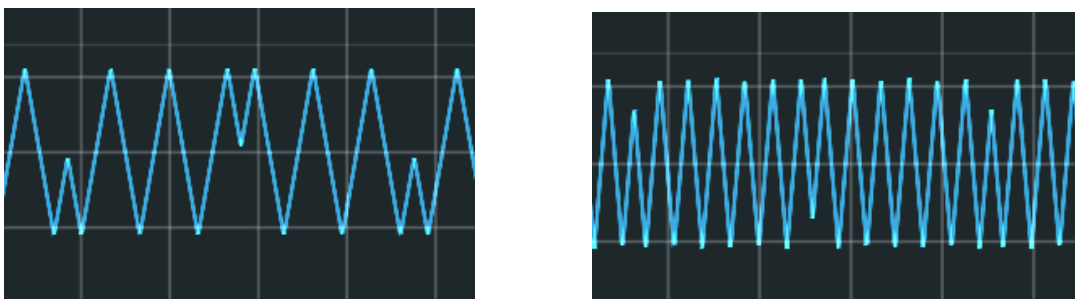


Figura 4.3.6 Formas de onda de la señal X después de ser distorsionada por wavefolding. En la imagen de la izquierda con una ganancia mínima y en la imagen de la derecha con una ganancia de 3,15.

Ahora procedemos a ejecutar el sistema descrito al principio del apartado y mostramos el resultado. Con el mínimo de distorsión y el desfase de 90°, ya podemos apreciar como aparecen numerosas componentes frecuenciales que observamos en la Figura 4.3.9 y cómo cambia considerablemente la forma de onda de las señales X' e Y' a la salida. En la Figura 4.3.8 vemos las señales X' e Y' y podemos destacar el hecho que, obtenemos una forma de onda parecida a la que esperabamos al pasar la señal X simple por el Shapper Wavefolding.

En la Figura 4.3.7 mostramos la forma de onda del módulo M' distorsionado junto con la señal M.

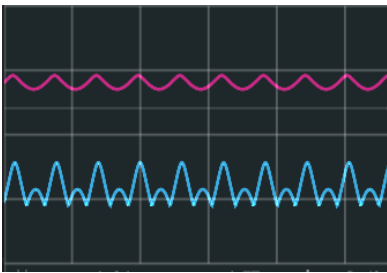


Figura 4.3.7 Señal del módulo M (rojo) y la señal M' (azul)

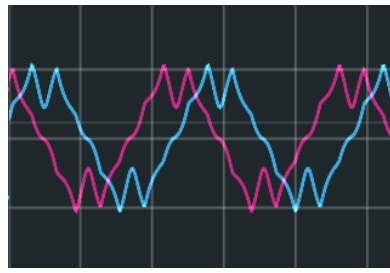


Figura 4.3.8 Señales a la salida X' (azul) e Y' (rojo)

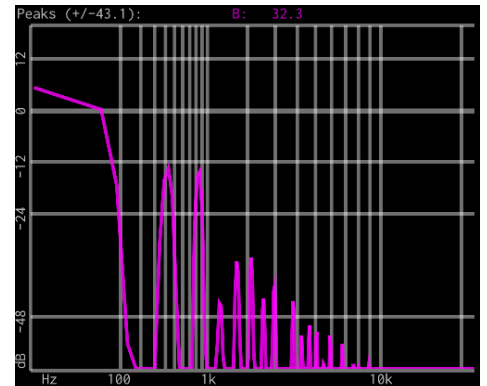


Figura 4.3.9 Espectro frecuencial de la señal a la salida X'

- ➔ Audio 4.3.8 Escucharemos en unos primeros segundos el triángulo simple pasado por un wavefolind con la ganancia mínima, y posteriormente la salida de nuestro sistema (Figura 4.3.8) con ganancia mínima, pero variando el desfase.

Para seguir ilustrando las peculiaridades de este sistema, en la Figura 4.3.10 vemos las señales de entrada X e Y en formato vectorial (izquierda) y las señales a la salida X' e Y' también en formato vectorial (derecha). Igual que en formato temporal podemos seguir diferenciando los triángulos de las señales, en este caso vemos como la forma cambia totalmente.

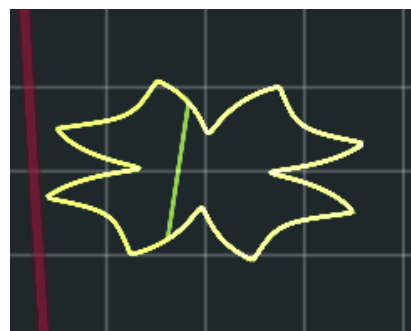
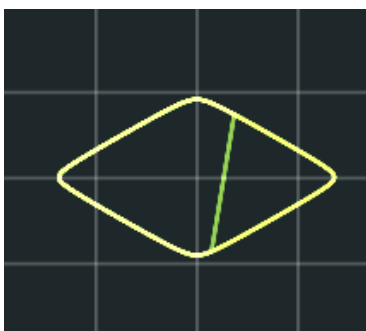


Figura 4.3.10 Señales X e Y en formato vectorial a la izquierda y señales de salida X' e Y' en formato vectorial a la derecha.

¿Qué resultados obtenemos si buscamos aún más distorsión y aumentamos la ganancia 3,15? A continuación, ilustramos los resultados con las Figuras 4.3.11, 4.3.12, 4.3.13 y 4.3.14 donde veremos el módulo antes (M) y después (M') del wavefolding, las señales X' e Y' a la salida del

sistema tanto en formato temporal como en formato vectorial y por último el espectro de la señal reconstruida X' , respectivamente.

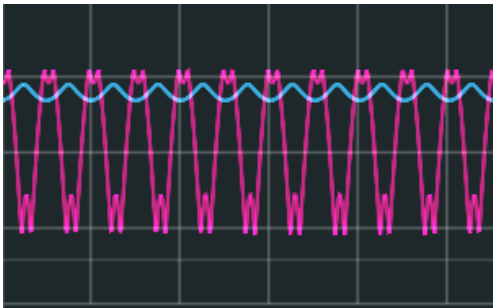


Figura 4.3.11 Modulo M (azul) y modulo M' (rojo)

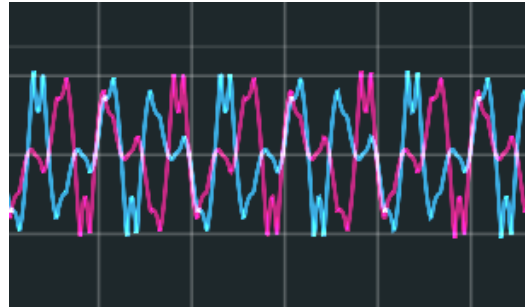


Figura 4.3.12 Señales a la salida X' (azul) e Y' (rojo) en formato temporal

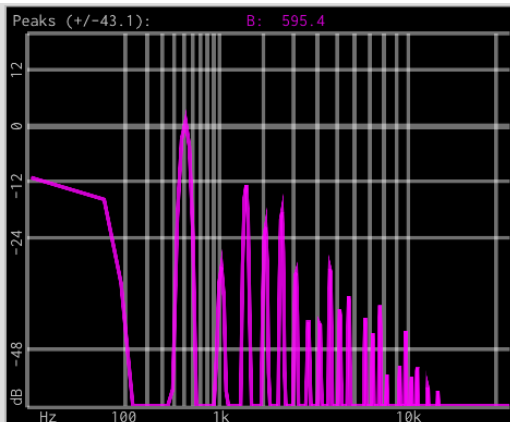


Figura 4.3.14 Espectro de la señal X' a la salida

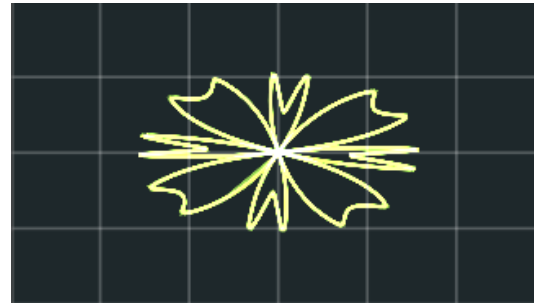


Figura 4.3.13 Señales a la salida X' e Y' en formato vectorial

- ➔ Audio 4.3.12 Escucharemos el triángulo simple pasado por el wavefolding con una ganancia de 3.15 en los primeros segundos y a continuación la salida de nuestro sistema con esa misma ganancia pero variando el desfase.

En base a estos resultados vemos que la densidad espectral va aumentando de amplitud y van apareciendo más armónicos. En cuanto al formato vectorial, van apareciendo “pétalos de una flor”, pero siempre nos encontramos un dibujo nítido en cuanto a las líneas y a medida que aumentamos la ganancia aparecen pétalos, pero nunca deja de ser “una flor”.

Como conclusión de analizar la distorsión en el módulo, podemos decir que resulta muy interesante a nivel auditivo. Para cada valor de la ganancia obtenemos un sonido diferente con un espectro muy rico en armónicos que a nivel subjetivo me recuerda a algún instrumento africano. Podemos decir con seguridad que de los tres casos estudiados este último es el que resulta más interesante con sonidos muy peculiares que es difícil conseguir con otros patches. Es un sistema que no deja de ser sencillo puesto que usamos pocos módulos y que da un resultado con muchas posibilidades de sonidos diferentes.

5. Budget

Este proyecto tiene la ventaja de poderse trabajar en su totalidad de forma gratuita. El software VCV Rack es gratuito y tiene algunos módulos de pago opcionales, pero no son estrictamente necesarios para tal. Por lo tanto el coste solamente recae en las horas dedicadas por mí, Junior Engineer. El cálculo de horas aproximado es de: 15h/semana durante 22 semanas = 330 horas a 9€ la hora = 2970€.

6. Conclusiones y desarrollo futuro:

Para concluir este estudio podemos decir que estamos muy satisfechos con el trabajo. Desde un principio sabíamos que era un trabajo experimental en donde cabía la posibilidad que el resultado no fuera interesante. Falta profundizar un poco más, pero en estos primeros pasos hemos encontrado sonidos nuevos que parecen interesantes.

En la defensa mostraremos un patch completo que pueda ilustrar la creación de sonidos puesto que a través de los archivos de audio no puede apreciarse correctamente.

Después de ver los tres casos concretos, vemos que acústicamente si tiene un interés partir de señales por pares, transformarlas a coordenadas polares y manipular las variables módulo y fase, pero no todas las manipulaciones dan buenos resultados.

En un principio, solo planteamos filtrar y distorsionar el módulo y la fase, pero haciendo pruebas vimos el interés que tiene escalar la fase, algo muy sencillo y que sin embargo ha dado unos resultados muy satisfactorios.

Por lo contrario, nos ha sorprendido el segundo caso, donde filtrar la frecuencia tanto del módulo como de la fase es menos interesante de lo que pensábamos. Aún y así, filtrar la fase, tanto en paso alto como paso bajo si da cierto juego a nivel acústico creando sonidos electrónicos, pero subjetivamente es una manipulación que no aporta demasiado a ningún patch.

Por último, el wavefolding es la manipulación más interesante de las tres a nivel subjetivo. Hemos hecho un tipo de manipulación al módulo mucho menos evidente que en los demás casos. No hemos entrado en lo que hace a nivel teórico puesto que es complejo y no nos concierne, pero nos ha dado muy buenos resultados a nivel acústico, creando sonidos muy interesantes y con infinidad de posibilidades con la misma configuración.

Puesto que este es un trabajo experimental, para un futuro desarrollo podríamos barajar más combinaciones y manipulación y perfeccionar los módulos para ofrecerlos en la librería de VCV Rack y mejorar su interfaz gráfica.

Una pregunta abierta para estudiar sería ver si hay manera de conseguir estas formas de onda o estos sonidos a través de otros patches, sin necesidad de trabajar con las coordenadas polares.

Bibliography:

A thorough reference list as in the following examples: Conference paper [1], journal paper [2], book [3], standard-1 [4], standard-2 [5], online reference [6], patent [7], M.S. thesis [8] and Ph.D. dissertation [9].

[1] L. Gabrielli . *Developping virtual synthesisers with VCV Rack*, 1st ed. New York, USA: Routledge, 2020.

[2] J.Aikin *How to Rack*, 1st ed July 1019.

[3] G.Roid, *Sound on sound, Synth secrets*, from may 1999 to July 2004.

[4]A.Belt “Plugging Development Tutorial” Tennessee, USA 2016 [online] Available: <https://vcvrack.com/manual/PluginDevelopmentTutorial>

[5] J.Gutiérrez-Ravé Olmos. “Diseño e implementación de un sintetizador de audio modular basado en síntesis sustractiva”. M.S thesis, Escuela Técnica Superior de Ingeniería del Diseño, Universitat Politècnica de València, España 2018.

[6] G.Recio Cervantes. “Diseño e implementación de un sintetizador analógico modular”. M.S.thesis, Escuela EUIT Telecomunicación, Universidad Politécnica de Madrid. España 2013