

MESTRADO

MULTIMÉDIA - ESPECIALIZAÇÃO EM MÚSICA INTERACTIVA E DESIGN DE SOM

# **A Sintaxe Harmónica Atonal no Espaço de Fourier: Para Uma Representação Esférica das Fases**

Samuel Filipe da Silva Pereira

**M**

2020

FACULDADES PARTICIPANTES:

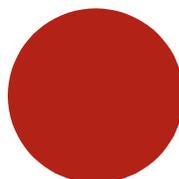
FACULDADE DE ENGENHARIA

FACULDADE DE BELAS ARTES

FACULDADE DE CIÊNCIAS

FACULDADE DE ECONOMIA

FACULDADE DE LETRAS





# **A Sintaxe Harmónica Atonal no Espaço de Fourier: Para Uma Representação Esférica das Fases**

**Samuel Filipe da Silva Pereira**

Mestrado em Multimédia da Universidade do Porto

Orientador: Gilberto Bernardes

Junho de 2020

© Samuel Pereira, 2020

# **A Sintaxe Harmónica Atonal no Espaço de Fourier: Para Uma Representação Esférica das Fases**

**Samuel Filipe da Silva Pereira**

Mestrado em Multimédia da Universidade do Porto

Aprovado em provas públicas pelo Júri:

Presidente: Maria Teresa Andrade

Vogal Externo: José António Oliveira Martins

Orientador: Gilberto Bernardes de Almeida

# Resumo

Os arquétipos da sintaxe harmónica e a função que desempenham dentro da estrutura temporal da música tonal ocidental estão amplamente sistematizados nos tratados de teoria musical. Por exemplo, a função que um acorde desempenha dentro das estruturas tonais é tipicamente reduzida a uma categorização tripla (tónica, subdominante e dominante), cujo alinhamento temporal comporta valor sintático. Na música pós-tonal do início do século XX, este fenómeno é muito menos sistematizado e compreendido, evoluindo mesmo, nos últimos anos, para a existência de quase tantas teorias e sistemas de composição como o número de peças ou compositores. Neste contexto, o presente estudo avança alguns métodos e ferramentas dentro da musicologia computacional, que visam encontrar padrões sintáticos em estruturas harmónicas atonais. Para tal, seguem-se métodos recentes de Yust (2015) sobre a utilização da informação de fase a partir da transformada discreta de Fourier para conjuntos de alturas.

O principal contributo desta dissertação, ampliando investigações anteriores, é uma representação esférica tridimensional na qual foi disposta a informação de fase de dois coeficientes da transformada discreta de Fourier para traçar o trajeto harmónico de uma peça atonal. Os mapeamentos espaciais daí resultantes identificam automaticamente grupos de conjuntos de alturas, bem como os padrões harmónicos formados a partir dos mesmos. Finalmente, para testar esta representação tridimensional, a mesma foi utilizada para analisar a obra *Sechs kleine klavierstück*, Op.19, de Arnold Schönberg, e também como ferramenta de apoio ao processo de composição, sugerindo movimentos padronizados no material harmónico apresentado pelo compositor. Em resumo, pretende-se com este estudo: (1) dispor os conjuntos harmónicos do Op. 19, de Arnold Schönberg, num espaço tridimensional de acordo com as suas classes de altura; (2) organizar esses conjuntos em diferentes agregados harmónicos; (3) identificar a existência de movimentos padronizados entre eles; (4) comparar os resultados da representação com análises anteriores e (5) avaliar a utilidade do sistema no processo criativo do compositor.

Os resultados mostram, entre outros fatores, que as trajetórias harmónicas padronizadas no espaço esférico de fases tendem, em cada repetição, a preencher a totalidade das classes de altura.

**Palavras-chave:** sintaxe harmónica, composição musical, transformada discreta de Fourier



# Abstract

Harmonic syntax archetypes and the function they play within the temporal structure of Western tonal music are vastly systematized across music theory treatises. For instance, the function a chord plays within tonal structures is typically reduced to a threefold categorization (tonic, subdominant, and dominant), whose temporal alignment implies syntactic value. In the post-tonal music of the early 20<sup>th</sup> century, this phenomenon is much less systematized and understood, evolving even, in recent years, to the existence of almost as many theories and compositional systems as the number of composers or individual works. In this context, this study advances some methods and tools within computational musicology, which aims to unpack typical syntactic patterns in atonal harmonic structures. To this end, recent methods of Yust (2015) were followed on the adoption of the phase information from the discrete Fourier transform of pitch-class sets.

The main contribution of this dissertation, extending previous research, is a three-dimensional spherical representation where the phase information of two discrete Fourier transform coefficients was displayed to plot the harmonic path of an atonal music piece. The resulting spatial mappings automatically identify pitch-class set clusters as well as the harmonic patterns across them. Finally, to test this three-dimensional representation, it was used to analyze the *Sechs kleine klavierstück*, Op.19 by Arnold Schönberg, and as a assistance tool to the composing process, suggesting movement patterns in the harmonic material presented by the composer. In summary, it is intended with this study to: (1) plot the harmonic sets of Arnold Schönberg's Op. 19 in a three-dimensional space according to its pitch-class content; (2) group such sets into different harmonic clusters; (3) identify the existence of standardized movements between these clusters; (4) compare the results of the representation with previous analysis and (5) assess the utility of the system in the composer's creative process.

The results show, among other factors, that the harmonic trajectories within the spherical phase space tend, with each repetition, to fill all pitch-classes.

**Keywords:** harmonic syntax, musical composition, discrete Fourier transform.



# Agradecimentos

A todos os colegas e professores do MM, por tanto me ensinarem nos últimos anos.

À Nádia, por falar tão bem Pythonês como Português.

Ao Gilberto, pelo indispensável conhecimento, inesgotável paciência e infinita disponibilidade.

À Elisama, por aparecer na minha vida tão inesperada quanto redentoramente.

À Ana e à Rute, por só me saberem apoiar.

Aos meus pais, a quem devo o que sou.

A Deus, por tudo!

Estão todos, por certo, no meu coração!

Samuel Pereira



*[Schönberg's] music leads us into a realm where musical experience is a matter not of the ear  
but of the soul alone – and from this point begins the music of the future.*

Wassily Kandinsky ([2009](#), p. 49)



# Índice

<b>1. Introdução.....</b>	<b>1</b>
1.1 Motivação .....	2
1.2 Problema e Objetivos da Investigação.....	3
1.3 Metodologia de Investigação .....	3
1.4 Estrutura do Texto .....	4
<b>2. Sintaxe, Teoria Atonal e Fourier.....</b>	<b>5</b>
2.1 Música e Linguagem: A Sintaxe Musical.....	5
2.1.1 Sintaxe musical e linguística: diferenças e semelhanças .....	6
2.1.2 A organização estrutural da sintaxe musical.....	9
2.2 Teoria Musical Pós-Tonal.....	13
2.2.1 Princípio da equivalência de oitava .....	14
2.2.2 Classes de altura .....	15
2.2.3 Conjuntos de alturas e a noção de classe .....	16
2.2.4 Classes intervalares e vetor intervalar .....	17
2.3 Espaços Geométricos e Fourier na Música.....	18
2.3.1 Espaços geométricos de representação harmónica: o Tonnetz.....	19
2.3.2 A transformada discreta de Fourier para conjuntos de alturas.....	20
2.3.3 Magnitude das componentes Fourier.....	21
2.3.4 Fase das componentes Fourier.....	22
2.3.5 O espaço de fases Fourier .....	22
2.3.6 Aplicações de composição assistida: o <i>PaperTonnetz</i> .....	23
2.4 Súpula .....	24
<b>3. A Esfera de Fases Fourier.....</b>	<b>27</b>
3.1 A Arquitetura da EFF .....	27
3.1.1 Objetivos e idealização .....	28
3.1.2 Concretização .....	28
3.2 Segmentação Harmónica .....	30
3.3 Estruturas de Dados .....	32
3.4 Transformada Discreta de Fourier .....	34
3.5 A Esfera de Fases.....	37

3.6 Agregados Harmónicos.....	40
3.7 Deteção de Padrões.....	42
3.8 Súpula.....	44
<b>4. Schönberg na EFF.....</b>	<b>47</b>
4.1 A Peça em Análise.....	47
4.1.1 Arnold Schönberg.....	48
4.1.2 O atonalismo livre.....	48
4.1.3 <i>Sechs kleine klavierstück</i> , Op.19.....	49
4.2 Op. 19 / n.º 3.....	50
4.2.1 Estrutura formal do andamento.....	50
4.2.2 Definição do espaço de fases.....	52
4.2.3 Interpretação dos agregados harmónicos.....	53
4.2.4 Secção I.....	55
4.2.5 Secção II.....	59
4.2.6 Secção III.....	60
4.2.7 Avaliação global.....	61
4.3 Súpula.....	64
<b>5. A EFF no Apoio à Composição.....</b>	<b>65</b>
5.1 Descrição Global.....	65
5.2 O Método Escolhido.....	66
5.2.1 Seleção do material harmónico.....	67
5.2.2 Definição dos agregados harmónicos.....	68
5.2.3 O processo de composição.....	69
5.3 Súpula.....	71
<b>6. Considerações Finais.....</b>	<b>73</b>
6.1 Conclusões.....	73
6.2 Contributos.....	75
6.3 Perspetivas Futuras.....	76
<b>Referências Bibliográficas.....</b>	<b>79</b>
<b>Anexos.....</b>	<b>83</b>

# Lista de Figuras

2.1: A hierarquia sintática de uma frase em Inglês.	6
2.2: Redução temporal da canção para crianças “ <i>Hush little baby</i> ”.	7
2.3: A escala de CM.	10
2.4: Representação geométrica da semelhança percebida em CM.	11
2.5: Construção de tríades em CM.	12
2.6: Estabilidade perceptual das funções harmónicas da escala de CM.	13
2.7: Duas melodias equivalentes segundo o princípio de equivalência de oitava.	14
2.8: Exemplo demonstrativo da ordenação normal de um CA.	16
2.9: Exemplo demonstrativo da ordenação normal ideal de um CA.	17
2.10: A estrutura intervalar do Tonnetz.	19
2.11: As três maiores componentes Fourier associadas à tríade de CM.	21
2.12: Os acordes de CM e Am representados no espaço de fases $\varphi_{3/5}$ .	23
2.13: O trabalho de Chouvel em papel baseado no <i>Tonnetz</i> .	24
3.1: A macroestrutura da EFF.	29
3.2: A segmentação harmónica da biblioteca <i>Music21</i> .	31
3.3: A contagem do total de CI.	33
3.4: As fases referentes às doze componentes da TDF para um acorde de CM.	35
3.5: A representação gráfica de todas as EH de uma obra.	36
3.6: A esfera de fases desenvolvida para este trabalho.	39
3.7: O algoritmo <i>k-means</i> .	41
3.8: As trajetórias na esfera de fases e o centro dos AH.	42
3.9: As ferramentas de deteção de padrões.	43
3.10: A interface final da aplicação.	44
4.1: A partitura do Op.19 / n.º 3, de Schönberg.	51
4.2: A contagem de CI do Op.19 / n.º 3, de Schönberg.	52
4.3: O posicionamento na EFF da primeira EH do Op.19 / n.º 3, de Schönberg.	54
4.4: A contagem de classes de altura do Op.19 / n.º 3, de Schönberg.	55
4.5: As EH da secção I.	56
4.6: Exemplo de uma mudança de AH num movimento da CI 6 para a CI 4.	58
4.7: As EH da secção II.	59

4.8: As EH da secção III.	60
4.9: Os resultados obtidos para deteção de padrões no Op. 19 / n.º3.	62
4.10: A trajetória mais frequente entre AH no Op.19 / n.º 3.	63
5.1: A estrutura formal do quinteto de cordas escrito com recurso à EFF.	66
5.2: EH escolhida que sobrepõe dois acordes maiores à distância de $1/2t$ .	67
5.3: A contagem de CI presentes em todo o material harmónico escolhido.	68
5.4: O posicionamento das centroides dos AH para os CA escolhidos.	69
5.5: A Utilização do material harmónico escolhido.	70

# Lista de Tabelas

2.1: Os números inteiros de classes de altura associados às suas respectivas notas.	15
2.2: Os números inteiros de CI associados aos seus respectivos intervalos.	18
3.1: O vetor de classes de altura para um acorde de CM.	32
3.2: O conjunto de notas MIDI que representa a tríade de CM.	34
4.1: A correspondência entre as EH da secção I e o seu agregado.	57
4.2: A correspondência entre as EH da secção II e o seu agregado.	60
4.3: A correspondência entre as EH da secção III e o seu agregado.	61
5.1: A correspondência entre todas as EH escolhidas e o seu respetivo AH.	68



# Abreviaturas e Símbolos

TDF	Transformada Discreta de Fourier
EGRH	Espaço(s) Geométrico(s) de Representação Harmónica
EFF	Esfera de Fases Fourier
EH	Estrutura(s) Harmónica(s)
AH	Agregado(s) Harmónico(s)
CA	Conjunto(s) de Alturas
CI	Classe(s) Intervalar(es)
XML	Extensible Markup Language
MIDI	Musical Instrument Digital Interface
M	Maior
m	Menor
P	Perfeita
A	Nota Lá
B	Nota Si
C	Nota Dó
D	Nota Ré
E	Nota Mi
F	Nota Fá
G	Nota Sol
<i>b</i>	Bemol
<i>bb</i>	Duplo Bemol
#	Sustenido
<i>x</i>	Duplo Sustenido
◦	Diminuto



# A Propósito da Terminologia

Com o objetivo prévio de esclarecer alguns dos termos extensivamente utilizados ao longo das próximas páginas (muitos deles forçosamente traduzidos a partir do Inglês) e, com isso, evitar dificuldades de interpretação dos mesmos, explicarei de seguida o seu significado no âmbito desta dissertação, nomeadamente, o que deve inferir de: (1) classes de altura, (2) conjuntos de altura, (3) estruturas harmónicas, (4) ordenação normal, (5) ordenação normal ideal, (6) classe, (7) agregados harmónicos, (8) a forma de notação de tons inteiros e meios tons, (9) a forma de notação para intervalos musicais e (10) a forma de notação para acordes.

Classes de Altura – O termo classes de altura é utilizado no contexto da noção de *pitch classes* tendo exatamente o mesmo significado. Por essa razão ele refere-se aos valores entre 0 e 11 que correspondem a cada uma das notas da escala cromática.

Conjunto de alturas (CA) – O termo conjunto de alturas é utilizado com a mesma significância do termo em Inglês *pitch class sets*. Dessa forma, diz respeito a duas ou mais notas tocadas em simultâneo, e que, durante esta dissertação, aparecem muitas vezes também chamados de tricorde, tetracorde, pentacorde, hexacorde e heptacorde consoante o número de notas que o constitui é três, quatro, cinco, seis ou sete, respetivamente. Como constatado na secção anterior, os conjuntos de alturas são abreviados para CA.

Estrutura harmónica (EH) – A expressão estrutura harmónica é utilizada meramente para efeito de variedade dos termos mencionados ao longo da escrita e tem, por essa razão, um significado em tudo semelhante ao de conjuntos de altura. No fundo, sempre que se lê estruturas harmónicas, deve-se entender a referência a duas ou mais notas tocadas simultaneamente, como que se de um acorde se tratasse. As estruturas harmónicas são abreviadas para EH.

Ordenação normal – No âmbito da teoria musical pós-tonal enunciada por Forte (1980), o termo ordenação normal é referido nestas páginas de acordo com a noção de *normal order*, e que será explicado em detalhe no Capítulo 2.

Ordenação normal ideal – No âmbito da teoria musical pós-tonal enunciada por Forte (1980), o termo ordenação normal ideal é referido nestas páginas de acordo com a noção de *best normal order*, e que será explicado em detalhe no Capítulo 2.

Classe – No âmbito da teoria musical pós-tonal enunciada por Forte (1980), o termo classe é referido nestas páginas de acordo com a noção de *set class*, e que será explicado em detalhe no Capítulo 2.

Agregado harmónico (AH) – O termo agregado harmónico é utilizado para nos referir a grupos constituídos por vários conjuntos de alturas ou várias estruturas harmónicas. Um dos objetivos deste trabalho é agrupar a informação harmónica em diferentes grupos, sendo eles chamados, desta feita, de agregados harmónicos. Estes são abreviados para AH.

Tons e meios tons – A forma de notação para tons inteiros ou meios tons é  $1t$  e  $1/2t$ , respetivamente.

Intervalos – Os intervalos musicais são notados nas páginas seguintes através do seu número correspondente seguido de M, m ou P, caso o intervalo seja maior, menor ou perfeito. Por exemplo, o intervalo de quinta perfeita é escrito como 5<sup>a</sup>P.

Acordes e tonalidades – Os acordes ou tonalidades referidos nesta dissertação são escritos através da letra correspondente à sua fundamental (A, B, C, D, E, F e G) seguidos da letra M, para acordes e tonalidades maiores, e m, para menores. A utilização da letra M para a designação de acordes maiores, embora pouco utilizada (normalmente a sua omissão já implica a ideia de acorde maior), foi, ainda assim, utilizada para evitar ambiguidades com as referências a notas individuais. Exemplificando, o Acorde de dó menor é notado como Cm e o de lá maior como AM.

# 1. Introdução

A música e a coerência das suas progressões harmónicas vivem, largamente, de um conjunto de regras sintáticas muitas vezes escondidas ou subentendidas, que garantem um desenrolar, com foco e direção, dos sons no tempo. No caso da música tonal, o conhecimento existente sobre tais regras é bastante vasto o que nos permite identificar, por exemplo, diferentes funções harmónicas que um determinado acorde pode desempenhar dependendo do contexto tonal em que se encontra. Este entendimento é, em contrapartida, muito mais escasso nas estruturas harmónicas (EH) da música atonal do início do séc. XX, tornando-se necessário, desse ponto de vista, estudar a sua sintaxe de uma forma mais aprofundada. Idealmente, esse estudo deve identificar os conjuntos de alturas (CA) de uma obra atonal em diferentes agregados harmónicos (AH), interpretando, desse modo, aquilo que EH pertencentes a um mesmo agregado têm em comum, mas também, por outro lado, perceber se se identificam movimentos padronizados entre AH de forma a que se possam deduzir regras de organização sintática para esses movimentos. Esta ideia, a de procurar padrões nas movimentações harmónicas de obras atonais pode, à primeira vista, parecer bastante contraintuitiva, porque os compositores que iniciaram esse período estético, como por exemplo Arnold Schönberg, evitavam a todo o custo a ideia de utilizar as lógicas de pensamento harmónico funcional dos séculos anteriores. No entanto, a música referente, por exemplo, ao período do atonalismo livre deste compositor está muito longe de ser música aleatória e, por essa razão, também nela existe lógica, coerência e padrões, mas, como referido anteriormente, bastante menos conhecidos e sistematizados.

Com isto em mente, considerou-se que um dos caminhos possíveis para este estudo pode ser traçado através da representação dos CA de uma peça atonal num espaço geométrico que permita compará-los, avaliar a sua distância relativa e detetar padrões nas suas trajetórias. Alguns espaços geométricos de representação harmónica (EGRH) não são uma ideia recente na literatura, no entanto, são-no quando criados através da informação de fase calculada através da transformada discreta de Fourier (TDF) para CA e têm demonstrado resultados muito encorajadores nos últimos anos para análises harmónicas do período tonal ([Yust, 2015](#)).

Para além disto, algum entendimento a propósito da sintaxe atonal pode também ser motivador da criação de uma ferramenta de apoio à composição musical que, por exemplo, sugira

ao compositor diferentes possibilidades harmónicas de acordo com os movimentos padronizados detetados ao longo do estudo. Ora, é exatamente sobre todas estas áreas de estudo que a presente dissertação se debruça e para a qual serão enunciados seguidamente: a motivação para a sua escrita (§1.1), o problema e os objetivos que estão no seu centro (§1.2), a sua metodologia de investigação (§1.3) e, por fim, a estrutura global do texto (§1.4).

## 1.1 Motivação

Até onde a minha memória consegue alcançar, sempre tive um interesse especial não só por ouvir e apreciar música, mas também por a pensar, por entender a sua construção e o seu funcionamento e por refletir nos seus fundamentos teóricos. Recordo-me, por exemplo, de ouvir, ainda em criança – sem ter sequer iniciado os meus estudos musicais – a tocata e fuga em Dm de Bach e, logo a seguir ao fascínio de a ter ouvido pela primeira vez, me interrogar de imediato como é que era possível tal obra ser escrita. Este interesse de criança por construções musicais levar-me-ia, muitos anos mais tarde, a estudar composição e, talvez contraditoriamente, à medida que ia obtendo respostas para as perguntas teóricas que antes fazia, muitas outras surgiam, continuando, até aos dias de hoje, a curiosidade pelo entendimento e pela construção musical completamente intacta.

Ora, é exatamente nesse período de estudos composicionais em que pude mergulhar verdadeiramente na música do séc. XX e na multiplicidade das suas correntes estéticas, estudando-as, mas também escrevendo música de acordo com elas. Por essa razão, pude contactar pela primeira vez com a teoria musical pós-tonal, nomeadamente com o período do atonalismo livre de Arnold Schönberg. Mais tarde, já no Mestrado em Multimédia e durante as aulas de geração automática de música, comecei a refletir um pouco mais a propósito da sintaxe harmónica e do que ela pode significar para o entendimento do discurso musical em diferentes períodos e percebi que muitas das lógicas estruturais que já conhecia são na verdade integrantes de uma estrutura sintática mais abrangente para a qual não tinha ainda conhecimento suficiente.

A motivação para o desenvolvimento do estudo que se pode ler nas presentes páginas advém precisamente da conjugação destes três aspetos, ou seja, por um lado, a minha profunda curiosidade pelo entendimento teórico da música e das estruturas lógicas que a sustentam, por outro, a proximidade que tenho desde há vários anos com linguagens musicais atonais ao longo dos séc. XX e XXI, quando associados à minha descoberta recente sobre a importância da sintaxe harmónica, incentivam largamente dedicar esta dissertação ao desenvolvimento de uma ferramenta que permita averiguar que lógicas de organização sintática estão presentes na música do período pós-tonal.

## 1.2 Problema e Objetivos da Investigação

No âmbito da música tonal é evidente a existência de múltiplos níveis de organização sintática que regulam a criação de estruturas musicais como acordes e os seus respetivos encadeamentos. São, por exemplo, bem conhecidas as funções harmónicas (tónica, dominante, subdominante) que um mesmo acorde pode desempenhar em construções frásicas distintas. No entanto, essa organização sintática (assumindo, naturalmente, características diferentes da anterior) é muito menos compreendida na música referente ao período pós-tonal de início do séc. XX, exigindo-se, portanto, o seu estudo e conhecimento.

Com base neste problema a presente dissertação tem como principal objetivo desenvolver uma ferramenta de análise harmónica baseada na informação de fase derivada da TDF para CA que permita investigar e levantar hipóteses quanto à organização da sintaxe harmónica de uma obra atonal. Tal ferramenta, para além do objetivo primeiro agora enunciado, deverá possibilitar, entre outros: (1) a disposição num espaço geométrico de toda a informação harmónica de uma peça de forma a simplificar a sua interpretação e analisar comparativamente o posicionamento no espaço de diferentes CA; (2) criar agregados distintos a partir dessa informação, ou seja, agrupar todas as EH em secções distintas e, através delas, identificar padrões no trajeto harmónico da obra; (3) refletir sobre a interpretação sintática que tais padrões sugerem, levantando para eles hipóteses com fundamento musical e (4) testar a utilidade da ferramenta no apoio ao processo criativo do compositor através do conjunto de conclusões sintáticas identificadas, discriminadamente, facilitando-lhe a criação de progressões harmónicas a partir do material por ele escolhido. Em suma, todos estes objetivos perfazem o verdadeiro propósito desta dissertação, pretendendo com eles garantir a criação de uma ferramenta que nos torne mais próximos de conseguir um contributo relevante para o entendimento da sintaxe harmónica no universo da música atonal.

## 1.3 Metodologia de Investigação

Face ao problema enunciado na secção 1.2 foi escolhido um percurso metodológico com as seguintes etapas: em primeiro lugar, seleccionar uma obra representativa do movimento atonal referente ao período do atonalismo livre de Schönberg; de seguida, editar essa partitura num *software* de notação musical e exportá-la em formato *.XML* para que a sua informação possa ser tratada e manipulada; em terceiro lugar, escrever todo o código em *Python* (2020) que possibilita segmentar as EH da peça, calcular para todas elas a TDF e as suas respetivas informações de fase e dispô-las num espaço geométrico desenvolvido em *Processing* (2020) para analisar o trajeto harmónico da obra e detetar nele padrões de movimentação; seguidamente, analisar e compreender os resultados obtidos e formular hipóteses sintáticas que os possam explicar e, por

último, testar a eficácia do sistema desenvolvido e dos resultados obtidos no processo criativo do compositor, ajudando-o a criar encadeamentos harmônicos no seu próprio material.

## **1.4 Estrutura do Texto**

As páginas seguintes deste trabalho estão divididas em cinco capítulos. No Capítulo 2, é possível ler-se a revisão da literatura onde se exploram três grandes áreas distintas, nomeadamente a sintaxe musical e a sua comparação com a sintaxe linguística, a teoria musical pós-tonal mais relevante no contexto deste trabalho e os EGRH construídos através da TDF para CA. Por seu lado, o Capítulo 3 descreve em detalhe o funcionamento de um sistema computacional a que chamei esfera de fases Fourier (EFF) que dispõe a informação harmónica de uma peça num espaço tridimensional e, através dele, analisa padrões no trajeto harmónico do material projetado. O Capítulo 4, por seu turno, utiliza esse sistema para análise de uma peça de início do séc. XX referente ao período do atonalismo livre de Schönberg e levanta hipóteses para a organização sintática da sua harmonia. Distintamente, o Capítulo 5 testa como é que a EFF pode ser utilizada como uma ferramenta de apoio ao processo de composição e, finalmente, o Capítulo 6 engloba as conclusões, principais contributos e perspetivas de trabalho futuro que se podem extrair desta dissertação.

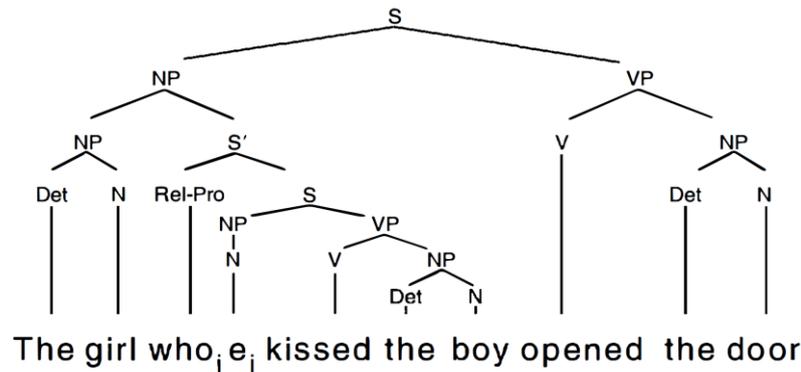
## 2. Sintaxe, Teoria Atonal e Fourier

Este capítulo trata da revisão bibliográfica associada a este estudo. Como tal, ele explora os conceitos, ferramentas e tecnologias de maior relevo na literatura no que concerne aos três principais pilares deste trabalho, ou seja, a sintaxe musical e as suas principais características (§2.1), a teoria musical pós-tonal (§2.2), mas também os espaços de representação harmónica através da TDF (§2.3). Por último, resumem-se as principais ideias discutidas neste estado da arte, salientando, uma vez mais, a sua importância para o estudo do problema (§2.4).

### 2.1 Música e Linguagem: A Sintaxe Musical

A sintaxe, no campo da linguística, é a área de estudo que investiga a forma como diferentes palavras, pertencentes ao léxico de uma determinada língua, se relacionam entre si com vista a garantir a construção de frases lógicas e inteligíveis. Mas, com uma natureza bastante diferente, por certo, também na música encontramos lógicas que permitem a combinação de vários sons em simultâneo, bem como a sua sucessão ao longo do tempo de forma estruturada e coesa, uma vez que qualquer ouvinte, mesmo sem conhecimento musical, é capaz de nela reconhecer lógica e coerência. No entanto, na música podem ser aplicadas teorias sintáticas a qualquer das suas constituintes, ou seja, à melodia, harmonia, ritmo, métrica, forma ou até a timbre ou dinâmicas. No fundo, as abordagens sintáticas têm sido aplicadas ao que acontece em sequências musicais (prevendo combinações de acordes ou notas) ao invés do momento em que acontece ([Rohrmeier, 2018](#)).

O problema, ainda assim, reside no facto de este conjunto de lógicas sintáticas ser bem menos conhecido e sistematizado no âmbito da música atonal, nomeadamente a referente ao atonalismo livre, ao contrário do conhecimento sintático existente para a música funcional anterior ao séc. XX. No entanto, para o estudo mais aprofundado da sintaxe harmónica atonal, é útil realçar previamente os aspetos mais relevantes que já existem na literatura relativos à música tonal e, por essa razão, esta secção 2.1 revê o que ela nos diz a propósito da relação entre sintaxe musical e linguística.



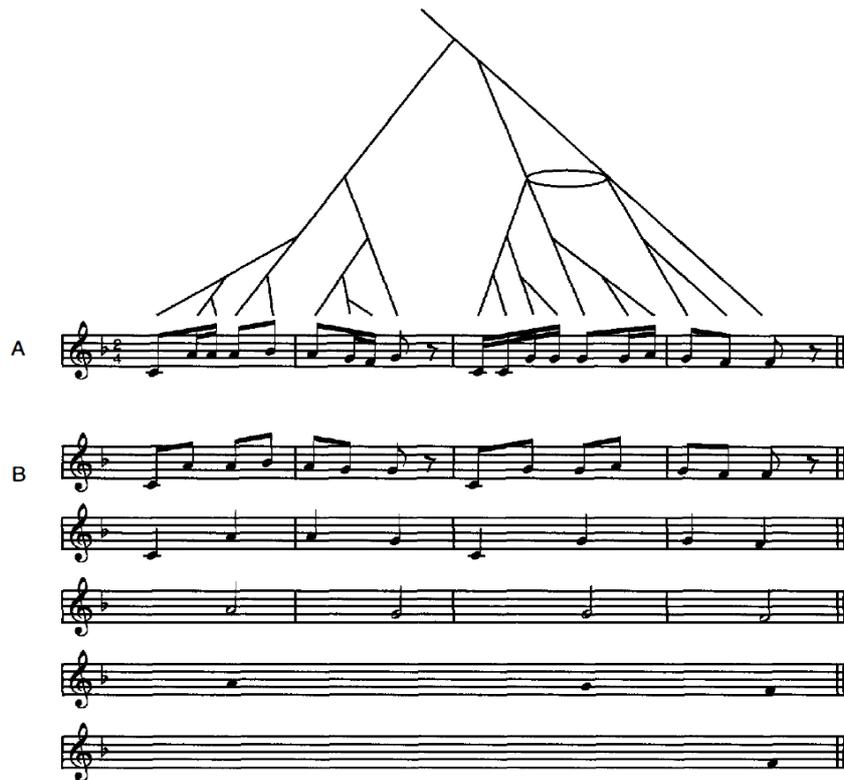
**Figura 2.1:** A hierarquia sintática de uma frase em Inglês. Como legenda: S=sentence, NP=noun phrase, VP=verb phrase, S'=sentence modifier, N=noun, V=verb, Det=determiner, Rel-Pro=relative pronoun ([Patel, 2010](#)).

### 2.1.1 Sintaxe musical e linguística: diferenças e semelhanças

A fusão entre sintaxe linguística e sintaxe musical é amplamente mencionada entre músicos, sobretudo em contexto de análise musical. Não raras vezes, são referidos termos que se cruzam entre as duas áreas como “frase” e “frase musical” ou “pergunta, resposta” e “antecedente, consequente”. No entanto, a literatura mostra-nos como tais comparações podem muitas vezes ser perigosas e levar-nos a conclusões precipitadas como, por exemplo, a tentativa de encontrar musicalmente conceitos que mimetizem a construção frásica na linguagem através de verbos, nomes e adjetivos, apenas para nomear alguns. Torna-se, assim, indispensável sistematizar como verdadeiramente se interligam sintaxe musical e linguística e perceber se existem entre si pontos de contacto, como também problemas associados à sua comparação.

#### I. Diferenças

Em 1973, Bernstein protagonizou um conjunto de palestras intituladas “The Unanswered Question”. Em “Musical Syntax”, a segunda da série, o compositor elabora um conjunto de equiparações entre conceitos musicais e linguísticos de forma a estabelecer um paralelismo exato entre as estruturas constituintes das duas áreas. No entanto, hoje é claramente aceite que uma das principais diferenças entre ambas as sintaxes é a inexistência na música de equivalentes a categorias gramaticais como verbos, nomes ou adjetivos ([Patel, 2010](#)), bem como as funções que estes desempenham em frases, como, por exemplo, a de sujeito ([Jackendoff, 2002](#)). Por outras palavras, não existem na música estruturas inquestionavelmente análogas às suprarreferidas, e que se possam interrelacionar. Tal exercício, o de tomar o exemplo da linguística e procurar na música os seus inquestionáveis equivalentes, é, tal como Lerdahl e Jackendoff ([1996](#)) disseram, “an old and largely futile game” (p. 5).



**Figura 2.2:** Redução temporal da canção para crianças “*Hush little baby*” e respetiva estrutura hierárquica. As pautas inferiores mostram sucessivas simplificações em termos de notas mais importantes na frase. (Patel, 2010)

Para além do ponto anterior, a forma como frases nos dois domínios são constituídas é manifestamente diferente, ou seja, a relação e combinação de estruturas sintáticas obedece a regras e hierarquias não equiparáveis (Patel, 2010). Por exemplo, no âmbito da linguística, como se pode verificar na Figura 2.1, os ramos da árvore hierárquica unem-se de tal forma que pequenos constituintes, juntos, formam estruturas gramaticais mais complexas. Exemplificando, um determinante (Det) e um nome (N) formam uma “noun phrase” (NP – “the door”), uma “noun phrase” e um verbo (V) formam uma “verb phrase” (VP – “opened the door”), constituindo, assim, estruturas cada vez mais complexas do ponto de vista sintático. De forma contrária, as árvores sintáticas musicais preveem que quanto maiores forem os seus ramos mais importantes são as notas a si associadas e, naturalmente, quanto mais curtas forem essas ramificações menos importantes são as suas notas (Large *et al.*, 1995). Dessa forma, não existe numa frase musical a ideia de estruturas sintáticas independentes cada vez mais complexas, mas, ao invés, a indicação de notas estruturalmente mais importantes a cada momento (Patel, 2010). No caso da Figura 2.2, assume maior preponderância estrutural a descida melódica A-G-F.

Por fim, a terceira diferença a salientar é o diferente espaço para ambiguidade sintática na música e na linguagem. Na música, um único acorde, por exemplo, pode ser ouvido com diferentes funções estruturais, consoante a tonalidade em análise. É o caso das modulações através

de acordes pivot, ou seja, acordes que apresentam uma determinada função harmónica numa tonalidade, mas que, por serem partilhados com outra, se tornam escolhas óbvias para preparar uma modulação, introduzindo, deste modo, ambiguidade à sua função estrutural numa frase pois podem ser vistos como parte integrante de tonalidades distintas. Quer isto dizer que esse mesmo acorde, quando utilizado como acorde pivot em contexto de uma modulação, ele poderá sempre ser visto de forma ambígua, ou seja, tendo uma função harmónica para a tonalidade de origem, e uma outra para a tonalidade destino. A propósito desta ideia, Krumhansl (1992) diz que:

“a chord may be heard simultaneously in its multiple roles in different keys, with the effect that modulations between closely related keys are easily assimilated. The role of a chord need never be disambiguated” (p. 199).

Deste modo, na música tonal, esta característica está não só presente, como também procura ser aproveitada pelos compositores por razões expressivas e estéticas. No entanto, tal ambiguidade sintática parece ter menos espaço no campo da linguística, pois, de alguma forma, o nosso sistema cognitivo procura interpretar construções sintáticas ambíguas sob a forma de uma frase com uma única possibilidade estrutural. Tome-se o exemplo da frase “The fireman left the building with the large sign,” em que a generalidade dos leitores ou ouvintes assume “the large sign” como um modificador que qualifica “building” e não a “fireman” (Patel, 2010).

## II. Semelhanças

Tal como foi abordado no ponto anterior existem várias diferenças na sintaxe associada à linguagem e à música, mas importa perceber também pontos de contacto entre as duas áreas. Estes existem, sobretudo, na estrutura hierárquica e lógica de ambas, manifestando-se sempre de formas distintas.

Uma das semelhanças é a existência de múltiplos níveis de organização sintática, ou seja, em ambos os domínios existem sucessivos conjuntos de regras que permitem dar sentido à forma como palavras ou sons se articulam de forma inteligível (Patel, 2010). Por exemplo, na linguagem existem princípios básicos sintáticos que definem a forma como pequenas unidades lexicais se agrupam formando palavras e como estas se interligam com o objetivo de criar frases com estrutura, lógica, hierarquia e significado. Também na música tonal, paralelamente, é evidente a existência de estruturas sintáticas que nos permitem sistematizar como diferentes sons se conjugam para formar acordes, como é que estes acordes se sucedem em progressões harmónicas coerentes e perceptualmente satisfatórias, bem como tais progressões sugerem um espaço tonal próprio capaz de possibilitar, até, a modulação para um outro distinto. Em súpula:

“In both domains, this multilayered organization allows the mind to accomplish a remarkable feat: a linear sequence of elements is perceived in terms of hierarchical relations that convey organized patterns of meaning” (Patel, 2010, p. 264).

Um outro paralelismo claro entre os domínios musical e linguístico é a presença de funções: harmónicas, no primeiro, e gramaticais, no segundo. Em pormenor, quando analisamos uma frase são facilmente identificáveis, por exemplo, as funções de sujeito ou predicado. Note-se que diferentes palavras podem ter uma destas funções e, ao invés, a mesma palavra, desempenhar uma função distinta em diferentes frases (Patel, 2010). Uma das resultantes desta questão é a necessidade de concordância<sup>1</sup> entre o sujeito e predicado em várias línguas e pessoas verbais (Jackendoff, 2002). Similarmente, encontramos na música tonal funções harmónicas. Estas funções harmónicas são amplamente referidas como tónica, subdominante e dominante, ou, em notação romana, I, IV e V, respetivamente (Patel, 2010). Tal como nas funções gramaticais, também diferentes acordes podem desempenhar uma mesma função consoante a tonalidade em análise (veja-se o caso da tonalidade de GM em que o acorde [GBD] tem a função de tónica, mas, na tonalidade de FM, o acorde que desempenha essa mesma função é [FAC]) e, por outro lado, o mesmo acorde pode também desempenhar funções distintas de acordo com o contexto tonal em se insere (como acontece na tonalidade de CM onde o acorde [CEG] tem a função de tónica, mas, na tonalidade de FM a mesma EH já desempenha a função de dominante). Tais funções não são exclusivas de tonalidades maiores, mas são, também, extensíveis a frases musicais em tonalidades menores (Krumhansl & Kessler, 1982). No fundo, as funções harmónicas dos acordes representam uma propriedade percetual que deriva das suas relações com outros acordes vizinhos e, por isso, é possível dizer que também na música encontramos um conjunto de funções dependentes do seu contexto que formam uma estrutura lógica na criação de sequências ou frases inteligíveis (Patel, 2010).

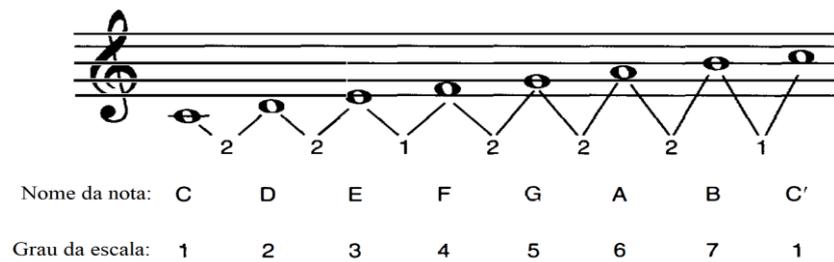
### 2.1.2 A organização estrutural da sintaxe musical

Mudando o foco da análise somente para a música, é nela evidente a existência de múltiplos níveis de organização sintática que regulam e possibilitam a criação de estruturas musicais, como por exemplo, acordes ou sequências de acordes coerentes (Patel, 2010). Importa por isso sistematizar essas regras e perceber como é que elas governam o desenvolvimento melódico, como é que ditam a criação de acordes ou, até, num patamar superior, a que nível influenciam a criação de música tonal.

Dois exemplos principais dessas estruturas são as escalas utilizadas para a construção de melodias e os acordes para o desenvolvimento harmónico. Nas páginas seguintes, é, por essa razão, descrito algum do conhecimento já sistematizado para a criação destas duas grandezas musicais, ou seja, a melodia e a harmonia.

---

<sup>1</sup> A concordância entre sujeito e predicado é feita a diferentes níveis. Em primeiro lugar, ela deve existir em número, ou seja, o predicado deve estar em harmonia com o sujeito, fazendo corresponder o número da pessoa gramatical do verbo com o sujeito da frase. No entanto, esta concordância não é suficiente, porque a expressão “O João fiz”, embora o sujeito e predicado concordem em número (são ambos singular), não é correta. É também necessária concordância em pessoa entre sujeito e predicado. Quer isto dizer que, em Português, os predicados assumem formulações diferentes consoante o sujeito e também a esse nível devem concordar. O exemplo anterior é corrigido através da concordância em pessoa e ficando “O João fez” (FLIP, 2020).



**Figura 2.3:** A escala de CM. Os números entre notas simbolizam a quantidade de meios tons entre elas e a sequência [2212221] é o que caracteriza uma escala maior (Patel, 2010).

### I. Estrutura melódica: escalas

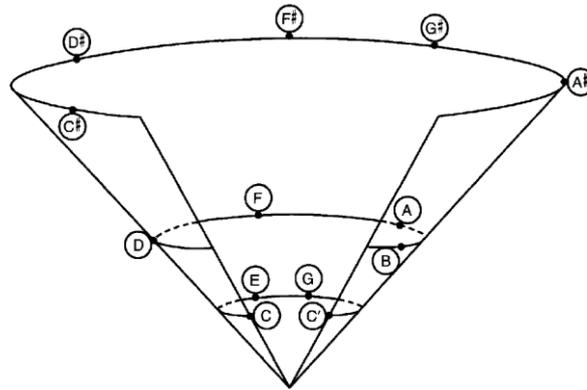
Um dos níveis mais elementares de organização sintática na música diz respeito às estruturas conhecidas como escalas. Elas pressupõem, à exceção da escala cromática, uma sequência de notas parcialmente selecionada a partir dos doze meios tons encontrados numa oitava e a existência de um padrão intervalar entre cada uma dessas notas, padrão esse fulcral para a identificação da escala em análise (Patel, 2010). No exemplo da Figura 2.3, é possível ver-se uma escala de CM uma vez que o padrão intervalar de meios tons [2212221] é o fator transversal a essa mesma escala. Estes padrões, no âmbito da música tonal, são tradicionalmente assimétricos e por isso incluem-se neles os modos antigos<sup>2</sup> ou as escalas maiores e menores, mas, no âmbito da música do séc. XX, várias escalas com conteúdo intervalar simétrico, a escala octatônica, hexáfona ou modos de transposição limitada<sup>3</sup>, foram largamente utilizadas por diversos compositores como Bártok, Debussy ou Messiaen, respetivamente.

No âmbito de muitas tradições musicais, não só na prática ocidental, é curioso destacar que os diferentes graus das escalas não são considerados nem sentidos como igualmente relevantes e estáveis no seu contexto. Isto implica, por exemplo, que o primeiro grau melódico (na Figura 2.3, a nota C) é sentido como o ponto de maior estabilidade e repouso numa qualquer frase musical construída com esta escala. Por isto mesmo, tal transversalidade parece sugerir que a organização estrutural melódica em torno de um centro estável é uma característica intrínseca à mente humana (Rosch, 1975; Krumhansl, 1979).

Para entender a diferente estabilidade que os graus melódicos perceptualmente assumem, Krumhansl (1979) estudou a questão e criou o conceito de distância perceptual entre as notas de

<sup>2</sup> Os modos antigos, também referenciados como modos Gregos, são escalas diatônicas, ou seja, escalas de sete notas constituídas por cinco intervalos de  $1t$  e dois de  $1/2t$ , com nome dórico, frígio, lídio, mixolídio, eólio, lócrio e jónio, construídas como extensão de uma escala maior. Por exemplo, a escala de CM é o modo jónio e se reescrevermos as suas notas a partir de D temos o modo dórico, de E o frígio, e, assim, sucessivamente (Michels, 2001).

<sup>3</sup> Os modos de transposição limitada de Messiaen são sete e dizem respeito a sequências de notas de número variável em que a sua organização intervalar é simétrica. É exatamente esta simetria que confere aos modos de Messiaen um número limitado de transposições porque implica a repetição das mesmas classes de alturas depois de algumas transposições (Messiaen, 1944).



**Figura 2.4:** Representação geométrica da semelhança percebida entre as diferentes notas da escala de CM em ambiente tonal (Patel, 2010).

uma escala. Para isso realizou uma experiência onde os participantes eram inicialmente introduzidos ao contexto harmónico de CM, ouvindo a respetiva escala ascendente ou descendente e, de seguida, ouviam duas notas isoladamente com o objetivo de identificar o quão perceptualmente próximas essas duas notas estavam. Com os resultados foi, então, possível gerar uma representação gráfica tridimensional visível na Figura 2.4.

Estes resultados indicam uma semelhança perceptual muito grande entre as notas C, E e G, seguidos num novo patamar das restantes notas pertencentes à escala de CM, ou seja, as notas D, F, A e B e, por fim, numa camada ainda mais distante, as notas C#, D#, F#, G# e A#, que não fazem parte da escala (Krumhansl, 1979). Por fim, é também de notar que a proximidade percebida entre notas pelos participantes, nada tem que ver com a proximidade acústica, uma vez que, por exemplo, as notas C e C#, mais próximas na grandeza da frequência, estão em patamares muito distantes do ponto de vista perceptual (Patel, 2010).

## II. Estruturas harmónicas: acordes

No que à sintaxe musical diz respeito, uma das mais importantes características é a combinação vertical ou, por outras palavras, combinação simultânea de diferentes sons de uma escala. A esta combinação chama-se, regularmente, harmonia. Tipicamente, a construção dos diferentes graus harmónicos de uma escala, que pode ser vista na Figura 2.5, associada à tonalidade de CM, é feita através da sobreposição de terceiras ( $3^{\text{a}}$ s) dada uma nota fundamental, e criando, assim, estruturas de três sons chamadas tríades. Note-se que, devido ao facto de as escalas maiores serem assimétricas, estas  $3^{\text{a}}$ s acrescentadas podem ser maiores ou menores, ou seja, terem quatro ou três meios tons, respetivamente. Este aspeto é de vital importância, pois possibilita que se formem para uma determinada escala acordes com tipologias diferentes (maiores, menores ou diminutos), conferindo aos encadeamentos harmónicos maior contraste expressivo (Patel, 2010). Cada acorde recebe, por norma, o nome da sua fundamental, ou seja, a nota mais grave da estrutura de terceiras, mas não impossibilitando a escrita de inversões, ou seja,

Nome do acorde:	C	d	e	F	G	a	b <sup>o</sup>	C'
Função harmónica:	I	ii	iii	IV	V	vi	vii <sup>o</sup>	I

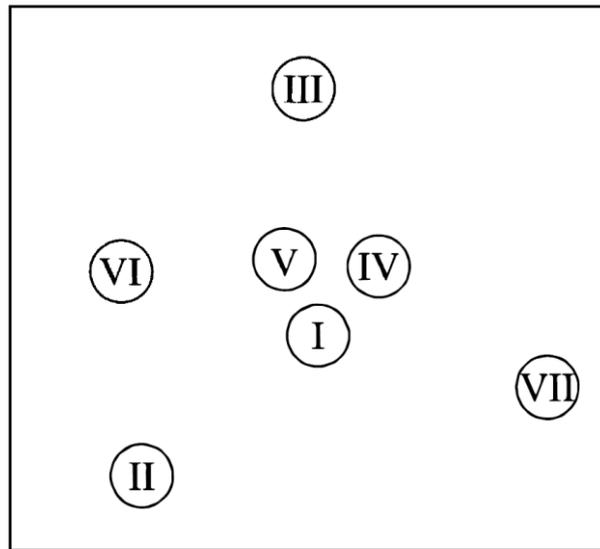
**Figura 2.5:** Construção de tríades a partir de cada uma das notas da escala de CM. Os números entre notas representam a distância em meios tons entre si. Letras maiúsculas são referentes a acordes maiores, minúsculas a acordes menores e, letras com o símbolo “<sup>o</sup>” dizem respeito a acordes diminutos. Em numeração romana lê-se a função referente ao acorde no contexto desta escala (Patel, 2010).

a reorganização da sequência de notas numa ordem diferente (para a tríade [CEG], correspondente a CM, as três notas podem ser reorganizadas para [EGC], a primeira inversão, e para [GCE], a segunda inversão do mesmo acorde). Uma outra transformação muito frequente é o acréscimo de mais uma terceira à tríade de forma a criar, por exemplo, acordes de sétima. Estes acordes, usualmente associados ao V grau da escala, são da maior importância no âmbito da música tonal, pois representam um tremendo ponto de atração para o acorde da tônica, ou I grau, que é identificado como o ponto de maior repouso e estabilidade (Krumhansl & Kessler, 1982). Este acorde de sétima, ou V7, pode também ser visto na Figura 2.5.

Todos estes aspetos suprarreferidos dizem respeito à verticalidade dos acordes, ainda que, paralelamente, exista um conjunto de regras lógicas de funcionamento sintático dos acordes do ponto de vista horizontal, ou seja, o seu encadeamento e respetivas progressões harmónicas com direccionalidade (Huron, 2008). Tome-se, como exemplo, um dos movimentos harmónicos mais tradicionais e simples da história da música tonal, a progressão I-V7-I, onde a ideia de progresso está associada à passagem do I grau para o acorde instável do V, seguido do retorno ao I com sensação de repouso. Um outro encadeamento típico e mais complexo, como as sequências por quintas, tem, tal como o nome indica, uma estrutura de organização sintática em que as fundamentais dos acordes desta progressão distam entre si quintas tonais<sup>4</sup> descendentes que percorrem todas as notas da escala. Voltando ao exemplo de CM, as fundamentais desta progressão seriam C-F-B-E-A-D-G-C e a sequência de acordes propriamente dita I-IV-vii<sup>o</sup>-iii-vi-ii-V7-I.

Tal como abordado no ponto I, também os diferentes graus harmónicos são percebidos com diferente importância estrutural e, portanto, mais ou menos próximos uns dos outros ou estáveis. Krumhansl (1982) desempenhou uma experiência em tudo semelhante à descrita no ponto I e percebeu como os ouvintes percecionam os acordes do I, IV e V graus como mais estáveis (de

<sup>4</sup> A expressão quintas tonais, neste contexto, diz respeito a movimentos por quintas descendentes do baixo, mas a qualidade dessa quinta, ou seja, se é perfeita ou diminuta, é dada pela tonalidade em que a música se encontra escrita. Inversamente, em sequências de quintas reais, o baixo move-se em quintas descendentes, mas sendo elas sempre 5<sup>a</sup> sP (Michels, 2001).



**Figura 2.6:** Estabilidade perceptual das funções harmônicas da escala de CM. Os acordes do I, IV e V graus são reconhecidos como mais estáveis e mais próximos. Na imagem a numeração romana é toda maiúscula não diferencia a tipologia do acorde (Krumhansl *et al.*, 1982).

salientar a utilização na experiência do V grau sem sétima) e os restantes num segundo patamar de estabilidade. A representação gráfica destas distâncias pode ser vista na Figura 2.6.

## 2.2 Teoria Musical Pós-Tonal

No início do séc. XX, aquilo que muitos compositores já antecipavam (por exemplo, Liszt, na sua peça “Bagatelle without tonality”, de 1885) aconteceu. Devido ao sucesso do cromatismo em toda a música do séc. XIX, não sobrou alternativa aos compositores senão abandonar o sistema tonal de composição que prevaleceu durante séculos ([Kostka, 2018](#)). Este passo, dado essencialmente por compositores como Arnold Schönberg ou Anton Webern, deu o mote para o nascimento da música atonal e, embora já se tenha passado mais de um século desde esse período, é ainda muito ténue o nosso conhecimento teórico sobre esta música, sobretudo quando comparado com o entendimento de que dispomos do sistema tonal ([Straus, 2016](#)). Há, no entanto, um consenso alargado relativamente a algumas ferramentas de análise teórica de música atonal que importa destacar neste estudo. Entre elas, abordarei de seguida nesta secção 2.2, o que a literatura nos diz a propósito de alguns princípios básicos da teoria pós-tonal, bem como conceitos como CA, classes ou vetor intervalar.

The image shows a musical score with two systems, labeled 'a.' and 'b.'. System 'a.' is for Violin 1, showing a melodic line in 4/4 time with various notes and rests. System 'b.' consists of two staves: Violin 1 and Cello. The Violin 1 staff in 'b.' contains a melodic line that is an octave lower than the one in 'a.'. The Cello staff in 'b.' contains a melodic line that is two octaves lower than the one in 'a.'. Dotted lines connect the notes in 'a.' to their corresponding notes in 'b.', demonstrating the principle of octave equivalence. The notes in 'b.' are labeled with '275' and '2' to indicate their frequency or position relative to the original notes.

**Figura 2.7:** Duas melodias equivalentes segundo o princípio de equivalência de oitava presentes no Quarteto de Cordas N.º 4 de A. Schönberg (Straus, 2016).

### 2.2.1 Princípio da equivalência de oitava

O intervalo de oitava é, acusticamente, muito característico e consonante. É, na verdade, o intervalo mais consonante de todos e cuja nota mais aguda tem o dobro da frequência da mais grave. Esta característica faz de notas separadas por uma ou mais oitavas, de alguma forma, equivalentes (Straus, 2016). É, por essa mesma razão, que a essas notas, independentemente da oitava em que se encontram, se dá exatamente o mesmo nome, ou seja, a nota A, por exemplo, é sempre chamada dessa forma não relevando a oitava em que encontra. Este facto não é de todo casuístico porque notas separadas por uma oitava soam, efetivamente, sobretudo para ouvidos pouco treinados, como exatamente o mesmo som.

No entanto, este princípio da equivalência de oitava não é sinónimo de equivalência de identidade (Straus, 2016). Esta ideia pode ser comprovada através da Figura 2.7. Nesse exemplo, embora se constate que ambas as melodias são equivalentes em oitava, não podemos afirmar que sejam melodias iguais nem tão pouco semelhantes. Na verdade, elas têm características radicalmente diferentes. Por exemplo, a melodia b. está disposta num registo muito mais amplo que a a. e, consequentemente, com intervalos e saltos muito mais pronunciados. Como se não bastasse também o ritmo de ambas é muito distinto e a melodia b. é partilhada entre dois instrumentos diferentes.

Por todas estas razões, há uma grande diferença entre melodias equivalentes em oitava e melodias com a mesma identidade mas, ainda assim, este princípio está na base de muita teoria da música pós-tonal, como, por exemplo, a ideia de classes de altura que é abordada na subsecção seguinte.

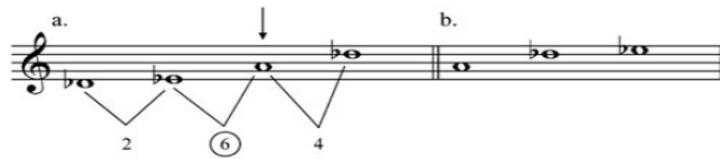
**Tabela 2.1:** Os números inteiros de classes de altura associados às suas respectivas notas ([Straus, 2016](#)).

Número inteiro	Nota
0	C, B#, Dbb
1	C#, Db
2	D, Cx, Ebb
3	D#, Eb
4	E, Dx, Fb
5	F, E#, Gbb
6	F#, Gb
7	G, Fx, Abb
8	G#, Ab
9	A, Gx, Bbb
10	A#, Bb
11	B, Ax, Cb

### 2.2.2 Classes de altura

Assim que se começou o estudo da música referente ao período atonal, percebeu-se a necessidade de um novo paradigma teórico que tratasse as alturas de forma distinta, uma vez que, nesta música, era habitualmente identificado um novo tipo de motivo, chamado muitas vezes de célula ou sonoridade de referência, cuja aparição ocorria melodicamente, harmonicamente ou ainda como uma combinação de ambas, e em diferentes inversões, transposições ou permutações ([Kostka, 2018](#)).

Uma das ideias fundamentais desta nova base teórica para a música pós-tonal é a de equivalência enarmônica. Por outras palavras, ao contrário da música tonal, em que escritas diferentes implicam resoluções também distintas, é completamente indiferente escrever-se, por exemplo, A# ou Bb, uma vez que representam exatamente a mesma classe de altura. Por esta razão, os compositores acabam por fazer as suas opções de escrita simplesmente por razões de facilidade de leitura para o músico que interpreta as suas obras ([Straus, 2016](#)). Como tal, devido ao princípio de equivalência de oitava e ao princípio de equivalência enarmônica, é possível identificar, somente, doze classes de altura distintas e correspondentes a cada uma das notas da escala cromática. Outro aspeto digno de se salientar, é a referência a cada uma destas classes de altura recorrendo a um número inteiro, em vez do nome da nota em questão, pois simplifica largamente a análise das obras deste período. Assim sendo, as notas C, B# e Dbb são representadas pelo número 0 e assim sucessivamente. Na Tabela 2.1, podem ler-se cada uma das doze classes de altura e os seus respectivos números inteiros.



**Figura 2.8:** Exemplo demonstrativo das transformações necessárias para se encontrar a ordenação normal de um determinado CA. Em b. pode ler-se a ordenação normal do conjunto a. (Kostka, 2018).

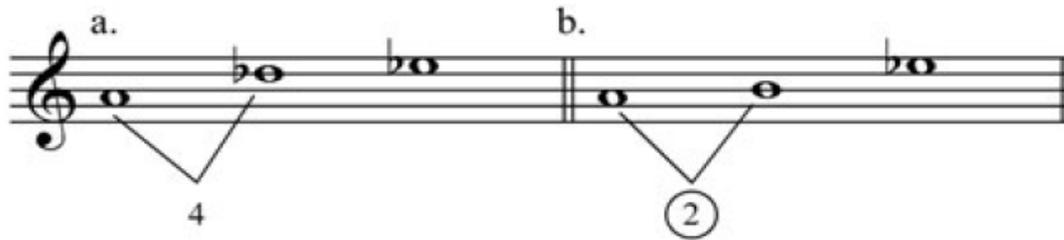
### 2.2.3 Conjuntos de alturas e a noção de classe

O termo CA deriva da ideia de sonoridade referência a que Kostka (2018) se refere. Ele é, portanto, utilizado para um conjunto de notas (duas ou mais) que pode aparecer em simultâneo, ou seja, em forma de acorde, mas também tocadas individualmente ou, por outras palavras, melodicamente. Para entender como podem estas estruturas ser analisadas e compreendidas, importa convocar o trabalho de Forte (1980) na sistematização dos processos necessários ao seu estudo (Kostka, 2018), que culmina no conceito de classe. Neste âmbito, a primeira ideia a abordar é, no entanto, a de ordenação normal.

#### I. Ordenação normal e ordenação normal ideal

A ordenação normal de um CA é uma ideia semelhante à de estado fundamental num acorde em contexto de música tonal. No fundo, para se encontrar a ordenação normal é necessário um processo cujo objetivo passa por perceber qual a inversão em que uma EH complexa se encontra e o que é necessário fazer para a colocar no seu estado fundamental (Kostka, 2018). Esse processo consiste na escrita das notas pertencentes a uma qualquer sonoridade referência de forma inarmónica no âmbito de uma única oitava e em sentido ascendente, como que se de uma escala se tratasse. De seguida, avalia-se o maior intervalo entre as diferentes notas desse conjunto e, posteriormente, a nota superior referente a esse intervalo deve ser considerada a nota inferior da ordenação normal. Na Figura 2.8 é possível identificar que o maior intervalo do conjunto a. se encontra entre as notas Eb e A, e, portanto, A deve ser considerada a nota inferior da ordenação normal. Esta escreve-se entre parêntesis retos com as respetivas classes de altura ou nomes de nota, ou seja, [9, 1, 3] ou [A, Db, Eb].

Depois de encontrada a ordenação normal, é necessário introduzir um outro conceito chamado ordenação normal ideal. Ele é relevante, pois é uma representação genérica de todas as transposições e inversões de um determinado conjunto de notas (Kostka, 2018) e essencial para a escrita da classe de um determinado CA. Para identificar a ordenação normal ideal parte-se da ordenação normal e escreve-se, de seguida, a sua inversão. Com esse objetivo basta manter a nota mais grave e aguda do conjunto e trocar a ordem dos intervalos presentes. Este processo pode ser visto na Figura 2.9, que retoma o exemplo da Figura 2.8, em que os intervalos entre notas em a. são 4 e 2, e em b., ou seja, na sua inversão, 2 e 4, mantendo iguais as notas mais grave e aguda do



**Figura 2.9:** Exemplo demonstrativo das transformações necessárias para se transformar a ordenação normal na ordenação normal ideal de um determinado conjunto. Em b. pode ler-se a ordenação normal ideal de a. depois da inversão dos seus intervalos (Kostka, 2018).

conjunto. Finalmente, a ordenação normal ideal é dada comparando o intervalo inicial da primeira hipótese com a sua inversão e, aquela em que ele for mais pequeno, é considerada a melhor opção. A opção b. é, portanto, a ordenação normal ideal deste conjunto e escreve-se [9, 11, 3] ou [A, B, Eb].

## II. A classe dos conjuntos de alturas

O conceito de classe consiste em transformar as notas da ordenação normal ideal em números que possam representar de forma abrangente todos os CA que a si pertencem, tal como a expressão 3ª M abrange todas as hipóteses concretas de 3ªs maiores, sendo dois exemplos dela o intervalo entre D e F# ou Bb e D (Kostka, 2018).

Para tal, o primeiro número de qualquer classe deve ser sempre 0, número esse que corresponde à nota mais grave da ordenação normal ideal e, seguidamente, os outros números dizem respeito à distância em número de meios tons das restantes notas do conjunto em relação à sua nota mais grave. No exemplo da Figura 2.9, constatou-se que a ordenação normal ideal é [A, B, Eb]. Como a nota B está dois meios tons acima de A, a sua nota mais grave, e Eb seis meios tons, o nome desta classe é [026] e representa todas as transposições e inversões que se podem obter a partir do CA analisado nos pontos I. e II. (Kostka, 2018).

### 2.2.4 Classes intervalares e vetor intervalar

Para se poder comparar o conteúdo intervalar de qualquer classe é útil o conceito de classe intervalar (CI). Ele consiste em atribuir um número inteiro entre 0 e 6 a todos os intervalos possíveis dentro de uma oitava, uma vez que também eles possuem inversões. Por outras palavras, a 7ªM é uma inversão da 2ªm e a 6ª M da 3ªm, assim sendo ambas podem ser representadas pela CI 1 e 3, respetivamente (Kostka, 2018). A Tabela 2.2 mostra em exatidão todas as seis CI.

Por último, é possível também identificar-se a noção de vetor intervalar, pois é ela que nos permite comparar os intervalos entre diferentes classes. Por essa mesma razão, o vetor intervalar é uma contagem das CI que se encontram num determinado conjunto. Para isso, identificam-se todos os intervalos nele presentes escrevendo-se de seguida entre parêntesis angulares o número

**Tabela 2.2:** Os números inteiros de CI associados aos seus respectivos intervalos (adaptado de [Kostka, 2018](#)).

Classe intervalar	Intervalo
1	2 <sup>a</sup> m, 7 <sup>a</sup> M
2	2 <sup>a</sup> M, 7 <sup>a</sup> m
3	3 <sup>a</sup> m, 6 <sup>a</sup> M
4	3 <sup>a</sup> M, 6 <sup>a</sup> m
5	4 <sup>a</sup> P, 5 <sup>a</sup> P
6	4 <sup>a</sup> A, 5 <sup>a</sup> d

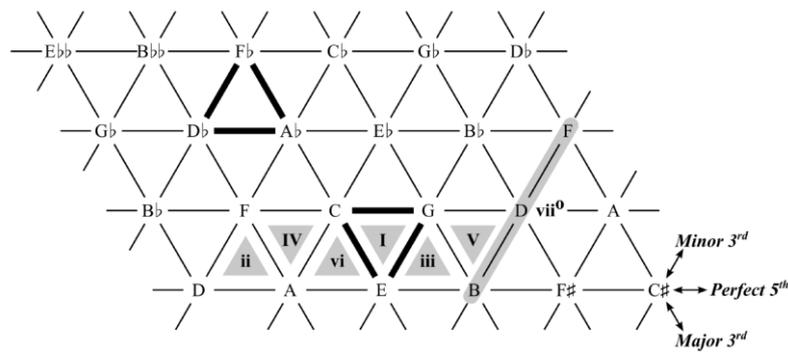
de ocorrências identificadas ([Kostka, 2018](#)). Assim, no exemplo da classe [026], abordado no ponto II. da subsecção 2.2.3, vemos que, entre A e B, a CI é 2, entre A e Eb é 6 e, por fim, entre B e Eb, a CI é 4. Esta classe tem, portanto, uma única ocorrência para as CI 2, 4 e 6 e, por isso, o seu vetor intervalar é [010101].

## 2.3 Espaços Geométricos e Fourier na Música

Yust inicia o seu artigo intitulado “Schubert’s Harmonic Language and Fourier Phase Space”, de 2015, com uma frase que considero de lapidar a propósito da ideia de EGRH e da sua utilidade para a análise musical e interpretação sintática na música. Diz Yust:

“The idea of harmonic space is a powerful one, not primarily because it makes visualization of harmonic objects possible but, more fundamentally, because it gives us access to a range of metaphors commonly used to explain and interpret harmony: distance, direction, position, paths, boundaries, regions, shape, and so on” ([Yust, 2015](#), p. 121).

Esta frase vinca a importância de EGRH, presentes na literatura desde há vários séculos, para a deteção de movimentos harmónicos padronizados e, no fundo, para o estudo sintático da música. Ora, esta ideia, a de espaços harmónicos, na qual assenta muita da motivação deste estudo, pode ser alcançada através da TDF para CA, tal como foi estudada por Lewin ([1959, 2001](#)), Quinn ([2006-7](#)), Callender ([2007](#)) e ainda Amiot ([2013](#)) e Sethares ([2011](#)). Nesta secção 2.3, pode ler-se sobre um dos primeiros e mais estudados espaços, o *Tonnetz*, bem como o contributo destes e outros autores para a criação de EGRH através da TDF. Por fim, tais espaços propiciam, também, a criação de ferramentas de apoio à composição e, sobre uma delas, o *PaperTonnetz*, pode ler-se em 2.3.6.



**Figura 2.10:** A estrutura intervalar do Tonnetz e formas geométricas em forma de triângulo associadas aos acordes da tonalidade de CM (Bernardes *et al.*, 2016).

### 2.3.1 Espaços geométricos de representação harmónica: o Tonnetz

O *Tonnetz* é um EGRH muito antigo atribuído ao matemático do séc. XVIII Leonhard Euler (Cohn, 1998) e que, por isso, por ser uma das primeiras abordagens ao problema, se torna importante a sua análise no contexto deste trabalho.

Este espaço é de extrema utilidade no que à música tonal diz respeito, porque a sua organização favorece intervalos consonantes como a 5ªP, a 3ªM ou 3ªm. Tal organização é conseguida através de uma geometria triangular em que a 5ªP se lê horizontalmente e as 3ªs na diagonal (Bernardes *et al.*, 2016).

A estrutura geométrica do *Tonnetz* pode ser vista na Figura 2.10, e lá é possível identificar as diferentes classes de altura e, de forma análoga, destacam-se acordes que aparecem sob a forma de figuras geométricas. Por exemplo, os acordes maiores assumem sempre a forma de um triângulo cujo vértice isolado aponta para baixo, enquanto que, nos acordes menores, o vértice isolado desse triângulo se direciona para cima. Curiosa é, também, a estrutura de três sons em linha referente a um acorde diminuto e que, em conjunto com as estruturas geométricas que foram referenciadas anteriormente, permite identificar numa mesma linha horizontal do *Tonnetz*, todas as funções harmónicas discutidas em 2.1.2 e referentes aos diferentes graus harmónicos de uma determinada tonalidade. No exemplo da Figura 2.10, temos o acorde de CM (I), que se encontra no centro do espaço, e, imediatamente a seguir, os acordes da relativa menor à esquerda, ou seja, Am (vi) e, à direita, GM (V). Nos lugares seguintes, vêem-se os restantes acordes que representam as funções harmónicas em falta.

O *Tonnetz*, devido a esta clareza no âmbito da representação das funções harmónicas tonais, foi amplamente estudado nos anos mais recentes por diversos autores e serviu de inspiração para uma abordagem mais matemática através da TDF, por exemplo, por Bernardes (2016) ou Yust (2015). A primeira das duas abordagens, intitulada espaço tonal intervalar, propõe um espaço de doze dimensões calculado através da TDF para CA a partir de um vetor intervalar de doze

elementos, cuja representação é feita através de seis círculos que cobrem todas as CI e, conseqüentemente, todos os intervalos musicais (Bernardes *et al.*, 2016). No que à abordagem de Yust (2015) diz respeito, mais próxima aos intentos desta dissertação, pode ler-se nas próximas subsecções uma análise mais detalhada.

### 2.3.2 A transformada discreta de Fourier para conjuntos de alturas

A TDF é conhecida tradicionalmente entre músicos, investigadores e especialistas como um processo matemático capaz de extrair informação musical no domínio das frequências, tendo como base ou *input* um sinal áudio. Este processo é largamente utilizado por uma multiplicidade de algoritmos cujo objetivo é extrair informação tímbrica, harmónica ou melódica através de um qualquer ficheiro áudio e baseia-se na decomposição do sinal em múltiplos componentes ou parciais, que, somados, formam o sinal original.

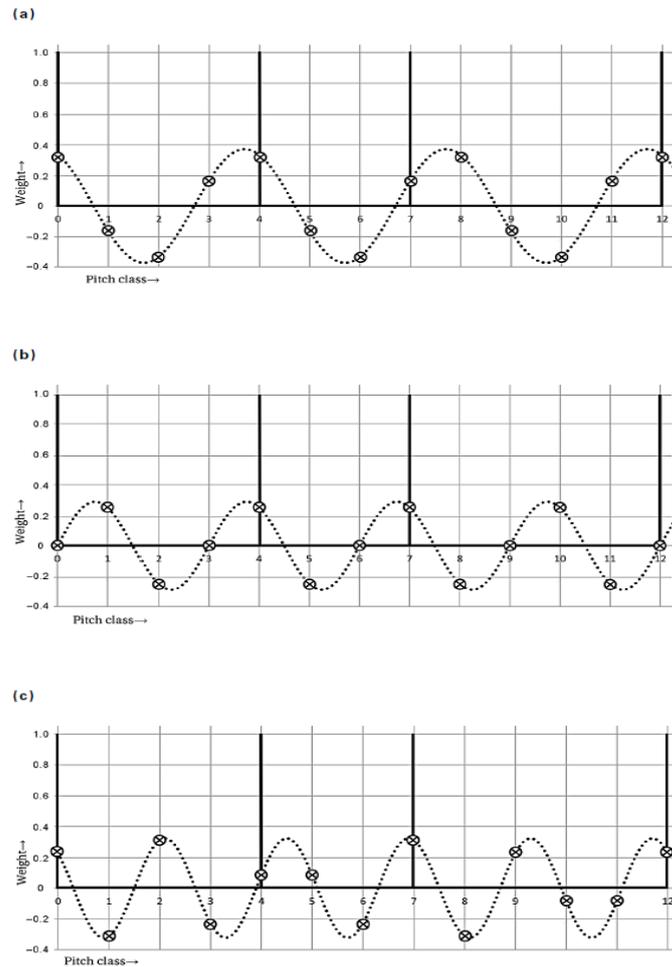
Este processo pode também ser utilizado para CA (Lewin, 1959, 2001) sendo que, do ponto de vista matemático, é explicado por Amiot (2016, 2017, 2018) e, como tal, a TDF para um CA  $A \subset \mathbb{Z}_n$  é dada, simplesmente, por:

$$\mathcal{F}_A = \widehat{1}_A : x \mapsto \sum_{k \in A} e^{-2i\pi kx/n}$$

em que  $\mathcal{F}_A$  é a TDF (sendo ela também um vetor contido em  $\mathbb{Z}_n$ );  $\widehat{1}_A$  é o vetor de classes de altura de um determinado CA e em que os valores  $\mathcal{F}_A(0) \dots \mathcal{F}_A(n-1) \in \mathbb{C}$  são chamados de componentes Fourier (Amiot, 2016, 2017, 2018).

No entanto, Lewin (1959, 2001) e Quinn (2006-7) teorizaram a utilização da TDF para CA, através de uma simples analogia. Nela, o eixo das coordenadas, tradicionalmente o tempo, é agora o conjunto de classes de altura de uma oitava. Desta forma, a todas as notas presentes numa determinada EH é dada amplitude ou altura 1 e, às restantes, amplitude 0, de tal forma que o sinal presente num contexto de TDF tradicional é agora substituído por um CA. Este novo sinal é, então, decomposto pela TDF em doze sinusoides e, cada uma delas, determinada por duas informações essenciais: magnitude e fase (Quinn, 2006-7). A Figura 2.11 mostra um exemplo deste processo.

Nela podemos ver linhas verticais a negrito como as classes de altura referentes ao acorde de CM (0, 4 e 7) com amplitude 1 e, conseqüentemente, a TDF decompõe esse sinal em doze sinusoides distintas, chamadas componentes de Fourier, que, quando somadas, voltam a formar o sinal original referente ao CA de CM. As componentes de Fourier são representadas pelos símbolos  $\hat{f}_0, \hat{f}_1, \dots, \hat{f}_{11}$ , nas quais o índice diz respeito ao número de ciclos por oitava da respetiva senoide. Também na Figura 2.11, as componentes mostradas têm, portanto, três, quatro e cinco picos por oitava, respetivamente.



**Figura 2.11:** As três maiores componentes Fourier associadas à tríade de CM em linhas tracejadas. Estas componentes assumem a forma de senoide e têm, respetivamente, 3, 4 e 5 ciclos por oitava. As linhas a negro representam as classes de altura das notas do acorde de CM (Yust, 2015).

### 2.3.3 Magnitude das componentes Fourier

Como referido anteriormente, cada uma das componentes da TDF para CA é uma senoide definida por um valor de magnitude e de fase. Explicando de forma genérica, a magnitude diz respeito à altura dessa senoide e para a componente Fourier  $\hat{f}_n$ , representa-se por  $|\hat{f}_n|$ . A sua interpretação do ponto de vista musical permite-nos verificar o quão estruturado é um determinado CA em torno de uma única CI. Por exemplo, quanto maior  $|\hat{f}_1|$ , maior será a preponderância da CI 1, ou seja, 2<sup>a</sup>s menores e 7<sup>a</sup>s maiores, na EH sujeita à TDF (Yust, 2015).

Com isto em mente, e recordando que estamos a analisar CI, a magnitude de cada componente é independente da transposição ou inversão do CA analisado, de tal forma que ela isola informação relacionada especificamente com a classe da EH em análise. Este aspeto esteve na base da motivação para o uso da TDF para CA por Lewin (1959, 2001) e Quinn (2006-7).

### 2.3.4 Fase das componentes Fourier

As fases assumem um valor e interpretação bastante distinto das magnitudes e o trabalho de Yust (2015) incide unicamente nelas. A fase de uma componente Fourier é o que permite mover a sua sinusoide para a esquerda ou direita, determinando, conseqüentemente, em que ponto é que ela interceta as diferentes classes de altura. De forma análoga às magnitudes, para a componente Fourier  $\hat{f}_n$ , a sua fase representa-se por  $\varphi_n$  e o seu valor é normalmente apresentado em radianos. Uma outra característica relevante das fases calculadas através da TDF é o seu carácter cíclico, ou seja, sendo estas medidas em radianos (variam entre  $-\pi$  e  $\pi$  ou em graus entre  $-180^\circ$  e  $180^\circ$ ), faz com que duas informações de fase com valor  $-\pi$  e  $\pi$  sejam coincidentes num espaço de fases, porque elas representam exatamente o mesmo ângulo de movimentação mas medido em sentidos contrários.

Por outro lado, ao contrário das magnitudes, as fases são dependentes da transposição ou da inversão da EH analisada, porque, de forma genérica, nos permitem identificar o posicionamento de um determinado CA através da sua distância relativa ao posicionamento de todas as classes de altura individuais que o constituem (Yust, 2015). As fases das componentes Fourier estão, por isso, na base do trabalho de Yust (2015) para a criação de um EGRH, que melhor se descrevem em 2.3.5.

### 2.3.5 O espaço de fases Fourier

O espaço de fases Fourier é o nome dado por Yust (2015) ao EGRH que utiliza somente as fases calculadas de duas componentes Fourier. Nele, o conceito de distância entre CA é obtido colocando nos referenciais de um gráfico simples de duas dimensões as fases de duas das componentes da TDF (Yust, 2015). Tal espaço bidimensional é uma garantia de facilidade na análise e na interpretação dos resultados. Esta abordagem é muito semelhante à que Quinn (2006-7) chama de “quality space”, mas, nela, são antes utilizadas as magnitudes como coordenadas do referencial.

No espaço de fases Fourier, Yust (2015) opta por utilizar as fases das componentes  $\hat{f}_3$  e  $\hat{f}_5$ , e, portanto, o autor chama a tal espaço  $\varphi_{3/5}$ . Estas componentes em específico são de especial relevância para o seu objeto de estudo, porque são as duas componentes com maior valor de magnitude para tríades consonantes.

Na Figura 2.12, podemos ver a proximidade entre os acordes CM e Am, bem como classes de altura individuais para efeitos de orientação. Em boa verdade, esta característica é fundamental num espaço de fases, porque o posicionamento de um CA é dado pelo posicionamento médio de todas as classes de altura individuais que o constituem (Yust, 2015) e, como tal, o acorde de CM situa-se entre o posicionamento das classes de altura 0, 4 e 7, ou seja, as notas C, E e G. O eixo das coordenadas e das abcissas denota incrementos de  $\pi/6$  pois é o valor correspondente a meio tom ( $1/2t$ ). Com este espaço é possível estabelecer o trajeto harmónico numa peça e detetar

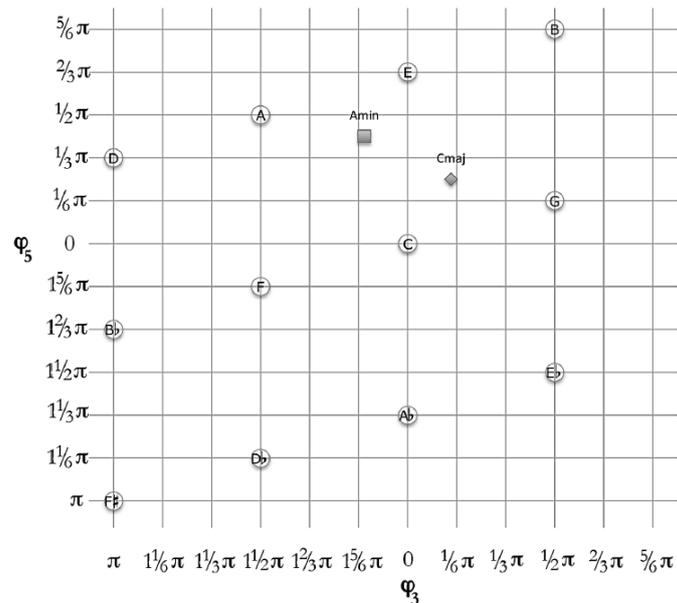


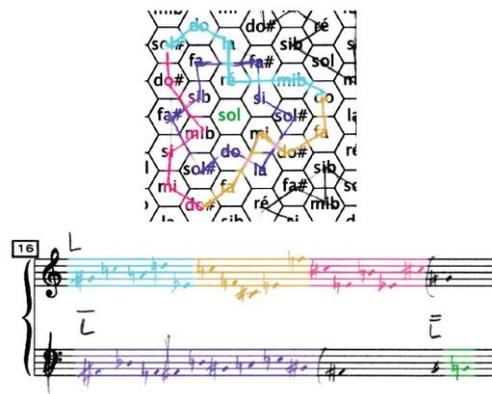
Figura 2.12: Os acordes de CM e Am representados no espaço de fases  $\varphi_{3/5}$  (Yust, 2015).

movimentos padronizados, ao longo do tempo. De resto, o espaço de fases que agora se descreve, sobretudo devido ao seu carácter cíclico e à preponderância do posicionamento de classes de altura individuais, apresenta uma série de pontos de contacto com outros espaços, necessariamente distintos, mas que também são utilizados na análise de obras referentes ao séc. XX. Um deles, proposto por Martins (2006), intitulado “dasian space”, tem o objetivo de estudar a prática composicional de combinar estratos de diferentes escalas diatónicas numa mesma obra, como é típico de compositores como Bartók ou Stravinsky. Este espaço é constituído por quarenta e oito classes de altura organizadas de forma cíclica, de tal forma que qualquer pentacorde escolhido dentro do espaço terá sempre uma de quatro configurações intervalares distintas, sendo elas: (1)  $1t, 1t, 1t, 1/2t$ ; (2)  $1t, 1t, 1/2t, 1t$ ; (3)  $1t, 1/2t, 1t, 1t$  e (4)  $1/2t, 1t, 1t, 1t$  (Martins, 2006).

### 2.3.6 Aplicações de composição assistida: o *PaperTonnetz*

O *PaperTonnetz* é uma ferramenta de apoio à composição musical inspirada no EGRH, discutido em 2.3.1. A aplicação permite aos utilizadores comporem, improvisarem e explorarem diferentes sonoridades, gestos musicais ou acordes através de uma representação do espaço *Tonnetz* desenhado em papel interativo (Bigo et al., 2013).

A premissa para esta aplicação surge da obra *Traversée*, de Chouvel, cujo processo composicional oferece um excelente ponto de partida para a forma como o espaço de *Tonnetz* pode ser vantajoso no ato da escrita. Este compositor utilizou esta ferramenta para desenhar um conjunto de linhas geométricas, linhas essas que representam diferentes sequências de notas. A



**Figura 2.13:** O trabalho de Chauvel em papel baseado no *Tonnetz* para a sua peça *Traversée* e que motivou o desenvolvimento do PaperTonnetz. Imagem de Jean:Marc Chauvel.

essas linhas, Chauvel chamou “constelações” (Bigo *et al.*, 2013). A Figura 2.13 mostra como essas constelações foram desenhadas e trabalhadas musicalmente pelo compositor. Para ele, a principal vantagem da exploração deste espaço é a coerência harmônica intrínseca à utilização de figuras ou linhas semelhantes em diferentes regiões do espaço.

Com este processo composicional em vista, o *PaperTonnetz* foi desenvolvido, em 2012, e permite aos seus utilizadores desenhar em papel interativo do espaço *Tonnetz* várias linhas ou formas geométricas, ouvir em tempo real o resultado auditivo desses desenhos, quer em forma de acorde ou de notas individuais, e guardar as suas escolhas preferidas.

## 2.4 Símula

Ao longo deste capítulo, reviu-se o que diz a literatura a propósito de três áreas essenciais ao desenvolvimento deste estudo, são elas a sintaxe musical, a teoria musical pós-tonal e os EGRH através da TDF.

Em relação à primeira, constatam-se um conjunto de diferenças entre sintaxe musical e linguística muito relevantes, mas, simultaneamente, discernem-se nelas semelhanças hierárquicas e lógicas. Entre elas destaca-se a presença de funções gramaticais numa e harmónicas noutra, que guiam o processo de elaboração de frases e sequências harmónicas, respetivamente. No entanto, a averiguação de tais comparações só será profícua se se evitar a busca forçada por estruturas musicais que mimetizem a construção sintática da linguagem. Para além disto, também se constata que as escalas e os acordes são duas formas essenciais de organização sintática musical. Por outras palavras, ambos têm a si associadas um conjunto de regras próprias de construção que nos permitem entender melhor como a sintaxe musical, para além de algumas semelhanças, tem características estruturais muito distintas da sintaxe linguística.

Em segundo lugar, ao longo da secção 2.2, percebem-se alguns dos aspetos mais relevantes no que à teoria musical do período pós-tonal diz respeito. Entre eles destaca-se o princípio da equivalência de oitava e como esse aspeto é relevante para a definição dos conceitos seguintes, como a noção de classes de altura – definidas por números inteiros entre 0 e 11 – ou até de CA, como grupos restritos de notas e dispostos numa única oitava. Outra ideia da maior importância abordada é a de classe, bem como todos os processos necessários para a determinar de forma exata. Esta estrutura é essencial na compreensão da música atonal, pois condensa num só elemento uma multiplicidade de CA formados através de transposições e inversões.

Por último, é possível ler em 2.3 o que se identifica na literatura a propósito de EGRP, nomeadamente o de *Tonnetz* e o de Yust (2015). Ambos são ferramentas poderosas do ponto de vista da análise harmónica e facilitam o estudo sintático da música. Em particular, o espaço de Yust (2015) intitulado espaço de fases Fourier é determinado através do cálculo da TDF para CA e das suas respetivas fases.

Este processo da TDF, conseguido através da decomposição de uma qualquer EH em doze sinusoides distintas, assim como o cálculo e interpretação das fases e magnitudes das suas componentes, é de vital importância para a identificação de um espaço geométrico útil ao propósito deste trabalho. Tais espaços, para além de facilitarem a deteção de movimentos harmónicos padronizados, são também importantes como ferramenta de apoio à composição pois permitem ao seu utilizador pensar harmonicamente de uma forma mais prática e intuitiva, não perdendo com isso qualquer tipo de coerência harmónica. O *PaperTonnetz* é um exemplo exato desta ideia.

Em suma, todos estes conceitos e ferramentas são de vital importância para a resolução do problema identificado neste estudo. Isto acontece porque, por um lado, percebe-se melhor, neste momento, o que é a sintaxe musical e os seus diferentes níveis de funcionamento, como também a dificuldade do seu entendimento na música atonal. Assim sendo, os conceitos teóricos de música pós-tonal, e, sobretudo, a criação de EGRH através da TDF para CA, são essenciais para a visualização das diferentes EH de uma peça no espaço e para a deteção de movimentos padronizados entre os seus CA, ou seja, exatamente o que se procura.



## 3. A Esfera de Fases Fourier

Este capítulo diz respeito à idealização e implementação de um sistema computacional, ao qual se chamou EFF, que, em última instância, cria um espaço esférico de representação harmónica derivado da informação de fase da TDF para CA. No espaço resultante podem-se, entre outros, agrupar diferentes EH e identificar as trajetórias harmónicas entre elas. Para esse efeito, o capítulo encontra-se estruturado em sete secções distintas: inicialmente, na secção 3.1, é feita uma descrição global do sistema, a partir da sua arquitetura modular; de seguida, da Secção 3.2 à 3.7, apresentam-se de forma detalhada cada um dos seus módulos, descrevendo as suas especificidades e objetivos, sendo elas o processo de segmentação harmónica de uma partitura (§3.2), a criação de todas as estruturas de dados necessárias para os módulos seguintes (§3.3), o cálculo de fases através da TDF (§3.4), o cálculo do espaço esférico de representação harmónica proposto neste estudo (§3.5), o método de agrupamento dos vários CA (§3.6) e a descrição das ferramentas de deteção de padrões desenvolvidas (§3.7). Por fim, pode ler-se, na secção 3.8, uma breve recapitulação dos principais pontos abordados ao longo deste capítulo. Nas secções referentes aos módulos do sistema (§3.2 a §3.7), é possível ler-se sempre no seu último parágrafo uma remissão para o anexo onde se encontram escritas as funções em *Python* ([2020](#)) importantes para a compreensão do módulo.

### 3.1 A Arquitetura da EFF

Para se poder ter uma noção clara sobre a idealização, mas também sobre o desenvolvimento e funcionamento do sistema proposto ao longo de todo este capítulo, é de extrema utilidade atentar, inicialmente, na sua arquitetura, pois ela permite obter uma visão global da aplicação e dos diferentes módulos necessários ao seu funcionamento, bem como, num segundo nível, entender o fluxo de informação partilhada entre os módulos. Por essa razão, podem ler-se em §3.1.1 os objetivos da EFF e a forma como ela foi idealizada e implementada. Seguidamente, é descrita a sua arquitetura modular.

### 3.1.1 Objetivos e idealização

Esta dissertação debruça-se sobre o estudo da sintaxe harmónica na música referente ao período pós-tonal do início do séc. XX. Pretende, portanto, procurar regras sintáticas que regulem como as EH se sucedem no tempo e averiguar a existência ou inexistência de padrões nas movimentações entre diferentes grupos de CA. Para além disto, existem dois outros objetivos de relevo que importa destacar. O primeiro diz respeito à utilização do valor das fases calculado através da TDF para a disposição de CA num espaço onde seja possível comparar o seu posicionamento. O segundo corresponde à avaliação da utilidade dos processos descritos anteriormente para o apoio à composição musical, através do estabelecimento de uma sintaxe harmónica no processo criativo do compositor.

Tais objetivos, idealmente, exigem um processo contínuo e algorítmico cujo produto final é uma aplicação multimédia que: (1) permita a leitura automática de partituras, de forma a ser possível isolar todas as suas EH distintas; (2) possibilite o cálculo da TDF para cada uma delas e das suas respetivas fases; (3) disponha tais EH num EGRH onde seja possível visualizar o seu posicionamento e trajetórias e (4) apresente algumas ferramentas para a identificação de padrões e movimentos dentro desse espaço. No ponto seguinte, é possível ver-se a arquitetura da EFF implementada para a concretização destes objetivos.

### 3.1.2 Concretização

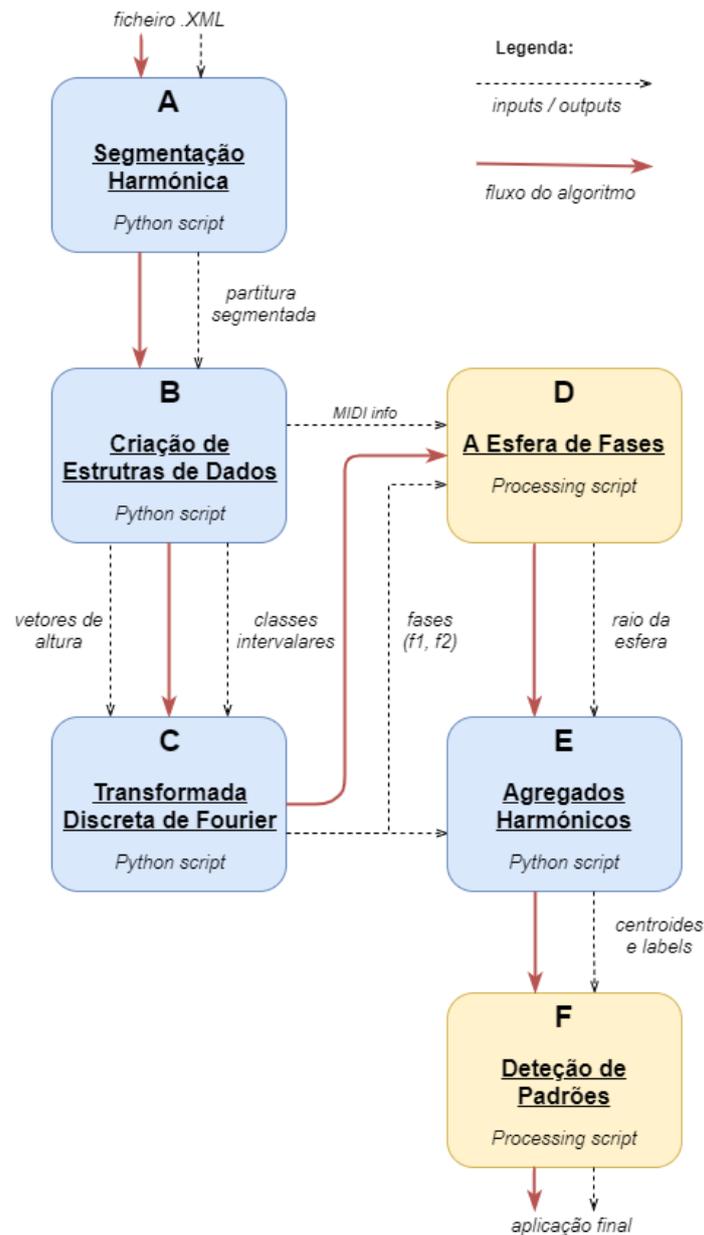
A implementação da representação gráfica idealizada foi conseguida através de seis módulos distintos que se podem ver na Figura 3.1 e que, nas secções seguintes, serão detalhadamente descritos. Estes módulos permitem entender o fluxo de funcionamento da EFF e, genericamente, realizam as seguintes operações:

- A. Segmentação harmónica – recebe como entrada um ficheiro em formato *.XML*<sup>5</sup> (ficheiro esse referente à partitura que se pretende dispor no espaço de fases) e, de seguida, de forma automática, reduz todas as EH de uma partitura complexa para uma única pauta representativa de todas as classes de altura de um determinado momento da peça;
- B. Estruturas de dados – este módulo, dada a partitura segmentada, elabora um conjunto de transformações cujo objetivo é criar todas as estruturas de dados necessárias ao resto do funcionamento da aplicação, tais como a contagem de classes de altura em cada EH e as suas notas MIDI<sup>6</sup> ou a contagem de CI;

---

<sup>5</sup> O formato *.XML* foi desenvolvido para permitir a codificação de todo o tipo de documentos de forma a que eles sejam, simultaneamente, legíveis por humanos ou computadores. Este formato tem aumentado a sua importância, nos últimos anos, na troca de informação digital na Web ([Extensible Markup Language, 2020](#)).

<sup>6</sup> O protocolo MIDI permite, sucintamente, a comunicação em tempo real entre instrumentos musicais eletrónicos e computadores. Uma das suas características é a possibilidade de identificar um total de 128 notas cromáticas distintas (valores de 0 a 127 em que representa o dó central), o que excede o âmbito tradicional do piano e é suficiente para a descrição de informação musical tradicional ([Midi Communication Protocol, 2020](#)).



**Figura 3.1:** A macroestrutura da EFF. Ela contém seis módulos: segmentação harmónica, criação de estruturas de dados, TDF, a esfera de fases, agregados harmónicos e deteção de padrões.

- C. Transformada discreta de Fourier – uma vez criada a informação da partitura, numa estrutura de dados adequada, é calculada a TDF, eliminadas as suas componentes desnecessárias e calculadas as fases para as restantes. Por fim, a informação de fase referente às duas componentes de maior relevo (obtidas através da contagem de CI) é armazenada numa nova lista e exportada para cálculos posteriores e para ser representada na esfera de fases;
- D. A esfera de fases – os dados exportados no módulo C referentes aos valores de fase, mas também as notas MIDI calculadas em B, são neste módulo utilizadas para criar a

representação esférica da aplicação. Nela é desenhada a esfera cuja superfície representa todo o espaço harmónico possível, estabelecido o valor do seu raio e apresentados todos os pontos referentes às EH da peça, assim como a sua reprodução áudio;

- E. Agregados harmónicos – através do raio da esfera definido em  $D$ , é feito o agrupamento desses pontos. O número de grupos e a sua correspondência com cada uma das EH são conseguidos através de um algoritmo de *clustering*<sup>7</sup> e, seguidamente, são exportadas as coordenadas no espaço referentes ao centro de cada agregado, assim como a informação que associa cada ponto ao seu respetivo grupo;
- F. Detecção de padrões – recebendo a informação dos AH, é agora possível, no espaço esférico de fases definido em  $D$ , distinguirem-se os diferentes agregados de CA, identificarem-se as trajetórias entre EH e visualizarem-se três ferramentas distintas de deteção de padrões criadas para o efeito deste trabalho.

Todos os módulos agora descritos são operacionalizados através quer da escrita de código em linguagem *Python* (2020) quer de um conjunto de outras ferramentas explicadas com maior rigor de seguida, como a biblioteca *Music21* (2020) (destinada a promover trabalho em musicologia computacional), o algoritmo de aprendizagem de máquina *k-means clustering* (2020), que possibilita a criação de diferentes grupos de CA, e o *software Processing* (2020), onde escrevo todo o código referente à representação multimédia final da aplicação.

Outro ponto a assinalar é o facto do sistema da Figura 3.1 ter sido progressivamente pensado e melhorado. Por exemplo, inicialmente fizeram-se representações num espaço de fases bidimensional, em tudo semelhante ao proposto por Yust (2015), e mencionado em 2.4.5, mas o objetivo a que este trabalho se propõe, de analisar diferentes grupos de EH, obrigou à criação de um espaço de fases esférico para garantir um agrupamento mais correto. Isto acontece devido à propriedade de circularidade das fases – explicada em maior detalhe na secção 3.6 –, uma vez que estas são calculadas em radianos. Esta necessidade esteve, por isso, na origem de uma das principais contribuições deste estudo.

## 3.2 Segmentação Harmónica

Uma das necessidades essenciais para este estudo é a possibilidade de extrair informação harmónica de uma qualquer peça. O método utilizado para esse fim inicia-se com a criação de uma partitura em formato *.XML*. Este ficheiro é amplamente utilizado para armazenar informação musical e qualquer *software* de notação atual possibilita tanto a leitura como a exportação de partituras neste formato (no caso dos resultados abordados no Capítulo 4, forçado pelo facto de não existir nenhuma coleção de partituras disponível online referentes a música atonal, foi

---

<sup>7</sup> Os algoritmos de *clustering*, como o algoritmo *k-means* (2020) permite agrupar qualquer tipo de informação em diferentes grupos, estudando, automaticamente, a melhor hipótese de o fazer para o *dataset* em análise ([k-means, 2020](#)).



**Tabela 3.1:** O vetor de classes de altura ( $v$ ) para um acorde de CM (Bernardes *et al.*, 2016).

Vetor de classes de altura ( $v$ )												
Posição ( $n$ )	0	1	2	3	4	5	6	7	8	9	10	11
Classe de altura	C	C#	D	D#	E	F	F#	G	G#	A	A#	B
Valor do vetor	1	0	0	0	1	0	0	1	0	0	0	0

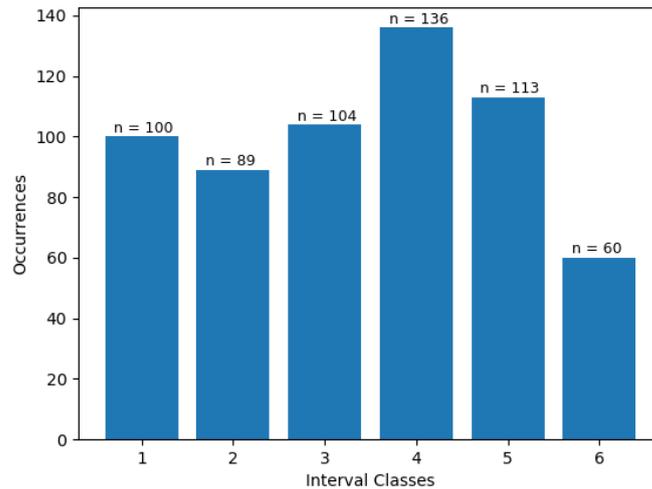
### 3.3 Estruturas de Dados

Depois da partitura ser segmentada é agora necessário trabalhar a informação de cada CA da peça para que ela possa ser compreendida nos módulos seguintes. Assim sendo, é crucial conseguir-se: (1) transformar cada EH num vetor de classes de altura; (2) fazer uma contagem de todas as CI e (3) exportar uma lista de CA com notas em valores MIDI para que esta possa ser importada num *software* diferente. Todas estas transformações, à semelhança do módulo anterior, são conseguidas através de processos algorítmicos escritos em *Python* (2020).

No primeiro caso, pretende-se criar uma estrutura de dados que contenha um vetor de classes de altura para cada EH da peça. Esse vetor caracteriza-se pela existência de uns ou zeros, para cada classe de altura, consoante ela esteja ou não presente na EH em análise, resultando, assim, num conjunto de doze números consecutivos. Para melhor entendimento deste vetor de classes de altura, a Tabela 3.1 mostra um exemplo para o acorde de CM. Esta estrutura de dados é criada recorrendo ao método da biblioteca *Music21* (2020) com o nome `normalOrder`, que cria uma lista com as classes de altura presentes numa dada EH – no exemplo de CM, essa lista é  $[0, 4, 7]$ , que é, seguidamente, transformada algoritmicamente num vetor de doze dígitos (mais concretamente  $v = [1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0]$ ). A lista final enviada para cálculos subsequentes tem todos os vetores de classes de altura da peça, tendo, portanto, a seguinte tipologia, em que  $v$  representa o vetor e  $n$  o número total de EH na peça:

$$pitch\_class\_vector\_list = [[v1], [v2], [v3], \dots, [vn]]$$

Outra informação de extrema importância para a representação final é a contagem de CI na peça, porque o EGRH que se pretende para este estudo confronta sempre a informação de duas fases calculadas através da TDF. Ora, no caso proposto por Yust (2015), as fases utilizadas são as referentes às componentes  $\hat{f}_3$  e  $\hat{f}_5$  da transformada e, por isso, o autor intitula o seu espaço de  $\varphi_{3/5}$ . A escolha das fases referentes a estas componentes em concreto é feita pois as CI 3 e 5



**Figura 3.3:** A contagem do total de CI obtidas para o primeiro andamento do Op. 19, de Arnold Schönberg, intitulado *Sechs Kleine Klavierstück*.

(associadas a intervalos como a 3<sup>a</sup>m, 6<sup>a</sup>M, 4<sup>a</sup>P e 5<sup>a</sup>P) são relevantes para o objeto de estudo de Yust (2015) – a música de Schubert – e que assenta em EH típicas do tonalismo muito marcadas por tríades à base de 3<sup>as</sup> e 5<sup>as</sup>. No entanto, no âmbito da música do período pós-tonal, cada peça contém EH distintas o que levanta ao problema de o espaço  $\varphi_{3/5}$  não ser o mais adequado para a sua representação. Desse ponto de vista, a abordagem agora escolhida permite que o espaço final seja sempre referente às fases associadas às componentes da TDF, que dizem respeito às duas CI mais repetidas numa peça. Simplificadamente: se a peça tem 2 e 6 como as duas CI mais repetidas, então, o espaço de representação escolhido é  $\varphi_{2/6}$  ou, ao invés, se elas são a 1 e 3 o espaço passa a ser  $\varphi_{1/3}$ , garantindo com isso maior flexibilidade nos resultados obtidos, flexibilidade essa exigida pela extrema liberdade e variedade harmónica na música atonal.

Esta abordagem contrasta com a hipótese levantada por Yust (2016) que propõe a utilização das duas componentes da TDF com maior valor de magnitude para o cálculo das fases e respetiva representação num espaço geométrico. No entanto, esta hipótese requer a seleção de uma EH que se considere representativa da obra para o cálculo da sua TDF e a determinação do espaço de fases. Tal abordagem ficaria sempre dependente da capacidade analítica do utilizador da EFF para a seleção dessa sonoridade e, sendo focada numa única EH, não daria completa representatividade da totalidade da obra.

Para este efeito, o sistema implementado permite a contagem das CI de cada EH da peça através do método `intervalVector` da biblioteca *Music21* (2020) e, depois de feita a soma de todas essas classes, é criado um gráfico de barras recorrendo a uma outra biblioteca intitulada *matplotlib* (que em ambiente *Python* (2020) permite a criação de múltiplas representações gráficas) para contabilizar o total de ocorrências de cada CI na peça. A Figura 3.3 mostra um exemplo de um desses gráficos de barras em que as CI escolhidas para a determinação do espaço de fases são as 4 e 5 (as mais repetidas na peça), resultando, desse modo, num espaço  $\varphi_{4/5}$ .

**Tabela 3.2:** O conjunto de notas MIDI que representa a tríade de CM.

Notas MIDI para CM			
Nota ( $n$ )	1	2	3
Valor MIDI	60	64	67

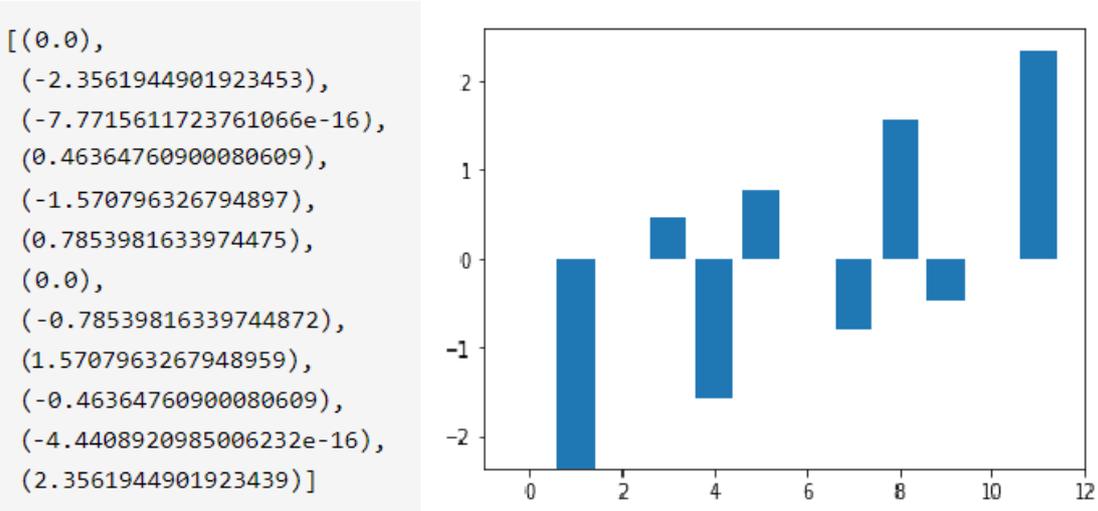
Por fim, é também necessário criar uma lista que contenha todas as notas MIDI de cada EH analisada na peça para que esta informação possa ser lida no *software* responsável pela representação harmónica e, lá, cada ponto colocado no espaço seja acompanhado com a sua respetiva reprodução de áudio. Este aspeto é bastante relevante pois permite ao utilizador da EFF confrontar a sonoridade de todas as EH da obra, com o seu posicionamento no espaço e daí extrair conclusões quanto ao trajeto harmónico da obra. Um exemplo do conjunto de notas MIDI necessárias para constituir o acorde de CM pode ser visto na Tabela 3.2. Como tal, recorrendo a um novo método de *Music21* (2020), desta vez com o nome `midi`, é possível transformar, primeiramente, a informação de cada CA da partitura segmentada em classes de altura e, de seguida, transformar essas classes de altura em notação MIDI, e armazená-la numa lista pronta a exportar do tipo seguinte, em que EH simboliza a estrutura harmónica guardada e  $n$  o número das suas notas:

$$midi\_list = [[EH1_1, \dots, EH1_n], [EH2_1, \dots, EH2_n], \dots, [EHn_1, \dots, EHN_n]].$$

Todo o código escrito em *Python* (2020) para este módulo pode ser consultado no [Anexo 1](#) e nos *scripts* intitulados [phaseSpace\\_main.py](#) e [phaseSpace\\_utilities.py](#), mais concretamente nas funções: (1) `pitch_classes_creator` e `pitch_classes_vector_creator`, no que à criação do vetor de alturas diz respeito; (2) `interval_class_counter`, para a contagem do total de CI na peça, (3) `midi_list_creator` e `midi_list_exporter`, para criar a estrutura de dados com notas MIDI e exportá-la para outro *software*.

### 3.4 Transformada Discreta de Fourier

Uma vez criada a estrutura de dados com todos os vetores de classes de altura representativos de cada EH na partitura, e que foi discutida em 3.3, torna-se agora possível calcular a TDF para cada um deles. Este cálculo, feito de forma simples e automática recorrendo ao método `fft` da



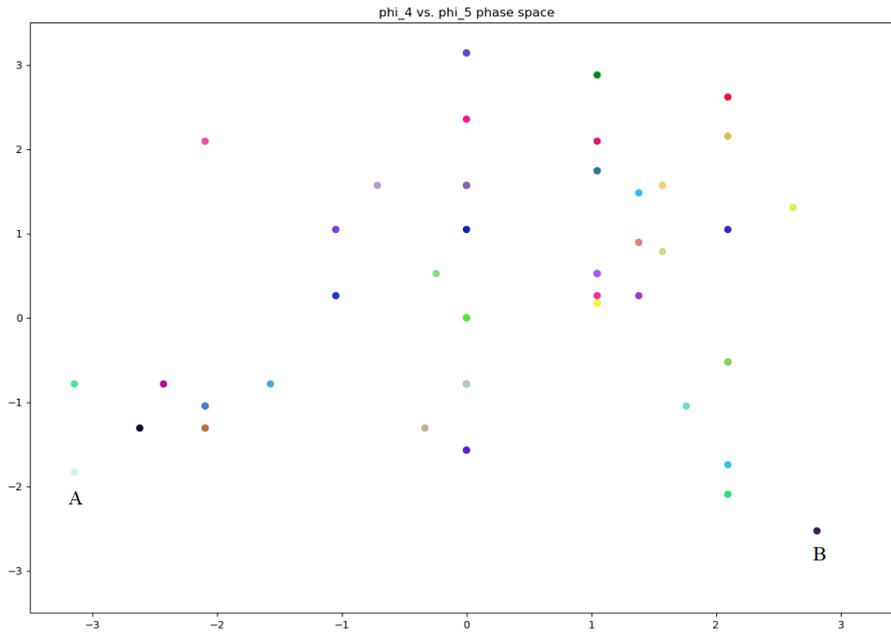
**Figura 3.4:** À esquerda cálculo das fases referentes às doze componentes da TDF para um acorde de CM e, à direita, a sua representação gráfica em que o eixo das abcissas ( $x$ ) representa a componente em análise ( $\hat{f}_n$ ) e o das ordenadas ( $y$ ) o valor de fase.

biblioteca para *Python* (2020), intitulada *scipy*, retorna doze componentes distintas ( $\hat{f}_0, \hat{f}_1, \dots, \hat{f}_{11}$ ) para cada vetor de classes de altura caracterizadas por doze números complexos, que sumarizam a informação de magnitude e fase que pode agora ser calculada.

Dado que para este estudo apenas interessa a informação de fase, ela pode ser facilmente calculada através do método `angle` da biblioteca *numpy* que, desta feita, retorna doze valores de fase por EH. Estes valores são obtidos em radianos, ou seja, variam entre  $-\pi$  e  $\pi$ . A Figura 3.4 mostra um exemplo dos valores de fase para um acorde de CM. No entanto, nem toda esta informação é relevante para a representação gráfica e pode em parte ser descartada. Isto acontece uma vez que, à semelhança das CI, em que uma mesma classe representa um intervalo e o seu simétrico (por exemplo, a CI 1 está associada a intervalos de 2<sup>a</sup>m, mas também de 7<sup>a</sup>m), também a informação de fase tem um comportamento semelhante. Logo, à partida, a fase da primeira componente ( $\hat{f}_0$ ) tem sempre valor 0 (a sua magnitude diz respeito à cardinalidade<sup>8</sup> do CA em análise) e pode, portanto, ser descartada, mas, seguidamente, as fases das componentes  $\hat{f}_{11}, \hat{f}_{10}, \hat{f}_9, \hat{f}_8$  e  $\hat{f}_7$  também o podem ser, pois elas apresentam valores simétricos às fases das componentes  $\hat{f}_1, \hat{f}_2, \hat{f}_3, \hat{f}_4$  e  $\hat{f}_5$ , respetivamente. A Figura 3.4 mostra de forma mais evidente esta mesma simetria. Como tal, foi necessário, depois de calculadas as fases, eliminar a informação irrelevante, passando agora a ter, para cada EH, um conjunto de seis valores de fase, correspondentes às componentes da TDF  $\hat{f}_1, \hat{f}_2, \hat{f}_3, \hat{f}_4, \hat{f}_5$  e  $\hat{f}_6$ .

Depois de eliminada a informação de fases desnecessária, é agora possível criar uma estrutura de dados que conta somente com as duas fases importantes para a representação. Aqui

<sup>8</sup> A cardinalidade de um CA trata-se do somatório da informação binária presente no vetor de classes de altura. Por exemplo, esse vetor para o acorde de CM é  $[1,0,0,0,1,0,0,1,0,0,0,0]$  e, por essa razão, a sua cardinalidade é três (Amiot, 2018).



**Figura 3.5:** A representação gráfica de todas as EH do terceiro andamento da peça *Sechs Kleine Klavierstück*, de A. Schönberg, no espaço de fases  $\varphi_{4/5}$ . É perceptível que neste espaço o ponto A e B estão em extremos opostos do referencial, mas, devido à propriedade cíclica das fases, eles estão altamente próximos no espaço.

é essencial a informação obtida pelo somatório de CI da peça abordado em 3.3, ou seja, se as duas principais CI contabilizadas forem, por exemplo, a 1 e a 6, então, para cada EH da peça, será somente guardada e exportada para outro *software* a informação de fase correspondente às componentes  $\hat{f}_1$  e  $\hat{f}_6$  (para ser disposta num espaço tridimensional esférico detalhadamente descrito na secção 3.5), resultando, portanto, numa lista com a seguinte tipologia, em que EH representa a estrutura harmónica a ser guardada e  $n$  o número total de EH na peça:

$$phase\_list = [[EH1_{phase_{\hat{f}_1}}, EH1_{phase_{\hat{f}_6}}], \dots, [EHn_{phase_{\hat{f}_1}}, EHn_{phase_{\hat{f}_6}}]]$$

A abordagem de Yust (2015), neste ponto, dispõe a informação de fases anteriormente descrita num espaço bidimensional em que cada eixo é representativo do valor de fase de uma dada componente da TDF. Assim sendo, inicialmente, tentou-se uma abordagem semelhante que pode ser vista na Figura 3.5, em que cada CA pode ser posicionado como um ponto num espaço geométrico de fácil interpretação bidimensional. No entanto, sendo um dos objetivos deste trabalho conseguir dispor uma multiplicidade de EH no espaço e, depois, ser capaz de agrupar tais pontos em grupos distintos, percebeu-se que esta representação não era a mais indicada, pois, na verdade, ela é uma simplificação da informação de fase. Por outras palavras, sendo ela medida em radianos (varia entre  $-\pi$  e  $\pi$  ou em graus entre  $-180^\circ$  e  $180^\circ$ ), há um carácter cíclico e circular na sua essência que um espaço plano e bidimensional não consegue expressar. Por exemplo, um ponto cuja fase seja  $-\pi$  encontra-se, em verdade, sobreposto com um outro ponto com fase  $\pi$ ,

porque devido a esta propriedade cíclica das fases os extremos são coincidentes. No entanto, a representação bidimensional não nos garante esta clareza, como pode ser visto na Figura 3.5, o que pode levar a resultados pouco fiáveis na distinção dos CA em diferentes grupos. Por conseguinte, foi necessário desenvolver uma nova representação gráfica que pudesse dispor os valores de fase num espaço também simples de ser entendido, mas que pudesse melhor transmitir a propriedade da circularidade. Essa nova representação gráfica explica-se na secção seguinte.

Todo o código escrito em *Python* (2020) para este módulo pode ser consultado no [Anexo 1](#) e nos *scripts* intitulados [phase\\_space\\_main.py](#) e [phase\\_space\\_utilities.py](#), mais concretamente nas funções `fft_calculator`, `phases_calculator`, `phases_exporter`, `color_definition`, `window_size`, `plot_in_all_phase_spaces`, `plot_just_one_phase_space` e `show_graphs`.

### 3.5 A Esfera de Fases

A solução encontrada no âmbito deste trabalho para fazer face ao problema anteriormente descrito tira proveito do facto da informação de fase ser medida em radianos ou em graus e, dessa forma, é possível criar uma abordagem, que transforma os dois valores de fase obtidos para representar cada EH (ou seja, uma lista em tudo semelhante à última descrita no módulo anterior) em três coordenadas cartesianas<sup>9</sup>, que dispõe todos as EH na superfície de uma esfera e criando, desta feita, uma nova lista com a seguinte tipologia, em que EH representa a estrutura harmónica a ser guardada e  $n$  o número total de EH na peça:

$$cartesian\_list = [[EH1_{x_1}, EH1_{y_1}, EH1_{z_1}], \dots [EHn_{x_n}, EHN_{y_n}, EHN_{z_n}]]$$

Esta transformação é possível recorrendo aos conceitos geográficos de latitude e longitude. A latitude é, no fundo, a distância à linha do equador medida ao longo do meridiano de *Greenwich* (indica, portanto, quão a norte ou a sul um ponto se encontra e varia entre  $-90^\circ$  e  $90^\circ$ ) e a longitude mede a distância à linha do meridiano de *Greenwich* medida ao longo da linha do equador (indicando quão a este ou oeste um ponto está e varia entre  $-180^\circ$  e  $180^\circ$ ). Desta feita, com os valores de latitude e longitude, é possível identificar o posicionamento de qualquer ponto à superfície do globo terrestre e, também eles, tal como as fases, podem ser medidos em graus ou radianos, possibilitando, assim, a disposição de um CA com dois valores de fase num espaço tridimensional na superfície de uma esfera.

---

<sup>9</sup> As coordenadas cartesianas permitem que se localize um qualquer ponto num espaço plano de duas dimensões, ou num espaço tridimensional. Num gráfico cartesiano, o ponto de intersecção entre os seus referenciais é chamado de origem. Tais coordenadas são do tipo  $(x,y)$  para informação bidimensional e do tipo  $(x,y,z)$  para espaços tridimensionais ([Math Inside, 2020](#)).

Mas, para isso, é necessário, em primeiro lugar, converter duas coordenadas em radianos oriundas dos valores de fase para três coordenadas cartesianas, para que cada ponto possa agora ser identificado num espaço tridimensional, que facilite a leitura das fases com o seu carácter cíclico. Este processo é conseguido através das equações matemáticas seguidamente descritas, nas quais  $r$  simboliza o raio da esfera,  $\theta$  a inclinação (ou latitude),  $\varphi$  o azimute (ou longitude) e em que  $r \in [0, \infty]$ ,  $\theta \in [0, \pi]$ ,  $\varphi \in [0, 2\pi]$ :

$$x = r \sin \theta \cos \varphi$$

$$y = r \sin \theta \sin \varphi$$

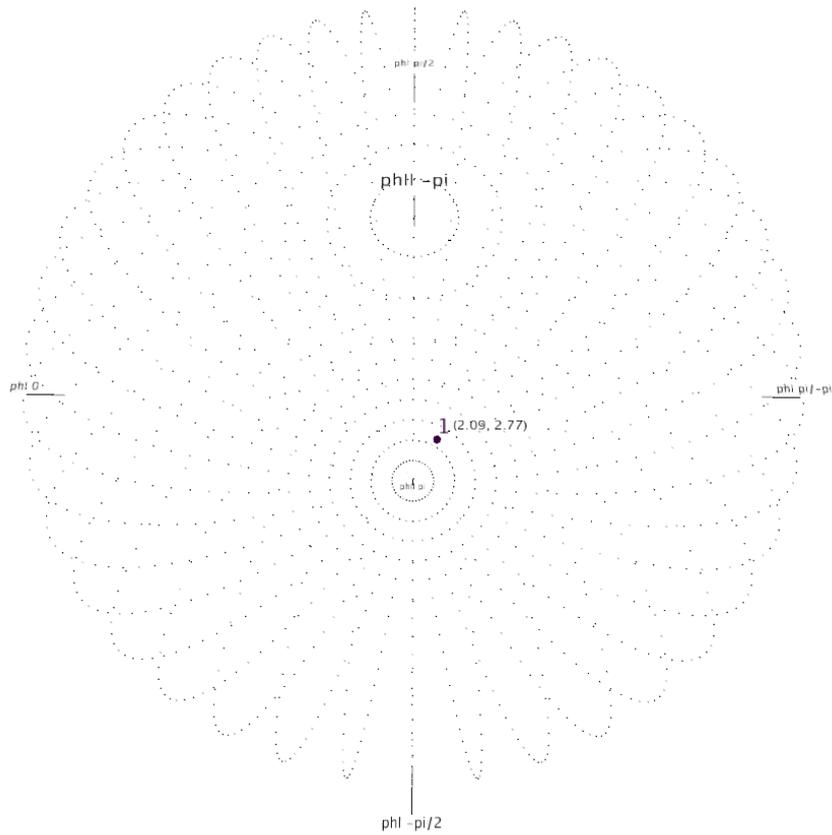
$$z = r \cos \theta$$

Com estas equações é agora viável a construção de um EGRH tridimensional, em que cada EH de uma peça, caracterizada por dois valores de fase correspondentes às componentes da TDF escolhidas de acordo com as duas principais CI existentes na peça, seja agora disposta na superfície de uma esfera, transformando para isso os valores de fase em coordenadas geográficas tridimensionais  $(x_1, y_1, z_1)$ , utilizando as equações anteriores. Esta representação foi desenvolvida no modo *Python* (2020), do *software Processing* (2020), que possibilita a criação de apresentações gráficas bidimensionais ou tridimensionais com grande facilidade através de um conjunto de funções próprias para esse efeito e que lhe são nativas. Com ela é possível representar cada um dos CA de forma progressiva, obedecendo à ordem do utilizador para que a próxima EH seja colocada no espaço e também reproduzido o seu áudio.

Como tal, inicialmente, foi necessário escrever o código que, algoritmicamente, desenha num espaço tridimensional um conjunto de pontos que formam a esfera onde mais tarde serão representados todos os CA em análise. Para isso, é definido o raio da esfera (no caso *700 pixels*), escolhido o número total de pontos que a vão definir (40) e calculadas todas as suas coordenadas no espaço. Foram, também, acrescentadas linhas de apoio à interpretação de resultados, que servem de referenciais ao gráfico. É possível, ainda, com o suporte da biblioteca para *Processing* (2020), intitulada *peasycam*, navegar livremente no espaço tridimensional, aproximando ou afastando a perspectiva de visão para um determinado ponto. A Figura 3.6 mostra o aspeto final da esfera.

Depois disto, são importados para este *software* os valores de fase calculados no módulo anterior (um dos *inputs* fundamentais deste módulo), ou seja, no exemplo em que a CI mais repetida na peça é 1 e a segunda mais repetida é 6, a lista de fases é do tipo:

$$phase\_list = [[A1_{phase_{\hat{f}_1}}, A1_{phase_{\hat{f}_6}}], \dots, [An_{phase_{\hat{f}_1}}, An_{phase_{\hat{f}_6}}]]$$



**Figura 3.6:** A esfera de fases desenvolvida para este trabalho e a representação nela de um ponto referente à primeira EH do terceiro andamento da peça *Sechs Kleine Klavierstück*, de A. Schoenberg.

e, posteriormente, calculadas todas as suas respectivas coordenadas cartesianas tridimensionais com recurso às equações suprarreferidas. Posteriormente, a lista anterior pode ser, então, interpretada como:

$$phase\_list = [[A1_{\varphi}, A1_{\theta}], \dots, [An_{\varphi}, An_{\theta}]],$$

em que  $\varphi$  simboliza a longitude e  $\theta$  a latitude e, portanto, o primeiro valor de fase de cada EH faz variar o quão a este ou oeste o ponto se encontra na superfície da esfera e, o segundo valor de fase, altera o quão a norte ou a sul ele se posiciona. Depois deste cálculo, passa-se a contar com uma nova lista de coordenadas cartesianas com a seguinte estrutura:

$$cartesian\_list = [[A1_{x_1}, A1_{y_1}, A1_{z_1}], \dots, [An_{x_n}, An_{y_n}, An_{z_n}]],$$

permitindo, assim, a representação de todas as EH de uma peça na superfície de uma esfera. Estas EH são sempre acompanhadas de uma indicação entre parêntesis dos valores de fase calculados a partir da TDF, antes de calculadas as suas coordenadas tridimensionais. A Figura 3.6 mostra um exemplo da representação de um CA no espaço de fases esférico  $\varphi_{4/5}$ , cujos valores de fase correspondentes são [2.09, 2.77].

Em paralelo, esta representação gráfica possibilita, de forma muito simples, a reprodução áudio de cada nova EH colocada no espaço. Isso acontece com a ajuda da biblioteca para *Processing* (2020) intitulada *sound*, que, importada a lista MIDI do módulo B, lê e reproduz todas as notas individuais do CA, tocando para o efeito as respetivas *samples* de piano que se encontram dentro da pasta *data* do *software*.

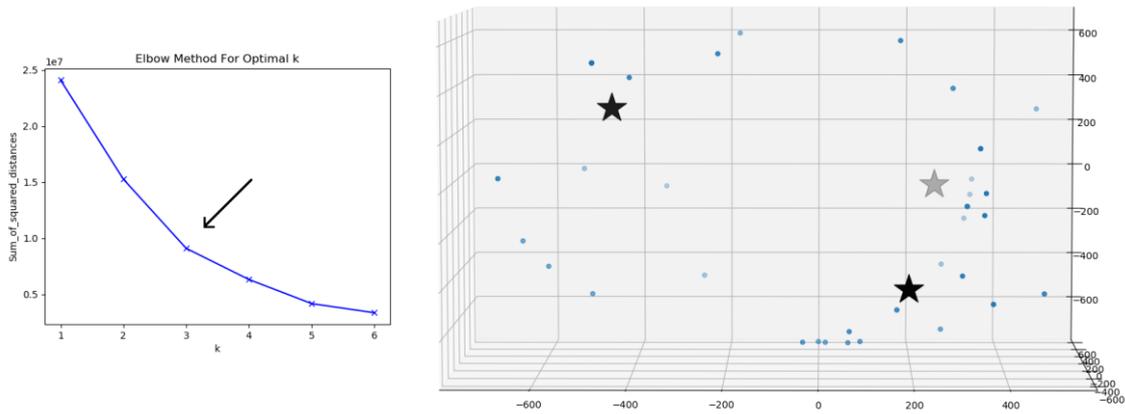
Todo o código escrito em *Python* (2020) para este módulo é referente ao *software Processing* (2020) pode ser consultado no [Anexo 1](#) e nos *scripts* intitulados [spherical\\_phase\\_space.pyde](#), [sps\\_utilities.py](#), [midi\\_utilities.py](#) e [draw\\_utilities](#), mais concretamente nas funções `points_sphere`, `points_rads`, `points_cartesian`, `midi_reader`, `draw_sphere` e `draw_references`.

### 3.6 Agregados Harmónicos

Uma vez criada a esfera de fases para a representação harmónica, é agora possível obter resultados mais fiáveis para a criação de grupos distintos de EH dentro do espaço. O objetivo é calcular algoritmicamente o número ótimo de agregados para um dado posicionamento de toda a informação harmónica e, de seguida, perceber a qual desses grupos cada CA pertence.

Para conseguir estes objetivos utiliza-se um algoritmo de *clustering*, intitulado *k-means* (2020). Este é um algoritmo de aprendizagem de máquina que permite agrupar informação similar e descobrir padrões nela e, para isso, procura a melhor forma de dispor essa informação em grupos distintos, dado um número total de grupos fixo  $k$ . O processo deste algoritmo passa por calcular um número total de centroides igual ao de  $k$ , ou seja, o posicionamento no espaço do centro de cada agregado, e, de seguida, mapear cada ponto à centroide mais próxima. Em alguns casos, este agrupamento é simples e intuitivo de ser feito, mas, em outros, é essencial a existência de um algoritmo que nos apoie neste processo.

O primeiro passo para a implementação do algoritmo é, uma vez importados para o *script* o raio da esfera e a informação de fases, calcular as coordenadas cartesianas, utilizando o método e equações enunciados no módulo anterior, e armazenar essas coordenadas numa *array* própria para o cálculo deste algoritmo. De seguida, graças à biblioteca para *Python* (2020), intitulada *sklearn*,



**Figura 3.7:** O algoritmo *k-means* (2020): à esquerda, o gráfico para determinar o número ótimo de  $k$  (3) para um número de grupos entre 1 e 6 e, à direita, o cálculo das centroides dos 3 grupos selecionados.

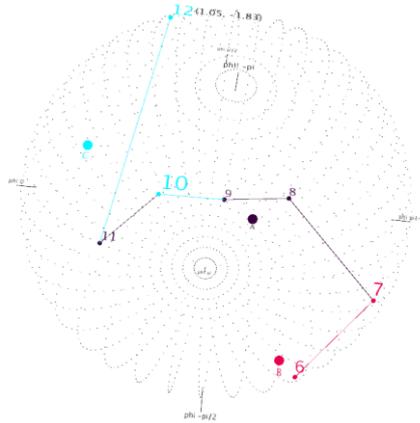
é possível, em primeiro lugar, determinar o número ótimo de  $k$  e, de seguida, calcular as centroides dos AH.

Para determinar o valor ideal de  $k$  (por exemplo, entre 1 e 6, porque o máximo de grupos distintos está associado ao número de CI) utiliza-se o método do “cotovelo”. Este método traduz-se em correr o algoritmo para os diferentes valores de  $k$  propostos e, de seguida, confrontá-los graficamente com a soma do quadrado das distâncias entre cada ponto e a sua respetiva centroide. Este gráfico tem, tendencialmente, um ponto de inflexão claro, o “ponto do cotovelo”, e o valor de  $k$  nesse ponto de inflexão é o número ótimo de grupos. No caso da informação harmónica analisada neste estudo, esse número é três agregados distintos e a Figura 3.7, à esquerda, mostra um exemplo de como esse valor de  $k$  foi determinado.

Conhecido o número ideal para o agrupamento ( $k = 3$ ) foi escrito o código que permite fazer o agrupamento da informação e exportar para o *Processing* (2020) as coordenadas  $x, y, z$  do posicionamento das centroides de cada grupo (a primeira centroide é o grupo 0, a segunda o 1 e a terceira o 2), bem como uma lista do agregado a que cada ponto pertence. Essa lista, neste caso, pode conter somente os valores 0, 1 ou 2 (já que  $k = 3$ ) e tem a seguinte tipologia:

$$labels\_list = [label_{EH_1}, label_{EH_2}, \dots, label_{EH_n}],$$

onde *label* representa o valor entre 0 e 2, que especifica o agregado; EH simboliza a estrutura harmónica em análise e  $n$  o número total de EH na peça. O posicionamento desse valor entre 0 e 2 na lista indica a que EH pertence essa informação. Por exemplo, se ele está na primeira posição da lista, então, diz respeito à primeira, se está na segunda, à segunda EH, e assim sucessivamente. A Figura 3.7, à direita, mostra a informação harmónica de uma peça disposta num espaço tridimensional e o posicionamento, nesse mesmo espaço, das centroides calculadas pelo algoritmo



**Figura 3.8:** As trajetórias na esfera de fases e o centro dos AH calculados com o algoritmo *k-means* (2020).

*k-means* (2020). Assim sendo, é possível perceber e exportar a informação sobre a que agregado pertence cada uma das EH e representar distintamente no espaço esférico de fases esses três grupos, bem como construir um conjunto de ferramentas de análise a partir deles.

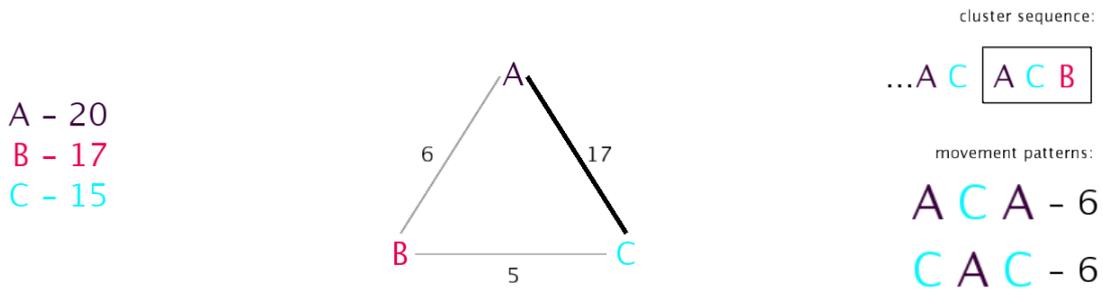
Todo o código escrito em *Python* (2020) para este módulo pode ser consultado no [Anexo 1](#) e em todas as funções encontradas nos *scripts* intitulados [kmeans.py](#) e [kmeans\\_utilities.py](#).

### 3.7 Detecção de Padrões

Criado o espaço de fases esférico referido em 3.5 e importado o agrupamento das EH em 3 agregados distintos mencionados em 3.6, segue-se a escrita do código em *Processing* (2020) que adicione à representação gráfica algumas ferramentas que facilitem a percepção de trajetórias e a detecção de padrões na obra em análise.

Entre as ferramentas implementadas está, em primeiro lugar, a distinção em três cores distintas dos grupos oriundos do algoritmo de *clustering*, que no *Processing* (2020) passam a ser representados pelas letras A, B e C (assim como o posicionamento das centroides calculadas no módulo E), mas também a possibilidade de perceber as trajetórias que o percurso harmônico que a peça vai fazendo, através de linhas que percorrem o caminho mais curto entre dois pontos colocados no espaço. Assim, a aplicação está predefinida para apresentar sempre um total de sete pontos e, conseqüentemente, as seis linhas que os vão unindo. A Figura 3.8 mostra um exemplo dessas trajetórias e da representação das centroides dos AH.

Para além disto, foram implementadas três outras ferramentas muito simples que auxiliam a detecção de movimentos padronizados no espaço e entre grupos de CA. A primeira contabiliza, à medida que os pontos vão aparecendo no espaço, o total de ocorrências em cada agregado. Por outras palavras, e a título de exemplo, se a peça tem vinte EH pertencentes ao grupo A, dezassete



**Figura 3.9:** As ferramentas de detecção de padrões: à esquerda o contador de ocorrências em cada agregado; ao centro, o contador de movimentos entre agregados e, à direita, o registro do último movimento entre três grupos distintos e aqueles que mais vezes se repetiram.

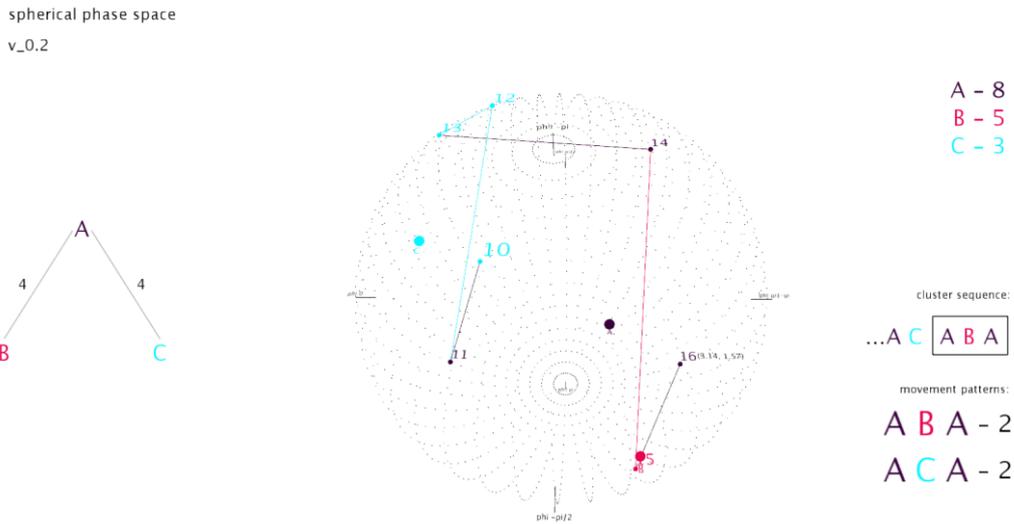
ao B e quinze ao C, então, no final da disposição de todos os pontos no espaço, essa indicação surgirá na interface da aplicação (tal como pode ser visto no primeiro exemplo da Figura 3.9).

Analogamente, foi também escrito o código que possibilita um segundo instrumento de análise de padrões, desta feita, contabilizando o número de movimentos ao longo da peça entre grupos de CA, ou seja, de cada vez que uma nova EH é disposta no espaço e existe uma mudança de agregado, então essa mudança é devidamente contabilizada. Com este método existem somente três movimentações possíveis, uma vez que também só existem três agregados distintos, ou seja, mudanças de grupo entre A e B, entre B e C ou C e A (esta ferramenta não distingue o sentido da mudança, isto é, contabiliza de igual forma uma mudança entre A e B ou B e A). Para melhor visualizar a importância de determinado movimento, também à medida que uma mudança entre grupos se vai repetindo torna-se cada vez mais saliente a linha entre eles na interface gráfica (o segundo exemplo da Figura 3.9 mostra as características deste instrumento).

Finalmente, o terceiro método concretizado para a análise de resultados, possibilita verificar qual a última combinação de duas mudanças entre os AH e, também, visualizar aquela que mais vezes se vai repetindo. Por outras palavras, existem doze hipóteses distintas para registrar essa mudança com três grupos (A, B e C) de EH, sendo elas:

$$[ABC], [ACB], [BAC], [BCA], [CBA], [CAB], [ABA], [ACA], [BAB], [BCB], [CAC], [CBC].$$

Como tal, o *software* vai registrando e colocando na sua interface (à medida que novos pontos vão aparecendo no espaço) qual destas doze movimentações foi a última a acontecer, como também aquelas que mais vezes se repetem ao longo da peça (o terceiro exemplo da Figura 3.9 mostra como esta ferramenta é desenhada na aplicação).



**Figura 3.10:** A interface final da aplicação com a representação esférica de fases e o conjunto de ferramentas de detecção de padrões desenvolvidas.

Uma vez programados o espaço esférico e os seus instrumentos de detecção de padrões é possível ver através da Figura 3.10 um exemplo da totalidade da interface da aplicação desenvolvida no âmbito deste projeto.

Todo o código escrito em *Python* (2020) para este módulo e referente ao *software Processing* (2020) pode ser consultado no [Anexo 1](#) e nos *scripts* intitulados [spherical\\_phase\\_space.pyde](#), [sps\\_utilities.py](#), e [draw\\_utilities.py](#), mais concretamente nas funções `cluster_center`, `labels_reader`, `color_generator`, `draw_cluster_centers`, `draw_n_paths`, `software_name`, `draw_paths_representation`, `draw_cluster_counter_representation`, `draw_cluster_sequence` e `draw_preferred_cluster_movements`.

### 3.8 Súmula

Este capítulo abordou todo o trabalho de implementação de uma ferramenta que permite a representação gráfica de todos os CA de uma peça num espaço esférico de representação harmónica, derivado da TDF. Para isso, a EFF parte de uma partitura em formato *.XML* e, através de um conjunto de operações algorítmicas divididas em seis módulos distintos, que permitem (1) analisar a partitura e segmentá-la harmonicamente, (2) transformar a informação obtida para cálculos futuros, (3) calcular a TDF e as fases a si associadas, (4) desenhar um espaço esférico de representação harmónica, onde se dispõem pontos referentes a todos as EH da partitura, (5)

agrupar esses pontos em três agregados distintos e (6) analisar a existência de padrões, cria uma aplicação multimédia controlada pelo utilizador, cuja interface pode ser vista na Figura 3.10.

Curiosamente, foram os principais problemas identificados ao longo do desenvolvimento da EFF, que, em boa verdade, motivaram os principais contributos desta implementação, dos quais destaco, em concreto: (1) a possibilidade de utilizar a contagem das CI de uma peça para o estabelecimento do espaço de fases a representar como forma de responder a conteúdos intervalares e EH muito distintos nas peças atonais; (2) o desenvolvimento de um novo espaço esférico de representação harmónica inspirado nos conceitos geográficos de latitude e longitude para fomentar uma representação final que possa expressar o carácter cíclico das fases calculadas através da TDF e, também, (3) a necessidade de calcular, através de algoritmos de *clustering*, um conjunto de grupos distintos referentes aos CA representados no espaço esférico para a deteção de movimentos padronizados entre eles.

No capítulo seguinte pode ler-se a utilização desta ferramenta para a análise sintática do terceiro andamento da obra *Sechs Kleine Klavierstück* referente ao período do atonalismo livre de Arnold Schönberg.



## 4. Schönberg na EFF

Ao longo deste Capítulo 4, pretende-se avaliar a eficácia da EFF descrita anteriormente na compreensão da sintaxe harmónica de obras referentes ao período atonal, nomeadamente, ao período conhecido como atonalismo livre de Schönberg. Para isso foi escolhido o seu Op.19, intitulado *Sechs kleine klavierstück* (seis pequenas peças para piano), mais concretamente o seu terceiro andamento, para, depois de notado e criada a sua partitura em formato *.XML*, dispor todas as suas EH na EFF e daí extrair conclusões quanto à organização sintática da obra. Note-se que a análise feita neste capítulo debruça-se meramente na componente harmónica da peça, uma vez que a EFF foi desenhada para avaliar unicamente esse campo, não se preocupando com aspetos melódicos, rítmicos e expressivos e não pretendendo, por essa razão, ser uma análise verdadeiramente abrangente. Para este efeito, este capítulo encontra-se dividido em três secções distintas: a primeira, a peça em análise (§4.1), foca-se na escolha da obra, no período composicional a que pertence e no seu compositor para uma breve contextualização histórica; a segunda, Op.19 / n.º 3 (§4.2), faz a análise propriamente dita dos resultados obtidos pela EFF, interpretando-os, por isso, do ponto de vista musical e, por fim, a súpula (§4.3) que refere muito brevemente as principais ideias abordadas ao longo das páginas do Capítulo 4.

### 4.1 A Peça em Análise

Sendo necessária a escolha de uma obra para a validação da esfera de fases no entendimento sintático atonal, nomeadamente para a deteção de movimentos padronizados entre diferentes AH e a sua interpretação musical, optou-se pela escolha de uma peça que diz respeito ao período estético da obra de Schönberg conhecido como atonalismo livre, por ser o primeiro momento em que conscientemente um compositor tentou sistematizar um processo composicional sem um centro tonal bem definido ([Perle, 1991](#)). Na verdade, obras anteriores, como a *Bagatelle sans tonalité*, de Franz Liszt, já enunciavam esta ideia, mas foi com Schönberg que ela verdadeiramente se sistematizou.

Assim sendo, esta secção 4.1 pretende contextualizar historicamente três aspetos: em primeiro lugar, em §4.1.1, o compositor Arnold Schönberg e os seus períodos estéticos; seguidamente, em §4.1.2, o atonalismo livre onde são enunciadas algumas das principais características que o definem e, em §4.1.3, a obra escolhida para a análise de resultados feita ao longo deste Capítulo 4.

### 4.1.1 Arnold Schönberg

Arnold Schönberg foi um compositor Austríaco nascido em 1874, em Viena. Ele foi o responsável pelo passo definitivo na criação de obras musicais cujo desenvolvimento harmónico não tem um centro tonal bem definido e às quais se chama, até aos dias de hoje, peças atonais ([Michels, 2001](#)). Este passo abalou um conjunto de tradições composicionais conquistadas durante séculos, mas, simultaneamente, era visto por outros compositores do início do séc. XX (Alban Berg ou Anton Webern, por exemplo) como inevitável face à progressiva emancipação da dissonância ao longo da história da música. Muito embora os teóricos tenham fixado, durante as décadas seguintes, o termo “atonalismo” para a definição desta música, Schönberg, curiosamente, não o defendia e, ao invés, preferia notar que se a existência de um termo definidor da sua música era importante para alguns, que se considerasse, então, outras hipóteses, devido ao facto de “atonalismo” ter, na sua opinião, uma conotação negativa, pois, como o próprio mencionava, não se chama à natação a “arte do não afogamento” ([Schönberg, 2010](#)).

É possível, então, distinguir na obra deste compositor três períodos estéticos distintos. O primeiro, compreendido entre 1895 e 1908, é amplamente descrito pelos analistas como tonal, ou seja, todo o discurso musical é influenciado pela presença de um centro claro. No entanto, a emancipação da dissonância e a ampla utilização do cromatismo, nas obras do final do séc XIX (não só em Schönberg, mas também em Wagner ou Webern), deixava antever o esgotamento deste sistema de composição e, assim, entre 1909 e 1914, surge um novo período estético intitulado de “atonalismo livre” descrito com maior detalhe na subsecção 4.1.2. Depois de vários anos sem escrever qualquer peça, o compositor desenvolve, na década de 20, o dodecafonismo, que tinha como principal objetivo o tratamento indiscriminado de cada um dos doze sons da escala cromática, sendo, por isso, o desenvolvimento musical alcançado por processos de inversão, retrogradação ou transposição de uma série de doze classes de altura distintas. Até à sua morte, Schönberg não abandonaria esta técnica composicional, sem, contudo, voltar em algumas obras ao tonalismo ([Michels, 2001](#)), não obstante, centrar-se-á o foco de análise no momento estético mais relevante para este trabalho.

### 4.1.2 O atonalismo livre

Caracterizar como “livre” o atonalismo que precedeu o período dodecafónico de Schönberg dificulta, por definição, a possibilidade de se sistematizar definitivamente um conjunto de

procedimentos composicionais generalizados associados a esta música ([Perle, 1991](#)), mas, por outro lado, reforça indubitavelmente a pertinência do seu estudo. Assim, entre 1909 e 1914, Schönberg escreveu um conjunto de peças que abririam caminho para um nível de ambiguidade métrica e harmónica sem precedentes na história da música, uma vez que, ao contrário da música tonal, em que a escrita correta de uma nota depende da EH pré-estabelecida a que pertence, na música atonal, ela deve ser avaliada recorrendo a fatores composicionais de maior abrangência, cujo significado deve ser descoberto a cada peça ([Perle, 1991](#)).

Ainda assim, é possível identificar, não raras vezes, como principal elemento integrador do discurso musical nas obras do atonalismo livre uma célula de CI (à qual se dá o nome de classe, que foi discutida no Capítulo 2), que, aparecendo horizontal ou verticalmente, ou através de um conjunto de transformações como transposição ou permutação, pode ser expandida e estabelecida como principal “motivo” da obra ([Perle, 1991](#)). Por essa razão, o compositor escolhe uma célula musical pequena, intencionalmente distante de estruturas típicas tonais e, seguidamente, usa-a múltiplas vezes na obra sobre variadíssimas formas, tirando proveito de notas comuns entre transformações da célula original, para as fazer coincidir no tempo. Por outras palavras, ela define como que um microcosmo de conteúdo intervalar fixo, que pode ser verificado na forma de acorde, na forma melódica ou ambos ([Perle, 1991](#)). No entanto, dado o carácter extremamente livre deste período composicional, nem sempre é fácil identificar essa célula, pois ela varia de obra para obra, mas, dessa forma, exponencia também a sua ambiguidade e expressividade.

Uma das últimas peças referentes a este período estético escrita por Schönberg é o seu Op.19, intitulado *Sechs Kleine Klavierstück*, e sobre o qual versarão as próximas páginas deste capítulo, mais concretamente, no seu terceiro andamento.

### 4.1.3 *Sechs kleine klavierstück, Op.19*

As seis pequenas peças para piano (*Sechs kleine klavierstück*) foram escritas no ano de 1911 num período em que Schönberg compunha pouco, porque, ele próprio o admitiu, sentia que tinha perdido a certeza no caminho composicional que vinha trilhando. Como tal, poucas obras eram iniciadas e menos ainda terminadas. No entanto, essa incerteza não impediu que o compositor escrevesse os cinco primeiros andamentos num único dia, a 19 de fevereiro de 1911, e, ao contrário de obras anteriores em que procurava expandir o seu idioma musical, estes cinco andamentos foram escritos consolidando as técnicas composicionais do atonalismo livre que antes tinha experimentado e são, definitivamente, miniaturas nas suas proporções globais, evitando, até, algumas, desenvolvimento motivico ([Simms, 2000](#)).

A verdade é que o compositor, inicialmente, projetava somente estes cinco andamentos para a obra, mas, a 17 de junho desse mesmo ano, escreveria o sexto e último andamento muito influenciado pela morte de Gustav Mahler, que falecera sensivelmente um mês antes. Assim, as *Sechs Kleine Klavierstück* foram publicadas, somente em 1913, pela *Universal Edition*, como Op.19 ([Simms, 2000](#)). Destas seis pequenas peças foi escolhida para a análise, que se pode ler

nas páginas seguintes, o terceiro andamento. Texturalmente falando, ele é bastante conservador, pois apresenta, maioritariamente, uma melodia tocada à oitava na mão esquerda, acompanhada por tetracordes na mão direita. Para além disso, a sua coerência harmónica é por demais evidente, salientando-se o facto da utilização da mesma EH (transposta uma 3<sup>a</sup>M descendente) no primeiro e último tetracorde da peça ([Simms, 2000](#)).

## 4.2 Op. 19 / n.º 3

Nesta subsecção podemos ler os resultados obtidos através da EFF, descrita no Capítulo 3, para a análise harmónica do terceiro andamento das seis pequenas peças para piano de Schönberg, bem como a sua respetiva interpretação musical. Quanto a essa interpretação, a generalidade dos resultados é explicável, havendo, contudo, perguntas que continuam a carecer de resposta para alguns dos resultados encontrados. É de notar também que para esta análise, nomeadamente para a segmentação em partes da obra, é utilizada a estrutura formal proposta por Boss ([Boss, 2019](#)), mas também as ideias de Newton ([Newton, 2013](#)) quanto à importância da CI 6 nesta música, como forma de as comparar com alguns dos resultados obtidos.

A estrutura desta subsecção é, então, a seguinte: a estrutura formal do andamento (§4.2.1), onde se separa a obra em três partes distintas, a definição do espaço de fases (§4.2.2), para justificar a utilização do espaço  $\varphi_{4/5}$ , a interpretação dos AH (§4.2.3), para melhor compreensão do significado do posicionamento no espaço das várias EH, as secções I, II, e III da peça (§4.2.4, §4.2.5 e §4.2.6 respetivamente), para análise mais pormenorizada, e, finalmente, a avaliação global (§4.2.7), para reflexão das principais ideias sintáticas obtidas para o andamento como um todo.

### 4.2.1 Estrutura formal do andamento

O terceiro andamento das seis pequenas peças para piano está dividido em três secções distintas. A primeira (I) encontra-se entre os compassos 1 a 4, a segunda (II) nos compassos 5 e 6 e, por fim, a terceira (III) nos três últimos compassos da obra, ou seja, entre o 7 e 9 ([Boss, 2019](#)). A Figura 4.1 mostra a partitura do andamento, a sua divisão formal e a numeração de cada EH analisada e disposta na EFF. A partitura integral deste andamento sem qualquer anotação pode ser consultada no [Anexo 2](#).

As duas primeiras secções têm uma textura contrapontística idêntica, com linhas melódicas em *legato* em ambas as mãos, mas, em contraste, o compositor distingue a mão direita da esquerda de uma forma muito clara na secção I através de patamares dinâmicos bastante distantes, ou seja, enquanto a mão direita deve ser tocada *forte*, a esquerda tem indicação de *pianíssimo*. Em I, é

## Sechs kleine Klavierstücke, Op. 19

## III.

Sehr langsame  Arnold Schoenberg - 1911

Secção I | 1 2 3 4 5 6 7 8 9 10 11 12 13

Piano

*f*

*pp*

*pp*

2 | 14 15 16 17 18 19 20 21 22 23 24

*pp*

*pp*

Secção II | 25 26 27 28 29 30 31 32

4

*p*

Secção III | 33 34 35

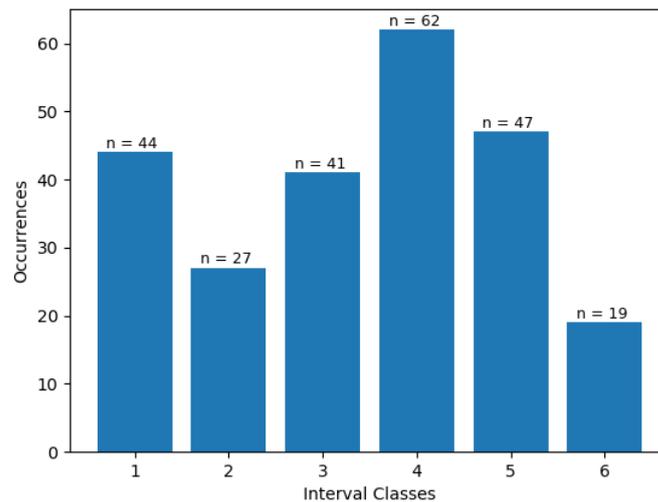
6

*pp*

*ppp*

**Figura 4.1:** A partitura do Op.19 / n.º 3, de Schönberg. É possível distinguir-se na figura as suas três secções distintas e a numeração de todas as EH analisadas na EFF. Partitura editada manualmente no âmbito desta dissertação.

também de salientar a utilização de uma melodia à oitava na mão esquerda, que é abandonada, de resto, em II. Por outro lado, a secção III tem um carácter bastante distinto à custa, sobretudo, da



**Figura 4.2:** A contagem de CI do Op.19 / n.º 3, de Schönberg.

mudança de articulação para o acompanhamento da melodia, isto é, ele agora é feito com recurso a tetracordes muito curtos, alguns até notados em *staccato* (Boss, 2019). A análise harmónica detalhada para cada uma destas três secções pode ser lida em §4.2.4, §4.2.5 e §4.2.6.

#### 4.2.2 Definição do espaço de fases

No âmbito da análise dos resultados obtidos com a EFF, descrita no Capítulo 3, a primeira informação de relevo é a extração da totalidade de CI presentes neste andamento, pois é ela que nos permite não só ter um indicador claro do conteúdo harmónico da peça e das suas estruturas intervalares mais frequentes, mas também definir quais as duas componentes da TDF às quais devem ser calculadas as respetivas fases para a representação esférica subsequente. Relembre-se o facto de se exportar somente as fases referentes a essas duas componentes numa lista própria. A Figura 4.2 mostra o cálculo de todas as CI presentes no terceiro andamento da obra.

Através da análise da Figura 4.2 podemos verificar que as duas CI que se destacam relativamente às demais são, por ordem decrescente, com número de ocorrências  $n = 62$ , a CI 4, correspondente à 3ªM e 6ªm e, com  $n = 47$ , a CI 5, que diz respeito aos intervalos de 4ªP e 5ªP. De facto, mesmo num olhar rápido pela partitura, este aspeto é claro. Por exemplo, atente-se nas EH da mão direita no primeiro e terceiro tempos do primeiro compasso. A primeira é um tetracorde constituído com as classes de altura [1, 2, 6, 11], ou seja, com as notas [C#, D, F#, B], verificando-se nele, entre D e F#, a presença de uma 3ªM, e entre F# e B, uma 4ªP. Também no terceiro tempo é possível ver-se na mão direita um tricorde com classes de altura [0, 4, 7] ou [C, E, G] em que, de novo, entre as notas G e C se verifica uma 4ªP e, entre C e E, uma 3ªM. Em boa verdade, estes exemplos multiplicam-se ao longo do andamento, sendo, por essa mesma razão, as CI 4 e 5 as mais preponderantes.

Num segundo patamar de importância, mas ao qual se pretende dar também destaque, surgem os intervalos de 2ªm ou 7ªM referentes à CI 1. Eles, embora em menor número face às CI 4 e 5, são a principal fonte de dissonância na obra, sendo que, preferencialmente, o compositor escreve ao longo do andamento a 7ªM, tornando-se, assim, a principal representante da CI 1. Este aspeto é, naturalmente, digno de registo pelo facto de a música deste período ter emancipado a dissonância, tornando-a intrínseca à sonoridade da obra sem qualquer necessidade de preparação ou resolução como nos períodos anteriores.

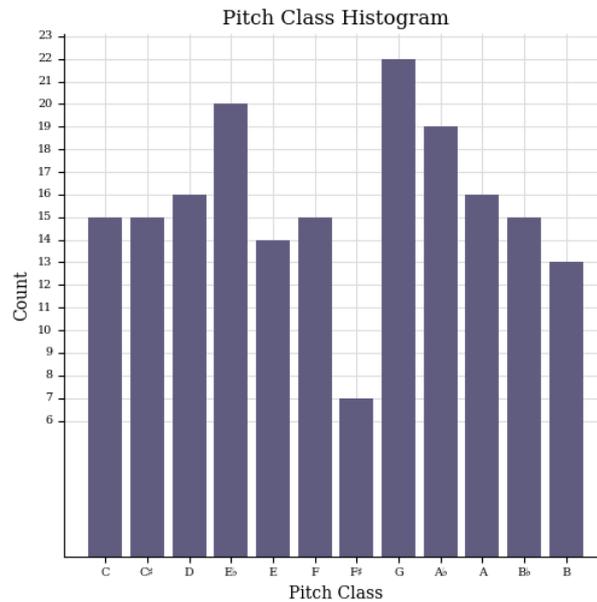
Deste modo, e com base na Figura 4.2, foram calculadas as fases correspondentes às componentes  $\hat{f}_4$  e  $\hat{f}_5$ , da TDF para cada EH da peça, com o intuito de as representar na superfície da esfera de fases  $\varphi_{4/5}$ .

### 4.2.3 Interpretação dos agregados harmónicos

Nos Capítulos 2 e 3 já foi abordado o espaço de fases, no entanto é útil que ele seja interpretado à luz dos resultados obtidos para esta obra em concreto, o que, convenientemente, facilita a compreensão dos AH obtidos com o algoritmo de *k-means* (2020). Para esse efeito, um dos primeiros conceitos sobre o qual importa refletir é o conceito de distância dentro de um espaço de fases, ou seja, que características partilham EH cujo posicionamento seja próximo, e, complementarmente, que características distinguem CA posicionados em pontos muito distantes dentro da EFF. Essa distância (dentro de um espaço de fases) baseia-se na ideia de partilha (ou não) de classes de altura, ou seja, quanto mais sons partilharem duas EH, mais próximas elas serão dispostas, uma vez que o seu posicionamento pode ser visto como uma aproximação do posicionamento de todas as classes de altura individuais que o constituem e é, conseqüentemente, sensível a transposições e inversões do material harmónico (Yust, 2015). Por exemplo, o primeiro tetracorde do andamento, constituído pelas classes de altura [2, 6, 11, 1], ou seja, com as notas [D, F#, B, C#], encontra-se no espaço posicionado muito próximo da localização de todas as suas classes de altura individuais, tal como pode ser visto na Figura 4.3. Para isto, não é inocente o facto de estarmos num espaço  $\varphi_{4/5}$ , porque é ele que determina o posicionamento concreto de cada classe de altura e, naturalmente, a localização das diferentes EH da peça. O que daqui resulta é que se o espaço fosse distinto, a mero título exemplificativo, se fosse um espaço  $\varphi_{1/6}$ , então todos os posicionamentos de classes de altura individuais teriam uma localização diferente daquela que pode ser vista na Figura 4.3 e, conseqüentemente, as diferentes EH constituídas por elas também.

Entendida que está a ideia de distância no espaço de fases é relevante entender os AH sugeridos pelo algoritmo *k-means* (2020) e interpretá-los do ponto de vista musical. Nesse sentido, aquilo que o algoritmo agrupa, quando procura os pontos mais próximos no espaço, são





**Figura 4.4:** A contagem de classes de altura do Op.19 / n.º 3, de Schönberg.

autores procuram ver nas obras deste período resquícios da tonalidade e têm identificado a nota *Eb* como centro tonal deste andamento (Newton, 2013). Por outro lado, a terceira classe de altura mais repetida é a 8 (*G#*), com 19 ocorrências, e, contrastando com os anteriores grupos, não é tão evidente a relação com a centroide do grupo A, provavelmente, devido a um efeito de maior atração das notas *Eb* e *G* no posicionamento dos restantes CA. Uma outra característica de relevo é que, estas três classes de altura, vistas pela ordem [3, 7, 8], validam as CI 4 e 5 abordadas na subsecção 4.2.2 como as mais repetidas na obra, porque o intervalo [*Eb-G*] é uma 3ªM, enquanto que o intervalo [*Eb-Ab*] uma 4ªP.

Analizadas as centroides de cada um dos três AH, foi então possível associar cada conjunto de altura da obra ao seu respetivo grupo. Para isso, foram identificados, neste andamento, um total de trinta e cinco EH distintas para dispor no espaço, sendo cada uma delas numerada entre 1 e 35. As partituras segmentadas da peça, que podemos ver nas secções seguintes, mostram essa numeração e permitem a comparação entre o que acontece no espaço esférico de fases com a respetiva EH na partitura. Assim, o total de trinta e cinco entradas no espaço pode ser dividido como dezassete pertencendo ao agregado A, dez ao B e oito ao agregado C.

#### 4.2.4 Secção I

A primeira secção da obra tem quatro compassos e vinte e quatro EH distintas. É a maior das três secções da obra e, por essa razão, também aquela que mais influencia os resultados

The figure consists of two parts. The upper part is a musical score for piano, spanning 24 measures. The right hand (treble clef) starts with a forte (*f*) dynamic, playing a melodic line with some rests. The left hand (bass clef) plays a continuous, flowing accompaniment in a lower register, marked with pianissimo (*pp*). The score includes various musical notations such as slurs, ties, and dynamic changes. The lower part of the figure is a diagram illustrating the harmonic elements (EH) of the first section. It shows several boxes, each containing a set of numbers representing notes. These boxes are interconnected with lines and labels, indicating their harmonic relationships. Labels include (0237), (0148), (01347), (013478), (016), (014589), (01234), (014589), (01378), (0156), (013568), (01458), (0257), and (0124589).

Figura 4.5: As EH da secção I. (Boss, 2019)

obtidos, ainda que, contudo, se verifiquem nas demais. Nestes 4 compassos, é possível distinguir-se na mão direita dois gestos musicais, o primeiro no compasso 1 e 2 com sentido ascendente, e o segundo, depois de uma pausa de semínima, nos compassos 3 e 4, com sentido descendente. Enquanto isso, a mão esquerda toca uma melodia com carácter ligado sem qualquer interrupção em pianíssimo e com elevado contraste face ao seu acompanhamento. Esse contraste, de resto, é por demais evidente em qualquer audição da peça, transparecendo, até, uma total independência entre as duas mãos. Do ponto de vista harmónico, tal como se pode avaliar pela Figura 4.5, o tetracorde [2, 6, 11, 1], pertencente à classe [0237], da sonoridade inicial da obra (numerado com o número 1) parece ser de especial relevância, pois, para além do facto de abrir o andamento, várias outras EH aparentam ser próximas dele, tanto vertical como horizontalmente, e em múltiplas combinações, como por exemplo a junção das quatro primeiras notas da linha melódica do baixo [10, 3, 5, 8], que pertencem, portanto, à classe [0257] (Boss, 2019). Todas essas sonoridades próximas, como as classes [0148] ou [01378], podem ser analisadas na Figura 4.5.

No que aos AH diz respeito, a Tabela 4.1 mostra a correspondência obtida entre as vinte e quatro EH desta primeira secção com o seu respetivo agregado e mostra também o percurso da harmonia da secção dentro do espaço de fases (para melhor compreensão de toda a informação, estes movimentos são apresentados sob a forma de tabela ao invés de figuras representativas do espaço de fases). Desta feita, quando confrontados estes resultados com a abordagem teórica de Newton em que o autor sugere três tipos de movimentações intervalares entre EH para a definição de um modelo de incremento e alívio de tensão harmónica (muito utilizado na música tonal, mas que não é, *per si*, tonal), constata-se algumas semelhanças com as mudanças de AH propostas pelo sistema.

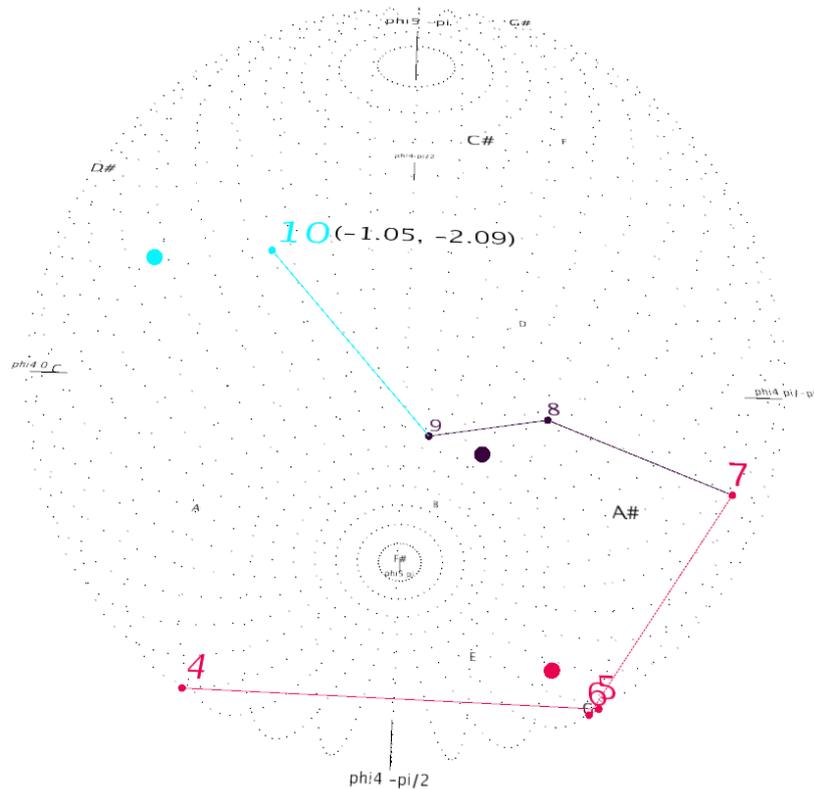
**Tabela 4.1:** A correspondência entre as EH da secção I e o seu agregado.

Correspondência EH/AH na secção I											
1	2	3	4	5	6	7	8	9	10	11	12
A	A	A	B	B	B	B	A	A	C	A	C
13	14	15	16	17	18	19	20	21	22	23	24
C	A	B	A	A	C	B	A	C	A	C	B

Um desses tipos de movimentação harmónica proposto por Newton (e, na verdade, aquele a que o autor dá maior relevância) ocorre quando a tensão intervalar do trítone (ou CI 6) resolve especificamente para uma 3ªM (CI 4). Exemplificando, um tetracorde constituído pelas classes de altura [2, 4, 10, 1], ou seja, pelas notas [D, E, A#, C#], tem na sua estrutura intervalar a CI 6, entre as notas [E, A#], e, dessa forma, cumpre o critério de Newton para uma movimentação harmónica de alívio de tensão, se este intervalo for sucedido pela CI 4 com movimento contrário entre as notas do trítone, ou seja, se a nota E se mover ascendentemente  $1/2t$  e A# descendentemente  $1/2t$  (Newton, 2013). Na mesma linha de pensamento, um tetracorde para suceder ao anterior e que cumpre este critério tem as classes de altura [1, 5, 9, 2], ou seja, as notas [C#, F, A, D], uma vez que a CI 6 do primeiro tetracorde [E, A#] é seguida da CI 4 [F, A] do segundo tetracorde.

Ora, este aspeto, tal como referido anteriormente, parece verificar-se em algumas das mudanças de EH que o sistema propõe. Isto deve-se, muito possivelmente, ao facto de este tipo de movimento intervalar não ter qualquer tipo de nota em comum, o que propicia distâncias maiores no espaço de fases, devido à não partilha de classes de altura. Um exemplo neste andamento, que Newton identifica como pertencendo a este tipo, pode ver-se no primeiro tempo do segundo compasso, em que a EH (numerada como 8) constituída pelas classes de altura [8, 11, 2, 7], ou seja, pelas notas [Ab, B, D, G], tem em si presente a CI 6 entre as notas Ab e D e, no segundo tempo do mesmo compasso (EH com número 10), a CI 4 é presente nas notas A e C# (movimento ascendente e descendente que Newton identifica), que se pode ler na EH [10, 5, 9, 1, 6], ou seja, com as notas [Bb, F, A, C#, F#]. Tanto na Tabela 4.1, quanto na Figura 4.6, é possível ver-se o trajeto no espaço esférico de fases dos pontos 8, 9 e 10, e constata-se que quando a CI 6 é resolvida para a CI 4 (dos pontos 8 e 9 para o 10), existe uma mudança clara de AH a que pertencem esses pontos, ou seja, enquanto os pontos 8 e 9 pertencem ao agregado A, o 10 já pertence ao C.

Para além das mudanças supramencionadas e que se constata na Tabela 4.1, as ferramentas de deteção de padrões, quando utilizadas para esta secção, mostram que existe um maior número de movimentos entre os agregados A e C (oito no total) e A e B (cinco) em qualquer das direções.



**Figura 4.6:** Exemplo de uma mudança de AH num movimento da CI 6 para a CI 4.

Esta característica de maior número de movimentações entre estes agregados (entre B e C existem apenas duas ocorrências) corrobora outro resultado obtido pelo sistema que destaca a sequência [BAC] entre grupos como a mais repetida nesta secção. As três aparências desta sequência podem ser verificadas na Tabela 4.1 entre os pontos 4 e 10, 15 e 18 e, por fim, entre os pontos 19 e 21. Através do posicionamento das centroides dos três agregados (visível nas Figuras 4.3 e 4.6), a sequência mais verificada [BAC] parece sugerir uma predominância de movimentos no espaço de fases no sentido contrário ao dos ponteiros do relógio<sup>10</sup>. Com efeito, a movimentação neste sentido é dada por três combinações distintas, [BAC] tal como agora mencionado, mas também pelas outras combinações que o garantem, nomeadamente, [ACB] e [CBA]. Desta forma, podemos verificar que este tipo de movimentação dentro do espaço de fases é ainda mais frequente porque se constata [ACB] entre os pontos 17 e 19, mas também entre 22 e 24 e [CBA] entre o 18 e 20. O total de trajetos no sentido agora mencionado é, por estas razões, de seis ocorrências. Uma possível explicação para este aspeto é avançada em 4.2.7, na avaliação global do andamento.

<sup>10</sup> Esta movimentação em sentido contrário ao dos ponteiros dos relógios pode ser entendida pela trajetória harmónica entre as EH 4 e 10 visíveis na Figura 4.6, que se inicia no ponto 4, desloca-se para a direita na esfera para os pontos 5 e 6, de seguida ascendentemente para 7, 8 e 9 e, finalmente, para a esquerda ao encontro do ponto 10. Este exemplo mostra um padrão típico de movimentação em sentido contrário ao dos ponteiros do relógio passando pelos agregados, B, C e A, por esta mesma ordem. Este sentido é ainda mais perceptível através das setas de movimentação presentes na Figura 4.10.

## 4.2.5 Secção II

The diagram below the score illustrates the pitch classes (EH) and their relationships. It features several boxes and lines connecting numbers 0 through 11. Key groupings include:

- A box containing 11, 7, 6, and 11, labeled (0125).
- A box containing 3, 9, 4, and 11, labeled (0157).
- A box containing 9, 3, 11, 8, and 2, labeled (01367).
- A box containing 9, 7, 3, and 2, labeled (015).
- A box containing 2, 0, 11, and 9, labeled (0136).
- A box containing 8, 10, 0, and 1, labeled (01367).
- A dashed box containing 9, 7, 3, 11, 8, and 2, labeled (0134679).

Figura 4.7: As EH da secção II (Boss, 2019).

A Secção II da obra tem somente dois compassos, nomeadamente entre o 5 e o 6, e apresenta um total de oito EH a dispor no espaço esférico de fases. Note-se que foram eliminadas para análise as classes de altura individuais 3 e 2, que aparecem no sexto compasso, uma vez que se pretende avaliar somente o trajeto harmónico da secção e, mais tarde, do andamento como um todo.

Estes dois compassos fazem a ligação entre o material musical da primeira parte e aquele que encontraremos na terceira e última secção. Note-se que a textura perdeu não só a melodia à oitava que antes se distinguiu, mas também a diferença dinâmica tão saliente (agora, ambas as mãos tocam em *piano*). No entanto, o carácter em *legato* mantém-se e não é certamente inocente que estes dois compassos terminem com duas notas da melodia desprovidas de qualquer acompanhamento, pois na secção seguinte essa característica assume uma maior preponderância. A classe [0237] continua a ser uma fonte indireta de material em múltiplas combinações horizontais e verticais, como se vê, por exemplo, pelo pentacorde do segundo tempo do compasso 5 constituído pelas classes de altura [0, 10, 4, 9, 3] e pertencente à classe [01367] (Boss, 2019). Todas estas relações indiretas entre classes derivadas da [0237] podem ser vistas com maior detalhe na Figura 4.7.

**Tabela 4.2:** A correspondência entre as EH da secção II e o seu agregado.**Correspondência EH/AH na secção II**

25	26	27	28	29	30	31	32
A	A	C	B	C	A	A	A

A Tabela 4.2 mostra a associação entre os pontos referentes a esta secção e o seu respetivo AH. Nela identificam-se, com facilidade, dois movimentos distintos entre agregados, sendo eles [ACB], entre os pontos 25 e 28, mas também [BCA], entre os pontos 28 e 32. O primeiro, [ACB], confirma o sentido contrário ao dos ponteiros do relógio no espaço esférico de fases detetado na primeira secção, mas o segundo, embora também sugerindo um movimento entre os três agregados, já diz respeito a um sentido igual ao dos ponteiros dos relógios. Também em 4.2.7 se pode ler uma reflexão a propósito deste facto.

**4.2.6 Secção III**

A terceira secção do andamento, que se encontra dos compassos 7 a 9, é, no âmbito desta análise, a mais curta de todas. Tal sucede, pois ela tem somente três EH (três tetracordes), um por cada compasso desta secção e, portanto, à semelhança do procedimento na secção II, foram eliminados da análise todos os momentos de classes de altura individuais visíveis na Figura 4.8.

The figure shows a musical score for three measures (33, 34, 35) and three corresponding pitch class diagrams. Measure 33 has a piano (*pp*) dynamic and a pitch class set of (0145) with fingerings 3, 4, 4, 3, 3. Measure 34 has a pianissimo (*ppp*) dynamic and a pitch class set of (0246) with fingerings 5, 7, 1, 3. Measure 35 has a pitch class set of (0237) with fingerings 9, 7, 2, 10.

**Figura 4.8:** As EH da secção III. (Boss, 2019)

**Tabela 4.3:** A correspondência entre as EH da secção III e o seu agregado.

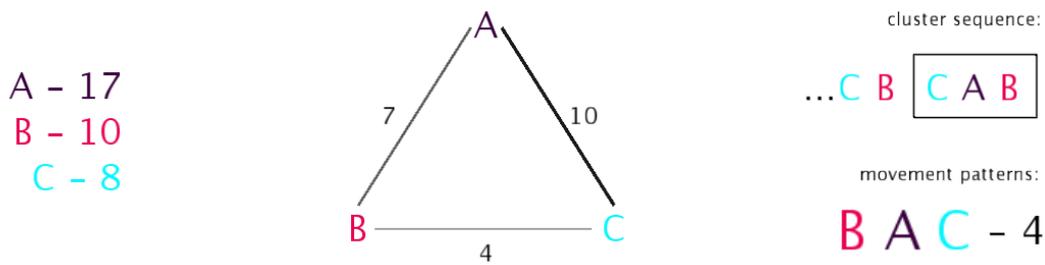
Correspondência EH/AH na secção III		
33	34	35
A	B	B

Para além disso é bastante contrastante relativamente às anteriores devido à presença maioritária de melodia sem acompanhamento, sendo este último reservado a três momentos de curtíssima duração, os dois últimos até notados em *staccato*. Esta característica textural é extraordinariamente eficaz para uma cadência bem sucedida, pois conjuga a nota G da melodia, que longamente se sustém nos compassos 8 e 9, com o acompanhamento em *staccato* tocado de forma quase inaudível. No seguimento das conclusões anteriores, a Figura 4.8 mostra, de forma ainda mais evidente, o destaque da classe [0237] não só por terminar a obra com as classes de altura [10, 2, 7, 9] (recorde-se que esta classe é igualmente a primeira que se ouve em todo o andamento), mas também por ser uma classe próxima das duas anteriores, nomeadamente de [0145] no tetracorde [8, 7, 0, 11], com as notas [Ab, G, C, B], presente no compasso 7, mas também, mais evidente ainda, da classe [0246], no compasso 8, com as notas [Eb, C#, G, F].

Sendo as EH a analisar somente as três que se podem ver na Tabela 4.3, é fácil perceber que existe uma única mudança entre os agregados A e B, do ponto 33 para o 34. Se a esta movimentação adicionarmos o último agregado, verificado na secção anterior, diferente de A, então, obtemos [CAB] que, embora num sentido diferente dos anteriores, ou seja, no sentido dos ponteiros do relógio, valida a tendência para a circularidade no espaço de fases das progressões harmónicas escolhidas pelo compositor.

#### 4.2.7 Avaliação global

A Figura 4.9 mostra os resultados obtidos para todo o andamento. Nela (na representação mais à esquerda) podemos ver que existe um total de dezassete EH registadas no agregado A, dez no B e oito no C. Esta característica, à primeira vista, parece contraditória com o gráfico do total de classes de altura visível na Figura 4.4. Nesse gráfico, é possível verificar que as classes de altura 3 e 7, ou seja, Eb e G, são as mais repetidas na obra e têm, muito próximas de si, as centroides dos agregados C e B, respetivamente. Ora, poderia ser expectável que estes agregados fossem, então, aqueles em que um maior número de pontos se encontra, pelo exato efeito de as notas Eb, e G serem muito frequentes. No entanto, esta ideia não acolhe nos resultados obtidos, porque cada uma destas classes de altura, tendo um agregado próprio, atrai para ambos a maioria das EH que têm as notas Eb e G, recebendo o agregado A a maioria das restantes combinações de classes de altura. Note-se, também, que, a zona no espaço esférico de fases  $\varphi_{4/5}$  em que se

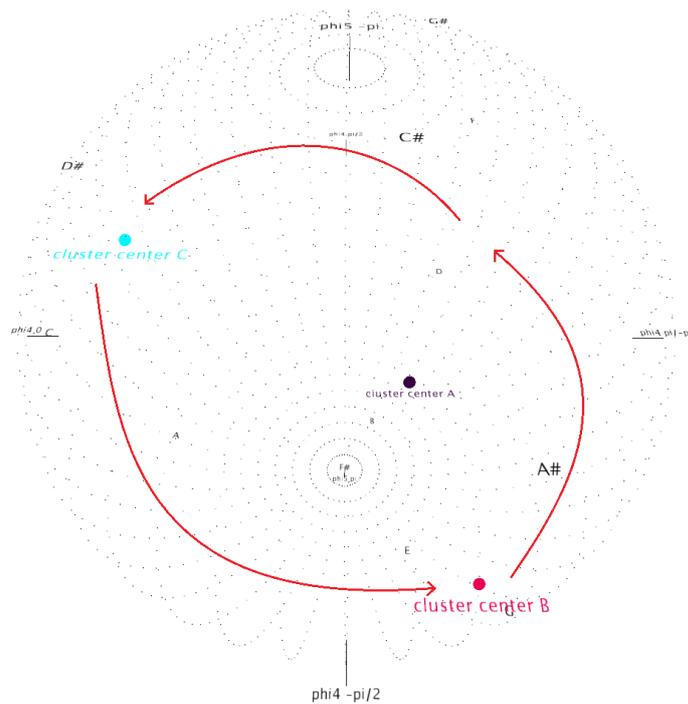


**Figura 4.9:** Os resultados obtidos para detecção de padrões no Op. 19 / n.º3.

encontra o agregado A, é uma zona próxima ao posicionamento das notas [D, F, F#, B], que, respetivamente, apresentam uma contagem de dezasseis, quinze, sete e treze ocorrências (visível na Figura 4.4), o que perfaz um total de cinquenta e um, explicando, assim, a maior preponderância do agregado A em número de EH individuais.

Até aqui, nas análises de secção, tem sido dado maior destaque aos momentos em que o sistema sugere mudanças entre AH, mas é também relevante perceber-se como se explica a permanência dentro de um determinado agregado, ou seja, o que se pode depreender quando o trajeto harmónico na peça é estável. A conclusão principal para esse facto tem que ver com a partilha de classes de altura, ou seja, quanto mais notas duas EH partilharem mais provável será que elas se encontrem no mesmo agregado. Desta feita, logo no primeiro compasso, as três primeiras EH encontram-se no agregado A (ver Tabela 4.1) e deve-se ao facto destes pontos partilharem as notas [D, B, C#], estando também este agregado muito influenciado pelo posicionamento das classes de altura 2 e 11, ou seja, as notas D e B.

Um outro aspeto de grande relevo já abordado na discussão das secções é a questão dos movimentos padronizados circulares dentro do espaço esférico de fases. A Figura 4.9 mostra, à direita, que foram identificadas quatro movimentações do tipo [BAC]. No entanto, quando percebemos que [BAC] simboliza uma movimentação no espaço no sentido contrário ao ponteiro dos relógios (visível na Figura 4.10) e consideramos, por isso, as outras movimentações com essa característica, ou seja, as movimentações [ACB] (com três ocorrências) e [CBA] (duas ocorrências), esse número aumenta para um total de nove. Este valor é altamente contrastante, por exemplo, quando comparado com o valor de ocorrências das combinações de AH [ABC], [BCA] e [CAB], que traduzem uma movimentação circular no sentido dos ponteiros do relógio e que só se verificam num total de três vezes no andamento. Aqui deve-se separar duas características que os resultados sugerem. A primeira é a questão da circularidade do trajeto harmónico dentro do espaço de fases, que parece apontar para progressões harmónicas que ocupam gradualmente grande parte, senão mesmo o total cromático (atente-se, por exemplo, no padrão [BAC] da Tabela 4.1 entre as EH 4 e 10, em que se pode verificar o preenchimento do total cromático ou o mesmo padrão entre os pontos 19 e 21, em que só faltam as classes de altura



**Figura 4.10:** A trajetória mais frequente entre AH no Op.19 / n.º 3. Exemplo demonstrativo de um movimento em sentido contrário ao dos ponteiros do relógio.

3, 4 e 6 para o total cromático), porque a passagem por três agregados distintos implica sequências harmônicas com partilha de poucas notas e, como tal, um maior número de notas distintas. Este facto é de especial relevância no âmbito deste estudo, pois pode representar um indicador fundamental para a organização sintática na música deste período. A segunda característica tem que ver com o sentido da circularidade. Neste andamento, é clara a predominância de movimentações circulares no espaço de fases em sentido contrário ao ponteiro dos relógios e, de alguma forma, é intrigante que essa diferença seja tão acentuada face ao outro sentido, porque nele também a ideia de utilização do total cromático se verificaria. Este facto sugere uma peça em falta na compreensão sintática da obra, pelo que uma possível explicação para esta característica poderá ter que ver com o fenómeno de direccionalidade na música que Huron (2016) propõe. Por outras palavras, na música encontramos uma sensação que nos impele para a frente no tempo (e só nessa direção), que faz com que progressões harmônicas com conduções melódicas próprias, quando tocadas em sentido contrário, percam toda a sua inteligibilidade e coerência (Huron, 2016). Isso deve-se, sobretudo, ao facto de quem ouve uma qualquer obra formar constantemente expectativas a propósito do que ouvirá de seguida, dando, naturalmente, um carácter unidireccional ao desenvolvimento musical que, quando em sentido contrário, tende a quebrar as expectativas formadas pelo ouvinte (Huron., 2016). De alguma forma, a predominância de movimentos dentro da EFF, num sentido muito concreto (em sentido contrário ao movimento dos ponteiros do relógio, tal como pode ser visto na Figura 4.10), sugere um carácter unidireccional

no desenvolvimento harmónico da obra, cujo real cruzamento com as ideias de Huron carece de estudos posteriores.

Uma outra característica não totalmente clara é o papel da CI 6 e da forma como ela é tratada (tal como Newton põe em hipótese) na mudança efetiva de AH dentro do espaço de fases. De facto, em alguns momentos deste andamento, o tratamento da CI 6 parece ter um efeito na mudança de agregado (como o exemplo dado em §4.2.4) e é ainda mais relevante esta ideia quando se compara o número total de mudanças entre grupos (vinte e uma), visíveis de forma individual na Figura 4.9, ao centro, com o número total de CI 6 presentes no andamento (dezanove), que podem ser consultadas na Figura 4.2. Esta proximidade entre os dois valores aponta, por isso, para uma possível função desempenhada pela CI 6 na mudança de AH, podendo, naturalmente, ser uma mera coincidência deste andamento, mas que requer, por certo, estudo futuro.

### 4.3 Súpula

Ao longo deste Capítulo, foi possível averiguar-se a plausibilidade da utilização da esfera de fases desenvolvida para o estudo da sintaxe harmónica atonal através da análise do terceiro andamento de uma obra pertencente ao período do atonalismo livre de Schöenberg, intitulada *Sechs kleine klavierstück*. Todas as suas EH foram, portanto, dispostas no espaço de fases correspondente, separadas em três agregados distintos e, seguidamente, foram apresentados os principais resultados obtidos, bem como a sua interpretação à luz do entendimento que o sistema providencia, mas também com recurso a outras análises presentes na literatura.

Este processo permitiu identificar neste andamento, entre outros, um indicador sintático relevante derivado de um padrão de movimentação circular entre AH. Por outras palavras, a forma como o trajeto harmónico da obra evolui no tempo acontece, maioritariamente, no sentido contrário ao ponteiro dos relógios dentro do espaço de fases, o que sugere progressões harmónicas que esgotam parte ou mesmo o total cromático. Esta ideia de circularidade é, por isso, utilizada, ao longo do próximo capítulo, para ajuda à definição de uma sintaxe harmónica no processo criativo do compositor.

# 5. A EFF no Apoio à Composição

A utilidade da EFF no estudo da sintaxe harmónica em obras do período atonal foi avaliada no Capítulo 4. Contudo, é igualmente relevante perceber se ela poderá desempenhar um papel ao longo do processo criativo do compositor, ajudando-o a estabelecer uma organização harmónica temporal, ou seja, a definir encadeamentos de CA para as suas obras e que derivem das ideias refletidas anteriormente. Ora é exatamente este o objetivo mais premente do Capítulo 5.

Para isso optou-se por um método próprio de utilização do espaço esférico, mas, por certo, são possíveis várias outras abordagens dependendo do compositor ou da obra a escrever. É também de notar que, assumidamente, não é objetivo deste Capítulo 5 a utilização da EFF para um ato criativo que tente fazer, meramente, cópia estilística das obras do atonalismo livre e, portanto, ela foi utilizada para a escrita de uma pequena peça com características composicionais e linguagem musical próprias. Esta opção, aliada ao facto da esfera de fases apenas sugerir movimentações entre agregados de EH, cuja escolha efetiva, em última instância, fica sempre ao critério do compositor, é a garantia da sua completa liberdade no processo criativo com recurso ao espaço esférico de fases.

Com estes objetivos em mente é possível lerem-se neste capítulo duas secções. A primeira faz uma breve descrição da obra composta (§5.1) e, seguidamente, descreve-se a utilização da EFF no processo de escrita (§5.2).

## 5.1 Descrição Global

No âmbito desta dissertação foi decidido escrever uma pequena peça para um quinteto de cordas constituído por dois violinos, duas violas de arco e um violoncelo. Esta decisão apoiou-se, sobretudo, na enorme variedade tímbrica que estes instrumentos de cordas proporcionam, nomeadamente através de uma multiplicidade de técnicas típicas neles como *sul tasto*, *sul ponticello*, *con sordina*, *pizzicato*, entre outros. Esta variedade tímbrica é, de resto, um dos

Secção	I	II	I'	III	I''	III'
Compasso	1-3	4-21	22-24	25-32	33-36	37-45

**Figura 5.1:** A estrutura formal do quinteto de cordas escrito com recurso à EFF.

elementos centrais da peça e, por outro lado, é também uma forma de manter o discurso musical novo e fresco ao longo do tempo.

É importante também realçar que este quinteto não é, na verdadeira essência da palavra, uma obra atonal e, verdadeiramente, não tem como objetivo sê-lo, pois dessa forma, provavelmente seria uma obra de imitação estilística das peças do atonalismo livre. Isto acontece porque, muito embora se identifiquem no quinteto EH altamente dissonantes e por isso longínquas do universo da harmonia funcional, é possível identificar-se um centro tonal claro na classe de altura 10 (A# ou Bb), o que torna essa mesma classe de altura uma referência central ao longo do desenvolvimento da peça.

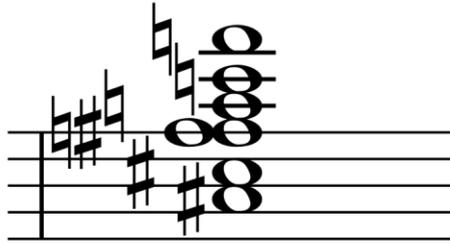
Quanto à sua estrutura formal, dos quarenta e cinco compassos que a constituem, distinguem-se um total de 6 pequenas secções, sendo algumas delas, ligeiras variações de anteriores. A Figura 5.1 mostra o esquema dessas secções e os compassos em que se iniciam e terminam.

A secção I, com três compassos e tocada *con sordina*, apresenta um primeiro gesto musical que se repetirá, em parte e integralmente, em mais duas secções, a I' e I''. Nela pode ver-se, no início do compasso 3, uma das EH que inspiraram grande parte da escrita deste andamento e que melhor explico em 5.2.1. A secção II tem somente uma aparição e caracteriza-se por uma textura que se assemelha a uma lenta metamorfose harmónica obtida através de vários *glissandi* nos diferentes instrumentos de cordas. Finalmente, a secção III, cujo material musical se ouve duas vezes, caracteriza-se por uma breve melodia dobrada à oitava e acompanhada por blocos de acordes cujo encadeamento é derivado da EFF. Em 5.2, poder-se-á perceber com maior detalhe o processo de escrita da peça.

A partitura completa de todo o quinteto pode ser consultada no [Anexo 3](#).

## 5.2 O Método Escolhido

À luz do que foi discutido ao longo do Capítulo 4, optou-se pelo focar na hipótese da circularidade em sentido contrário ao do ponteiro dos relógios entre AH dentro do espaço esférico de fases para a definição de um processo de composição que nele se apoia. Esta ideia implica, no essencial, que o compositor, previamente, escreva um conjunto de informação harmónica com a



## 19

**Figura 5.2:** EH escolhida que sobrepõe dois acordes maiores à distância de  $1/2t$ .

qual está interessado em trabalhar e, de seguida, a projete no espaço de fases para que dessa forma seja separada em diferentes AH e definida a distância relativa entre todos os pontos no espaço. Finalmente, com essa informação, cabe ao compositor fazer as escolhas composicionais que mais lhe apraz, sempre com a ideia de seqüências harmónicas que percorram o espaço num sentido concreto. Com isso em mente são referidos, de seguida, de forma mais pormenorizada, os processos: de seleção do material composicional (§5.1.1), de definição dos AH (§5.1.2) e de composição (§5.1.3) que guiaram a criação de uma pequena obra musical com recurso à EFF.

### 5.2.1 Seleção do material harmónico

O primeiro passo para a escrita da peça foi a seleção do material harmónico a dispor no espaço de fases. Para isso, num ficheiro do *software* de notação musical *Sibelius* foram escritos uma série de tri, tetra, penta e heptacordes, bem como alguns *clusters*<sup>11</sup> com os quais se pretendia trabalhar ao longo da obra para duas secções distintas, num total de trinta e uma EH diferentes. A sua escolha não foi, de todo, aleatória, uma vez que, logo à partida, existiu a preocupação de escrever CA que partilhassem registos próximos e, assim, pudessem garantir encadeamentos com conduções melódicas coerentes e que tivessem, ainda, estruturas intervalares lógicas e previamente pensadas. Por exemplo, o heptacorde número 19, que se pode ver na Figura 5.2, é constituído por dois acordes maiores sobrepostos à distância de  $1/2t$  (FM e F#M). De seguida, esse material foi exportado em formato *.XML* e submetido a todo o processo descrito no Capítulo 3 para a sua disposição no espaço esférico de fases. Um dos pontos chave desse processo é a contagem das CI mais presentes no material harmónico escolhido e, por essa razão, pode ver-se na Figura 5.3 que as CI mais frequentes são a 1 (referente a intervalos de 2<sup>a</sup>m ou 7<sup>a</sup>M) e a 5 (para intervalos de 4<sup>a</sup>P ou 5<sup>a</sup>P), resultando, por isso, num espaço de fases  $\varphi_{1/5}$ .

O conjunto de toda essa informação harmónica escolhida nesta fase e a sua respetiva numeração podem ser consultados no [Anexo 4](#).

<sup>11</sup> O termo *cluster* é utilizado para a referência a CA com mais de 8 notas e em que a sua estrutura é bastante densa.

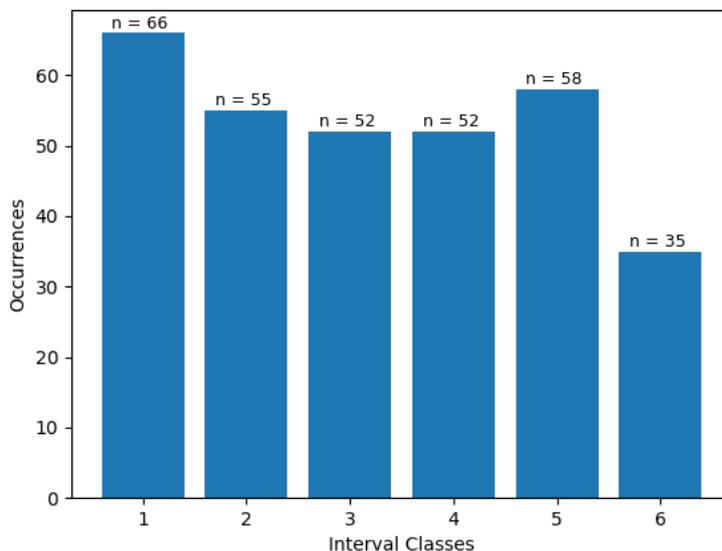


Figura 5.3: A contagem de CI presentes em todo o material harmónico escolhido.

### 5.2.2 Definição dos agregados harmónicos

Depois de calculado o valor de fases através da TDF referente às componentes  $\hat{f}_1$  e  $\hat{f}_5$  para todos os trinta e um CA escolhidos foi possível agrupá-las em três agregados distintos, cujos resultados podem ser vistos na Tabela 5.1.

Note-se que, agora, ao contrário do que acontecia na análise da obra de Schönberg, estes resultados não são representativos de uma sequência harmónica pensada, mas, somente, de um conjunto de material harmónico previamente pensado de forma individual, cuja sequência e encadeamento se pretende inferir através da EFF. Do agrupamento visível na Tabela 5.1 resultaram dez EH para o agregado A, doze para o B e nove referentes ao C. Para além disso, tal como é visível na Figura 5.4, o posicionamento dos três agregados no espaço de fases  $\varphi_{1/5}$  requer que, para movimentações entre EH feitas em sentido contrário ao ponteiro dos relógios, se verifiquem encadeamentos do tipo [ABC], [BCA] ou [CAB]. Eles foram a base do processo de

Tabela 5.1: A correspondência entre todas as EH escolhidas e o seu respetivo AH.

Correspondência AH/EH

Agregado	Número da EH
A	1, 4, 7, 14, 16, 20, 25, 27, 30, 31
B	2, 3, 5, 6, 8, 12, 13, 15, 17, 18, 24, 29
C	9, 10, 11, 19, 21, 22, 23, 26, 28



## A EFF no Apoio à Composição

The image displays a musical score for a string quintet, focusing on measures 2, 19, and 20. At the top, three chord diagrams are shown on a six-string guitar-like staff, labeled '2 B', '19 C', and '20 A'. Below these, the full score for Violino I, Violino II, Viola I, Viola II, and Violoncello is presented. The score includes dynamic markings such as *pp*, *f*, and *p*, and performance instructions like *s. t.*, *c. sor.*, *senza vib.*, and *al niente*. A red rectangular box highlights the harmonic material in measures 19 and 20 across all instruments, corresponding to the chord diagrams above.

**Figura 5.5:** A utilização do material harmónico escolhido. Nesta figura pode ver-se a utilização de uma trajetória do tipo [BCA] na abertura da peça através da utilização das EH 2, 19 e 20 presentes no material harmónico escolhido e que se pode consultar no [Anexo 4](#).

Ao longo da escrita da obra foi repetido este processo de criação de seqüências harmónicas (maioritariamente do tipo [BCA]), mas optou-se também por uma utilização livre desses mesmos encadeamentos, sendo eles, conseqüentemente, uma opção composicional que poderia ser muito diferente se tomada para uma outra obra ou por um compositor distinto. Quer isto dizer que ao longo deste quinteto de cordas, nem todos as EH derivam da informação obtida do espaço de fases. Por exemplo, na secção entre os compassos 4 e 16, escolheram-se quatro momentos distintos (nos compassos 6, 8, 10 e 14) em que, aí sim, as EH derivam da EFF e das suas movimentações. Esses momentos funcionam então, como âncoras harmónicas de uma secção que se pretende harmonicamente instável devido à presença dos vários *glissandi*. Dessa forma, todas as outras EH não pertencentes a esses pilares (muitas vezes de carácter microtonal por força desses mesmos *glissandi*) não derivam diretamente do agrupamento harmónico da EFF. Um outro exemplo em que a utilização dos AH não explica a totalidade da peça é entre os compassos 25 a 31. Neles encontramos uma textura de melodia acompanhada por blocos de acordes e, nela, somente esses blocos seguem movimentações [BCA], mais precisamente com os tetracordes numerados como 29, 28 e 27 (e que podem ser consultados no [Anexo 5](#)) constituídos pelas notas [D#, E, A, Bb], [C#, D, G, Ab] e [B, C, A, Bb], respetivamente. Dessa forma, a melodia presente

nesses compassos é completamente independente das EH que a acompanham e, por isso, não foi criada com recurso à EFF.

Para um melhor entendimento da totalidade dos processos utilizados ao longo da escrita, pode ver-se no [Anexo 3](#) a partitura completa deste quinteto de cordas, e no [Anexo 5](#) essa mesma partitura segmentada e comentada.

### 5.3 Súmula

Ao longo do Capítulo 5, foi explorada a utilidade da EFF para apoiar o compositor no seu processo criativo sugerindo-lhe uma forma de agrupar o material harmónico por si escolhido em diferentes agregados e, de seguida, para influenciar as opções harmónicas do compositor através da movimentação num sentido concreto dentro do espaço de fases (no caso desta obra em sentido contrário ao ponteiro dos relógios). Esta ideia pode, no limite, conferir maior coerência harmónica às obras escritas com recurso à EFF por estruturarem uma sintaxe harmónica baseada na ideia de circularidade dentro do espaço de fases, sem, com isso, retirar ao compositor a liderança do ato de escrita e continuando a colocar as suas ideias musicais no centro da atenção.

Nesse sentido, esta abordagem pareceu facilitar substancialmente o processo de escrita, sobretudo pelo facto de limitar as opções harmónicas à disposição (sem, contudo, as fazer escassear), e, por essa razão, tornando mais fluído o ato criativo. Outro aspeto de notar e que, por certo, requererá estudo futuro, é o facto da generalidade das sequências harmónicas que seguem o padrão no sentido contrário ao ponteiro dos relógios (nesta peça do tipo [BCA], por exemplo) resultarem, maioritariamente, em progressões satisfatórias e coerentes na avaliação do compositor. Esta característica constitui, por certo, um apoio de valor para o processo de composição com recurso a EH complexas, pois simplifica grandemente a procura por encadeamentos harmónicos em que elas estejam presentes.



## 6. Considerações Finais

Na música é possível encontrar-se um conjunto de funções dependentes do seu contexto que formam uma estrutura lógica na criação de sequências ou frases musicais inteligíveis ([Patel, 2010](#)). Com esta ideia, Patel refere-se à sintaxe harmónica que, tal como referido ao longo do Capítulo 2, sendo o conjunto de regras (muitas vezes subentendidas) que regulam a sucessão no tempo de EH distintas, é, para grande parte da música escrita depois do início do séc. XX, muito pouco entendida e sistematizada, contrariamente ao entendimento sintático de que já dispomos para a música anterior a esse período. Por essa razão, é importante que nela se busquem estruturas lógicas que permitam entender a forma como são construídos encadeamentos harmónicos em que a variedade de conteúdo intervalar de cada um dos seus constituintes é de enorme amplitude. Este é, exatamente, o caso da música referente ao período pós tonal.

Para procurar estas repostas, o presente estudo, amplamente inspirado no trabalho desenvolvido por Amiot ([2018](#)) e Yust ([2015](#)) no que diz respeito, respetivamente, à TDF para CA e à utilização da informação de fase das suas componentes, propõe um novo espaço no qual é possível dispor na superfície de uma esfera todos os CA de uma determinada peça através do cálculo das fases para duas componentes da TDF e, com isso, agrupá-los em diferentes AH com o objetivo de detetar padrões no trajeto harmónico da obra e deles extrair resultados que possam sugerir lógicas de organização sintática. Este método é utilizado para a análise do terceiro andamento do Op.19 de Arnold Schönberg intitulado *Sechs Kleine Klavierstück*, referente ao período do atonalismo livre deste mesmo compositor, mas também para avaliar a pertinência da sua utilização em contexto de apoio ao processo criativo do compositor, nomeadamente através da sugestão de grupos de CA que possibilitam uma mais fácil criação de progressões harmónicas. Destas duas vertentes do estudo podem-se extrair as seguintes conclusões (§5.1), principais contributos (§5.2) e perspetivas de trabalho futuro (§5.3).

### 6.1 Conclusões

Uma das primeiras conclusões que se pode extrair no final deste estudo prende-se com o facto de se verificar bastante pertinência na utilização de um espaço de fases derivado da TDF

para o entendimento da harmonia na música do período pós-tonal. Através do trabalho de Yust (2015), essa pertinência é avaliada para a música de Schubert, mas fica agora mais claro que em obras consideradas atonais também é possível utilizar o espaço de fases para investigar o seu trajeto harmónico e, com isso, deduzir lógicas que nos aproximem de um melhor entendimento sintático da sua harmonia.

Para esse efeito, não é irrelevante a interpretação do conceito de distância relativa entre CA dentro do espaço de fases. Nesse sentido, foi possível verificar que o posicionamento de um determinado CA é dado pela melhor aproximação ao posicionamento de todas as classes de altura individuais que o constituem. Por outras palavras, quanto mais classes de altura duas EH partilharem, também mais próximas elas estarão no espaço de fases e, posicionar-se-ão também próximas dos posicionamentos das suas classes de altura. A validação desta hipótese é, de resto, fundamental, por estar na base de várias outras conclusões deste estudo.

Uma delas é o facto de se perceber a utilidade da identificação de AH distintos para todos os CA de uma peça. Estes agregados, calculados algoritmicamente e que retornam a melhor forma de agrupar a informação harmónica disposta no espaço de fases em três grupos distintos, na prática, e tendo em conta a interpretação da distância entre pontos discutida previamente, criam agregados em que as suas EH partilham, regra geral, várias classes de altura individuais. É de realçar que o cálculo destes agregados é amplamente permitido pela criação de uma representação esférica do espaço de fases, uma vez que é ela que torna o posicionamento de todas as EH da peça consentâneo com o carácter cíclico das fases. Por outras palavras, se este agrupamento fosse feito para uma representação bidimensional, como a da Figura 2.12, os resultados obtidos não seriam tão fidedignos quanto aqueles que se verificam com a representação esférica. Posto tudo isto, é essencialmente através destes agregados que se torna viável identificar padrões no trajeto harmónico de uma obra e, com isso, levantar hipóteses para a sua organização sintática.

Para além disto, uma ideia que os resultados avançam neste andamento, é a influência clara que classes de altura mais frequentes numa obra têm na definição das centroides dos diferentes AH. De alguma forma, os resultados para o terceiro andamento das seis pequenas peças para piano de Schönberg, mostraram que duas das centroides calculadas pelo algoritmo *k-means* (2020) se encontram posicionadas no espaço de fases muito próximas do posicionamento das duas classes de altura mais repetidas na peça e, portanto, tornam-se como um centro de atração de vários CA para esse agregado.

Indubitavelmente, a principal ideia para a organização sintática harmónica que foi possível verificar neste andamento, é a tendência para movimentações circulares dentro do espaço de fases (ao invés de movimentações com carácter mais zigzagueante), ou seja, a tendência para trajetos harmónicos que passam pelos três agregados e que, com isso, percorrem a totalidade da superfície da esfera num determinado sentido. Esta característica de movimentações circulares pode ser interpretada como progressões harmónicas que procuram a utilização do total cromático, ou seja, progressões onde se podem encontrar as doze classes de altura. Isto acontece porque EH que percorrem todo o espaço esférico tendem também a ser constituídas por classes de altura

individuais distintas e, portanto, preenchem todo o total cromático. Esta ideia não deixa de ser especialmente de notar uma vez que vai ao encontro da neutralidade tonal que os compositores procuravam especialmente neste período.

No entanto, outra evidência que emana dos resultados deste andamento e para a qual ainda se deve buscar interpretação, é a clara prevalência de movimentações circulares no sentido contrário ao dos ponteiros do relógio em detrimento do sentido contrário. Este aspeto é intrigante porque tendo em conta a ideia de preenchimento do total cromático referida anteriormente (e que pode ser alcançada através de movimentações em ambos os sentidos) seria de esperar um maior equilíbrio entre o seu número de ocorrências. No entanto, isso não se verifica, e poderá indicar uma mera coincidência para este andamento propriamente dito, que, se tivesse um número maior de EH, ou seja, uma amostra maior, poder-se-ia obter um equilíbrio entre os dois sentidos. Ao invés, uma outra explicação poderá ser que a predominância de movimentações em sentido contrário ao dos ponteiros do relógio indica um carácter unidirecional do discurso musical, tal como sugerido por Huron (2016).

Finalmente, foi possível verificar que a EFF e a ideia referida no parágrafo anterior, podem também ser úteis durante o processo de composição. Muito embora fossem possíveis outras abordagens para a escrita de uma obra com recurso à EFF, e salvaguardando que há muito trabalho futuro possível para a criação de um sistema de apoio ao compositor mais interativo, é claro que a disposição de material harmónico no espaço de fases pode acrescentar fluidez à escrita do compositor porque facilita largamente a criação de progressões harmónicas com CA complexos do ponto de vista intervalar.

## 6.2 Contributos

De entre os principais contributos que esta dissertação avança destacam-se 6 ideias distintas. A primeira tem que ver com o estudo da música atonal, mais concretamente, da sua sintaxe harmónica. Este aspeto é relevante porque a sintaxe harmónica atonal tem ainda um caminho a percorrer para o seu completo conhecimento, e, neste trabalho, dá-se um pequeno passo nessa direção.

Em segundo lugar, foi proposta a hipótese da utilização das CI mais preponderantes numa obra para, através delas, definir as duas componentes da TDF que estarão na base do espaço de fases utilizado para a análise harmónica. Esta ideia resulta do facto de o espaço de fases  $\varphi_{3/5}$  (utilizado para o estudo de obras do período romântico por Yust), poder não ser o mais apropriado para a análise harmónica de obras atonais uma vez que as suas EH são tendencialmente mais complexas e variáveis de peça para peça. Desta forma, ao invés de utilizar as componentes com maior magnitude da TDF calculada para uma sonoridade representativa da obra cuja escolha seria sempre feita através de critérios questionáveis e diferentes de analista para analista, é utilizado o total das CI na peça para que, com as duas principais, seja definido o espaço de fases em concreto.

Para além disto, é inquestionável que uma das mais importantes contribuições que se podem ler nas páginas desta dissertação foi o desenvolvimento de um novo EGRH que dispõe à superfície de uma esfera os diferentes CA de uma peça. Esta ideia, portanto, estende o espaço de fases proposto por Yust (2015), no sentido de permitir uma melhor comparação da distância relativa entre pontos através da disposição da informação de fase (cujo carácter cíclico é conhecido) num espaço tridimensional que melhora a aproximação entre pontos face a uma representação bidimensional. Este último fator é, na verdade, quem possibilita a criação algorítmica de AH. Eles foram fundamentais para a deteção de movimentos padronizados entre eles e, a partir daí, para poderem ser deduzidas conclusões de organização sintática quanto às trajetórias identificadas para este tipo de música. Somando as ideias anteriores, com este trabalho desenvolveu-se também uma nova ferramenta multimédia que não só representa no espaço esférico de fases a informação harmónica pretendida como também possibilita a sua audição através da reprodução do seu áudio.

Finalmente, a possibilidade de utilização desta ferramenta multimédia baseada na informação de fase da TDF ao longo do processo criativo do compositor, nomeadamente assistindo-o na criação de progressões harmónicas para as suas obras é, também, um contributo a assinalar que este estudo propõe.

### 6.3 Perspetivas Futuras

No âmbito da sintaxe harmónica atonal e da utilização de espaços geométricos derivados da informação de fase da TDF há ainda um caminho longo para esgotar o seu estudo e, como não poderia deixar de ser, esta dissertação levanta tantas perspetivas de trabalho futuro quantas conclusões formula efetivamente. Uma dessas hipóteses para estudo subsequente é conseguir expandir os resultados da análise a um conjunto de obras mais alargado e que possa ser representativo não só do período do atonalismo livre, mas, também, de outros períodos estéticos posteriores. Esta possibilidade requer, contudo, uma base de dados de partituras em formato *.XML* de uma grande quantidade de obras referentes ao período pós-tonal, o que pode representar um entrave para a sua viabilidade. No entanto, seria bastante útil a análise de um conjunto maior de obras porque permitiria, por certo, levantar mais questões a propósito da sua organização sintática, generalizar resultados para um determinado período estético e, por fim, confrontar os resultados entre diferentes períodos e avaliar a existência de pontos de contacto entre si.

Ainda assim, esta ideia requer um sistema mais autónomo na perceção do material harmónico do que aquele que foi desenvolvido até este momento. Por outras palavras, seria útil desenvolver a EFF para identificar automaticamente os momentos de uma partitura em que é mais importante analisar a sua harmonia porque nem sempre o ritmo harmónico é coincidente com o ritmo melódico. No entanto, no âmbito da música do séc. XX, esta é uma tarefa de difícil implementação uma vez que a variedade métrica e a multiplicidade de alternativas para o ritmo harmónico são um entrave à criação de um sistema computacional sensível a estas ideias. Ainda

neste âmbito, melhorar as ferramentas de deteção de padrões de forma a que pudessem, por exemplo, identificar automaticamente o sentido concreto do movimento padronizado entre agregados e, simultaneamente, dispor o conteúdo intervalar das EH em análise, seria muito útil para perceber de que forma é que cada CI é tratada pelo compositor. Este último ponto permitiria, por exemplo, investigar o papel da resolução da CI 6 para a CI 4 tal como descrita no Capítulo 4, na mudança de AH. Esta ideia avançada por Newton (2013) requer, de facto, estudo futuro porque em alguns momentos deste andamento este movimento entre CI resultou, de facto, numa inflexão do discurso harmónico e mudança de agregado.

Também é de especial importância olhar no futuro para a predominância de movimentos padronizados dentro da EFF em sentido contrário ao dos ponteiros do relógio. Este aspeto é especialmente intrigante porque, para além da ideia de preenchimento do total cromático, sugere um outro aspeto de organização sintática ainda desconhecido. É de especial interesse, por exemplo, perceber de que forma este fenómeno poderá estar associado à ideia proposta por Huron (2016), de que a música transmite a sensação de empurrar o ouvinte em frente no tempo, através da realização das expectativas que inconscientemente vai criando. Este carácter unidirecional, que origina progressões harmónicas desprovidas de sentido quando tocadas em sentido contrário ao sentido original, deve também, no futuro, ser avaliado como uma possível explicação para a predominância de movimentações num sentido específico dentro da EFF.

Outra hipótese para qual seria produtivo atentar no futuro, é a ponderação da criação de um EGRH que unifique as informações de fase e de magnitude derivadas da TDF para CA. Sendo o espaço de fases sensível às inversões de acordes e moldado pela proximidade às classes de altura individuais que fazem parte de um determinado CA, e, por outro lado, sendo a informação de magnitude não sensível às transposições e fornecendo ela uma indicação do conteúdo intervalar de uma dada EH, é evidente que ambas as informações são complementares e essenciais ao entendimento sintático das obras do período pós-tonal. Como tal, seria de todo o interesse a reflexão num único espaço geométrico que possa unificar as duas grandezas e permitir a avaliação de trajetórias harmónicas em que ambas as informações possam estar presentes.

Por fim, há, por certo, bastante caminho a explorar no que à utilização da EFF como assistente à composição diz respeito. Em primeiro lugar, seria relevante perceber como é que outros compositores a utilizariam no seu processo criativo e, dessa forma, poder-se-ia refletir sobre que transformações são necessárias na ferramenta para melhor responder às necessidades levantadas por eles. Um desses caminhos, por certo, poderá passar por tornar a aplicação mais interativa ao ponto de permitir ao compositor, em tempo real, seleccionar uma qualquer EH a seu gosto e, automaticamente, o sistema propor um conjunto variável de progressões harmónicas entre as quais se poderá não só navegar e ouvir, mas também consultar o seu conteúdo intervalar e respetiva partitura.



# Referências Bibliográficas

Amiot, E. (2013). The Torii of Phases. *Mathematics and Computation in Music Lecture Notes in Computer Science*, 1-18. doi:10.1007/978-3-642-39357-0\_1

Amiot, E. (2016). Discrete Fourier Transform of Distributions. *Music Through Fourier Space Computational Music Science*, 1-26. doi:10.1007/978-3-319-45581-5\_1

Amiot, E. (2017). A Survey of Applications of the Discrete Fourier Transform in Music Theory. *Computational Music Science The Musical-Mathematical Mind*, 17-28. doi:10.1007/978-3-319-47337-6\_3

Amiot, E. (2018). *Music Through Fourier Space: discrete fourier transform in music theory*. SPRINGER.

Bernardes, G., Cocharro, D., Caetano, M., Guedes, C., & Davies, M. E. (2016). A multi-level tonal interval space for modelling pitch relatedness and musical consonance. *Journal of New Music Research*, 45(4), 281–294. doi: 10.1080/09298215.2016.1182192

Bigo, L., Garcia, J., Spicher, A., & Mackay, W. E. (2013). PaperTonnetz. *CHI 13 Extended Abstracts on Human Factors in Computing Systems on - CHI EA 13*. doi: 10.1145/2468356.2479608

Boss, J. F. (2019). *Schoenbergs atonal music: Musical idea, basic image, and specters of tonal function*. Cambridge: Cambridge University Press.

Callender, C. (2007). Continuous Harmonic Spaces. *Journal of Music Theory*, 51(2), 277–332. doi: 10.1215/00222909-2009-004

Cohn, R. (1998). Introduction to Neo-Riemannian Theory: A Survey and a Historical Perspective. *Journal of Music Theory*, 42(2), 167. doi: 10.2307/843871

Extensible Markup Language (XML). (2020). Retirado de <https://www.w3.org/XML/>

Forte, A. (1980). *The structure of atonal music*. New Haven: Yale Univ. Press.

Foundation, P. (2020). Processing.org. Retirado de <https://processing.org/>

Huron, D. (2008). *Sweet anticipation: music and the psychology of expectation*. Cambridge, MA: MIT.

Huron, D. B. (2016). *Voice leading: The science behind the musical art*. Cambridge, MA: MIT Press.

Jackendoff, R. (2002). *Foundations of language: how language connects to the brain, the world, evolution, and thinking*. New York, NY: Oxford University Press.

Kandinsky, W. (2009). *Concerning the spiritual in art*. The Floating Press.

Kostka, S., & Santa, M. (2018). *Materials and techniques of post-tonal music*. New York: Routledge, Taylor & Francis Group.

Krumhansl, C. L., & Shepard, R. N. (1979). Quantification of the hierarchy of tonal functions within a diatonic context. *Journal of Experimental Psychology: Human Perception and Performance*, 5(4), 579-594. doi:10.1037/0096-1523.5.4.579

Krumhansl, C. L., Bharucha, J. J., & Kessler, E. J. (1982). Perceived harmonic structure of chords in three related musical keys. *Journal of Experimental Psychology: Human Perception and Performance*, 8(1), 24-36. doi: 10.1037/0096-1523.8.1.24

Krumhansl, C. L., & Kessler, E. J. (1982). Tracing the dynamic changes in perceived tonal organization in a spatial representation of musical keys. *Psychological Review*, 89(4), 334-368. doi: 10.1037/0033-295x.89.4.334

Krumhansl, C. L. (1992). Internal representations for music perception and performance. *Cognitive Bases of Musical Communication*, 197-211. doi: 10.1037/10104-011

Large, E. W., Palmér, C., & Pollack, J. B. (1995). Reduced Memory Representations for Music. *Cognitive Science*, 19(1), 53-96. doi: 10.1207/s15516709cog1901\_2

Lerdahl, F., & Jackendoff, R. (1996). *A generative theory of tonal music*. Cambridge: The MIT Press.

Lewin, D. (1959). Re: Intervallic Relations between Two Collections of Notes. *Journal of Music Theory*, 3(2), 298. doi: 10.2307/842856

Lewin, D. (2001). Special Cases of the Interval Function between Pitch-Class Sets X and Y. *Journal of Music Theory*, 45(1), 1. doi: 10.2307/3090647

Martins, J. O. (2006). *Dasian, guidonian, and affinity spaces in twentieth-century music*. Tese de Doutoramento apresentada ao Departamento de Música da Universidade de Chicago.

Math Insight. (2020). Retirado de [https://mathinsight.org/cartesian\\_coordinates](https://mathinsight.org/cartesian_coordinates)

Messiaen, O. (1944). *Technique de mon langage musical*. Paris: A. Leduc.

Michels, U., Cardo, A. M., Doderer, G., Latino, A., & Vogel, G. (2001). *Atlas de música Vol. 2*. Lisboa: Gradiva.

MIDI Communication Protocol. (2020). Retirado de <http://www.ccarh.org/courses/253/handout/midiprotocol/>

Music21. (2020). Retirado de <http://web.mit.edu/music21/>

Newton, N. (2013). An Aspect of Functional Harmony in Schoenberg's Early Post-Tonal Music. *Music Analysis*, 33(1), 1-31. doi:10.1111/musa.12018

Patel, A. D. (2010). *Music, language, and the brain*. Oxford: Oxford University Press.

Perle, G. (1991). *Serial composition and atonality: An introduction to the music of Schoenberg, Berg and Webern*. Berkeley: University of California Press.

Quinn, I. (2006-7). General Equal-Tempered Harmony (in two parts). *Perspectives of new music* 44/2: 114-59; 45/1: 4-63.

Regras gerais de concordância. (2020). Retirado de <https://www.flip.pt/flip-online/gramatica/sintaxe/regras-gerais-de-concordancia>

Rohrmeier, M., & Pearce, M. (2018). Musical Syntax I: Theoretical Perspectives. *Springer Handbook of Systematic Musicology Springer Handbooks*, 473-486. doi:10.1007/978-3-662-55004-5\_25

Rosch, E. (1975). Cognitive reference points. *Cognitive Psychology*, 7(4), 532–547. doi:10.1016/0010-0285(75)90021-3

Schöenberg, A. (2010). *Style and idea: Selected writings of Arnold Schoenberg*. Berkeley: University of California Press.

Sethares, W. A. (2013). *Tuning timbre spectrum scale*. Place of publication not identified: Springer London.

Simms, B. R. (2000). *The atonal music of Arnold Schoenberg, 1908-1923*. New York: Oxford University Press.

Sklearn.cluster.KMeans¶. (2020). Retirado de <https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html>

Straus, J. N. (2016). *Introduction to post-tonal theory: 4th edition*. New York: W.W. Norton.

Welcome to Python.org. (2020). Retirado de <https://www.python.org/>

Yust, J. (2015). Schuberts Harmonic Language and Fourier Phase Space. *Journal of Music Theory*, 59(1), 121–181. doi: 10.1215/00222909-2863409

Yust, J. (2016). Special Collections – Renewing Set Theory. *Journal of Music Theory*, 60(2), 213-262. doi:10.1215/00222909-3651886

# **Anexos**



## **Anexo 1: Código**

## phaseSpace\_main.py

```
# phaseSpace_main.py
# Written in Python 3.7.

import phaseSpace_utilities as ps

def main():

    # to define the directory of the file
    ps.set_directory()

    # to read the .xml file and to transform it in chroma vectors.
    salami_sheet = ps.sheet_music_reader("skk_III", "option - print")

    # to create midi list and to export it.
    midi_list = ps.midi_list_creator(salami_sheet, "option - print")
    #ps.midi_list_exporter(midi_list, "V")

    # to transform data in pitch class vectors.
    pitch_class_dataset = ps.pitch_classes_creator(
        salami_sheet, "option - print", start = 1)
    pitch_classes_vector_dataset = ps.pitch_classes_vector_creator(
        pitch_class_dataset, "option - print")

    # to show the histograms.
    ps.pitch_classes_counter(salami_sheet)
    ps.interval_class_counter(salami_sheet)

    # to calculate fft, magnitudes and phases.
    fft_dataset_transformed = ps.fft_calculator(
        pitch_classes_vector_dataset, "option - print")
    magnitudes = ps.magnitude_calculator(
        fft_dataset_transformed, "option - print")
    phases = ps.phases_calculator(
        fft_dataset_transformed, "option - print")

    # to export phases data.
    #ps.phases_exporter(phases, "III", 1, 5)

    # to create random colors for each chord of the piece.
    colors = ps.color_definition(phases)

    # to define the window size where the results will appear.
    ps.window_size(15, 15)
```

```
# to show all the chords as points.
ps.plot_in_all_phase_spaces(
    phases, magnitudes, phase_scope=[1,2,3,4,5,6], c=colors)
ps.plot_just_one_phase_space(
    phases, magnitudes, 4, 5, c=colors, text=False, start=0)

# to show the chords path.
ps.path_in_all_phase_spaces(
    phases, magnitudes, phase_scope=[1,2,3,4,5,6], c=colors)
ps.path_just_one_phase_space(phases, magnitudes, 4, 5, c=colors)

# to print the required graphs.
ps.show_graphs()

if __name__ == "__main__":
    main()
```

## phaseSpace\_utilities.py

```
# phaseSpace_utilities.py
# All the functions for phaseSpace_main.py are defined in this file.
# Written in Python 3.7.

import os
import pathlib
import scipy.fftpack
import numpy as np
import music21
from itertools import combinations as cbn
from matplotlib import pyplot as plt

#1) Function to define the directory of the file:
def set_directory():
    os.chdir(pathlib.Path(__file__).parent.absolute())

#2) Function that reads all the chords in one .xml file.
def sheet_music_reader(piece_name, print_option=""):
    xml = piece_name
    sheet = music21.corpus.parse(xml)
    salami_sheet = sheet.chordify()
    count = 1
    for each_chord in salami_sheet.recurse().getElementsByClass('Chord'):
        each_chord.addLyric(str(count))
        count += 1
    if print_option == "print":
        salami_sheet.show()
    elif print_option == "":
        pass
    else:
        pass
    return salami_sheet

#3) Function that creates the MIDI list.
def midi_list_creator(salami_sheet, print_option=""):
    pitches_list = []
    midi_list = []
    chords_to_process = salami_sheet.recurse().getElementsByClass(
        'Chord')
    for each_chord in chords_to_process:
        pitches_list.append(each_chord.pitches)
    for i in range(len(pitches_list)):
        midi_list.append([])
```

```

        for j in range(len(pitches_list[i])):
            midi_list[i].append(pitches_list[i][j].midi)
if print_option == "print":
    print(midi_list)
elif print_option == "":
    pass
else:
    pass
return midi_list

```

#4) Function that exports MIDI list.

```

def midi_list_exporter(midi_list, movement):
    with open("midi_list_" + str(movement) +
        "_movement" + ".txt", 'w') as filehandle:
        for listitem in midi_list:
            filehandle.write('%s\n' % listitem)

```

#5) Function that creates the pitch classes data structure.

```

def pitch_classes_creator(
    salami_chords, print_option="", start=0, end=None):
    pitch_class_dataset = []
    chords_to_process = salami_chords.recurse().getElementsByClass(
        'Chord')
    if end is None:
        end = len(chords_to_process)
    if start > end:
        if end == len(chords_to_process):
            start = 0
        else:
            tmp = start
            start = end
            end = start
    for each_chord in chords_to_process:
        if start <= int(each_chord.lyrics[0].text) <= end:
            pitch_class_dataset.append(each_chord.normalOrder)
    for i in range(len(pitch_class_dataset)):
        pitch_class_dataset[i].sort()
    if print_option == "print":
        print(pitch_class_dataset)
    elif print_option == "":
        pass
    else:
        pass
    return pitch_class_dataset

```

#6) Function that transforms the pitch classes in chroma vectors.

```

def pitch_classes_vector_creator(pitch_classes, print_option=""):

```

```

pitch_class_vector_dataset = []
for i in range(len(pitch_classes)):
    pitch_class_vector_dataset.append([])
    for k in range(12):
        pitch_class_vector_dataset[i].append(0)
        if k == 11:
            break
    for j in range(len(pitch_classes[i])):
        pitch_class_vector_dataset[i][pitch_classes[i][j]] = 1
if print_option == "print":
    print(pitch_class_vector_dataset)
elif print_option == "":
    pass
else:
    pass
return pitch_class_vector_dataset

```

#7) Function that prints the pitch classes histogram.

```

def pitch_classes_counter(salami_sheet):
    salami_sheet.plot("histogram", "pitchClass")

```

#8) Function that prints the interval classes histogram.

```

def interval_class_counter(salami_sheet):
    interval_vector_list = []
    interval_vector_sum = []
    interval_vector_labels = []
    x = [1, 2, 3, 4, 5, 6]
    for each_chord in salami_sheet.recurse().getElementsByClass(
        'Chord'):
        interval_vector_list.append(each_chord.intervalVector)
    interval_vector_sum = [sum(i) for i in zip(
        *interval_vector_list)]
    for i in range(len(interval_vector_sum)):
        interval_vector_labels.append(
            "n = " + str(interval_vector_sum[i]))
    plt.bar(x, interval_vector_sum, align = 'center')
    plt.xlabel('Interval Classes')
    plt.ylabel('Occurrences')
    for i in range(len(x)):
        plt.text(x = x[i]-0.3, y = interval_vector_sum[i] +
            max(interval_vector_sum)/100, s = interval_vector_labels[i],
            size = 9)

```

#9) Function that calculates the fft of pitch class sets.

```

fft_dataset = []
fft_dataset_transformed = []
fft_dataset = scipy.fftpack.fft(chroma_vectors)

```

```

for fft_vector in fft_dataset:
    fft_dataset_transformed.append([complex(
        np.real(fft_vector[index])/np.real(
            fft_vector[0]), np.imag(fft_vector[index])/np.real(
                fft_vector[0])) for index in range(1,7)])
if print_option == "print":
    print(fft_dataset_transformed)
elif print_option == "":
    pass
else:
    pass
return fft_dataset_transformed

```

#10) Function that calculates the magnitudes for the FFT elements.

```

def magnitude_calculator(fft_dataset, print_option=""):
    magnitudes = np.absolute(fft_dataset)
    if print_option == "print":
        print(magnitudes)
    elif print_option == "":
        pass
    else:
        pass
    return magnitudes

```

#11) Function that calculates the phases for the FFT elements.

```

def phases_calculator(fft_dataset, print_option=""):
    phases = np.angle(fft_dataset)
    if print_option == "print":
        print(phases)
    elif print_option == "":
        pass
    else:
        pass
    return phases

```

#12) Function that exports phase information.

```

def phases_exporter(phases, movement, print_1, print_2):
    new_list = []
    for i in range(len(phases)):
        new_list.append([])
        new_list[i].append(phases[i][print_1 - 1])
        new_list[i].append(phases[i][print_2 - 1])
    with open("phases_skk_" + str(movement) + "_" + str(
        print_1) + "_" + str(
            print_2) + ".txt", 'w') as filehandle:
        for listitem in new_list:
            filehandle.write('%s\n' % listitem)

```

```

#13) Function that generates one random color for each chord.
def color_definition(phases):
    colors = [np.random.rand(3,) for i in range(len(phases))]
    return colors

#14) Function that defines the size of the window.
def window_size(x, y):
    plt.rcParams['figure.figsize'] = (x, y)

#15) Function that shows the chords as points.
def plot_in_all_phase_spaces(
    phases, magnitudes, phase_scope=[1, 2, 3, 4, 5, 6], c='b'):
    spaces = list(cbn(phase_scope, 2))
    fig, axarr = plt.subplots(5, 6, sharex=True, sharey=True)
    plt.xlim(-3.5, 3.5)
    plt.ylim(-3.5, 3.5)
    for space in spaces:
        xs = []
        ys = []
        x, y = space
        colors = []
        for i in range(len(phases)):
            if magnitudes[i][
                x-1] != 0 and magnitudes[i][y-1] != 0:
                xs.append(phases[i][x-1])
                ys.append(phases[i][y-1])
                if len(c) > 1:
                    colors.append(c[i])
            else:
                colors.append(c)
        axarr[x - 1, y - 1].scatter(xs, ys, c=colors)
        axarr[x - 1, y - 1].set_title(
            'phi_{x} vs. phi_{y}'.format(x, y))

#16) Function that shows the path of chords.
def path_in_all_phase_spaces(
    phases, magnitudes, phase_scope=[1, 2, 3, 4, 5, 6], c='b'):
    spaces = list(cbn(phase_scope, 2))
    fig, axarr = plt.subplots(5, 6, sharex=True, sharey=True)
    plt.xlim(-3.5, 3.5)
    plt.ylim(-3.5, 3.5)
    for space in spaces:
        xs = []
        ys = []
        x, y = space
        colors = []

```

```

for i in range(len(phases)):
    if magnitudes[i][x-1] != 0 and magnitudes[i][
        y-1] != 0:
        xs.append(phases[i][x-1])
        ys.append(phases[i][y-1])
        if len(c) > 1:
            colors.append(c[i])
        else:
            colors.append(c)
xs = np.array(xs)
ys = np.array(ys)
axarr[x - 1, y - 1].quiver(xs[:-1], ys[:-1], xs[
    1:] - xs[:-1], ys[1:] - ys[:-1], color=colors,
    scale_units='xy', angles='xy', scale=1)
plt.title('phi_{x} vs. phi_{y} phase space'.format(x, y))

```

#17) Function that shows the chords as points.

```

def plot_just_one_phase_space(phases, magnitudes, space_x,
space_y, c='b', text=False, start=0):
    fig, _ = plt.subplots()
    xs = []
    ys = []
    colors = []
    for i in range(len(phases)):
        if magnitudes[i][space_x-1] != 0 and magnitudes[i][
            space_y-1] != 0:
            xs.append(phases[i][space_x-1])
            ys.append(phases[i][space_y-1])
            if len(c) > 1:
                colors.append(c[i])
            else:
                colors.append(c)
        if text:
            plt.text(xs[-1], ys[-1], str(start +
                i), color=colors[-1])
    if not text:
        plt.scatter(xs, ys, c=colors)
    plt.xlim(-3.5, 3.5)
    plt.ylim(-3.5, 3.5)
    plt.title('phi_{x} vs. phi_{y} phase space'.format(
        space_x, space_y))

```

#18) Function that shows the path of chords.

```

def path_just_one_phase_space(phases, magnitudes, space_x,
space_y, c='b'):
    fig, _ = plt.subplots()
    xs = []

```

```

ys = []
colors = []
for i in range(len(phases)):
    if magnitudes[i][space_x-1] != 0 and magnitudes[i][
        space_y-1] != 0:
        xs.append(phases[i][space_x-1])
        ys.append(phases[i][space_y-1])
        if len(c) > 1:
            colors.append(c[i])
        else:
            colors.append(c)
xs = np.array(xs)
ys = np.array(ys)
plt.quiver(xs[:-1], ys[:-1], xs[1:] - xs[:-1], ys[1:] -
ys[:-1], color=colors, scale_units='xy', angles='xy',
scale=1, width = 0.004, headlength = 4)
plt.xlim(-3.5, 3.5)
plt.ylim(-3.5, 3.5)
plt.title('phi_{ } vs. phi_{ } phase space'.format(
    space_x, space_y))

```

#19) Function that prints all the required graphs.

```

def show_graphs():
    plt.show()

```

## spherical\_phase\_space.pyde

```
# spherical_phase_space.pyde

add_library('sound')
add_library('peasycam')

import sps_utilities as u
import draw_utilities as d
import midi_utilities as m

# to define the radius (r) of the sphere
# the amount of points in every sphere line (total),
# and the number of chords to represent simultaneously (n);
r = 700
total = 40
n = 7
point_counter = 0

# to read all the files needed to create the rpresentation;
f_phases = "phases_skk_III_4_5.txt"
f_pitch_classes = "phases_cs_4_5.txt"
f_centers = "cluster_center_III_movement.txt"
f_midi = "midi_list_III_movement.txt"
f_labels = "labels_III_movement.txt"

# to calculate point coordinates that will constitute the sphere;
points_sphere = u.points_sphere(r, total)

# to read phases values and to calculate chord coordinates;
points_rads = u.points_rads(f_phases)
points_cartesian = u.points_cartesian(r, points_rads)
pitches_rads = u.points_rads(f_pitch_classes)
pitches_cartesian = u.points_cartesian(r, pitches_rads)
# to import midi values for each chord in the piece
midi_list = m.midi_reader(f_midi)

# to read cluster_center coordinates
cluster_center = u.cluster_center(f_centers)
cluster_letters = ["A", "B", "C"]

# to determine the cluster number of each chord
labels_list = u.labels_reader(f_labels)
labels_without_repetition = []
```

```

# to determine the color of each chord;
color_list = u.color_generator()

# list to update values for path representation
# with structure [[s1, s2, s3], [t1, t2, t3], [c1, c2, c3]]
path_hud = [[1, 1, 1], [0, 0, 0], [0, 0, 0]]
update_values = [0.25, 255/(len(points_cartesian)/3), 1]

# list to update values for cluster movement representation
cluster_movement_combinations = [[0, 1, 0], [0, 1, 2], [0, 2, 0],
[0, 2, 1], [1, 0, 1], [1, 0, 2], [1, 2, 1], [1, 2, 0], [2, 0, 1],
[2, 0, 2], [2, 1, 0], [2, 1, 2]]
cluster_movement_counters = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
max_index = []

# to define list with individual pitch class names
pitches_names = ["C", "C#", "D", "D#", "E", "F", "F#", "G", "G#",
"A", "A#", "B"]

def setup():
    global cam
    fullscreen(P3D)
    framerate(60)
    textSize(r/20)
    cam = PeasyCam(this, r*3)
    cam.setMinimumDistance(r/2)
    cam.setMaximumDistance(r*5)

def draw():
    lights()
    background(255)

    # 1. to draw all the points that constitute the sphere;
    d.draw_sphere(points_sphere)

    # 2. to draw the reference lines and points;
    d.draw_references(r, f_phases, pitches_cartesian,
pitches_names)

    # 3. to draw the cluster centers
    d.draw_cluster_centers(r, cluster_center, cluster_letters,
color_list, point_counter)

    # 4. to draw chords in the sphere
    # 4.1) to draw only the last n points and lines;
    d.draw_n_paths(r, n, points_rads, points_cartesian,

```

```

color_list, labels_list, point_counter)

# 5. to draw the analysis tools on heads-up display;
cam.beginHUD()
# 5.1) to draw software name;
d.software_name(r, f_phases)

# 5.2) to draw paths representation;
d.draw_paths_representation(r, cluster_center, color_list,
cluster_letters, point_counter, path_hud)

# 5.3) to draw cluster counter representation;
d.draw_cluster_counter_representation(r,
points_cartesian, labels_list, cluster_center,
cluster_letters, color_list, point_counter)

# 5.4) to draw cluster sequence;
d.draw_cluster_sequence(r, color_list, cluster_letters,
labels_without_repetition)

# 5.5) to draw preferred cluster movements;
d.draw_preferred_cluster_movements(r, cluster_letters,
color_list, cluster_movement_combinations,
cluster_movement_counters, max_index, labels_without_repetition)
cam.endHUD()

def keyPressed():
    global point_counter
    global max_index
    if point_counter <= len(points_cartesian) + 1:
        # 1. to define what happens when the UP key is pressed;
        if keyCode == UP and point_counter < len(points_cartesian) + 1:
            # 1.1) to update point_counter value;
            point_counter += 1

def mouseClicked():
    if point_counter > 0 and point_counter <= len(midi_list):
        for i in range(len(midi_list[point_counter - 1])):
            f = SoundFile(this, midi_list[point_counter - 1][i] + ".wav")
            f.play()

```

## sps\_utilities.py

```
# sps_utilities.py

def points_sphere(r, total):
    points_sphere = []
    for i in range(total):
        lon = map(i, 0, total, 0, TWO_PI)
        for j in range(total):
            points_sphere.append([])
            lat = map(j, 0, total, 0, PI)
            x = r * sin(lat) * cos(lon)
            y = r * sin(lat) * sin(lon)
            z = r * cos(lat)
            points_sphere[-1].append(x)
            points_sphere[-1].append(y)
            points_sphere[-1].append(z)
    return points_sphere

def points_rads(file_name):
    point_rads = []
    with open(file_name, 'r') as filehandle:
        for line in filehandle:
            currentPlace = line[:-1]
            point_rads.append(currentPlace)
    for i in range(len(point_rads)):
        point_rads[i] = point_rads[i].split(", ")
        for j in range(2):
            if j == 0:
                point_rads[i][j] = point_rads[i][j].replace("[", "")
            elif j == 1:
                point_rads[i][j] = point_rads[i][j].replace("]", "")
            point_rads[i][j] = float(point_rads[i][j])
    return point_rads

def points_cartesian(r, points_rads):
    points_rads_scaled = []
    points_cartesian = []
    for i in range(len(points_rads)):
        points_rads_scaled.append([])
        points_cartesian.append([])
        for j in range(2):
            if j == 0:
                points_rads_scaled[i].append(map(points_rads[i][j], -PI, PI, 0, TWO_PI))
```

```

        elif j == 1:
            points_rads_scaled[i].append(map(points_rads[i][
                j], -PI, PI, 0, PI))
    for k in range(3):
        if k == 0:
            points_cartesian[i].append(int(r * sin(
                points_rads_scaled[i][1]) * cos(points_rads_scaled[
                    i][0])))
        elif k == 1:
            points_cartesian[i].append(int(r * sin(
                points_rads_scaled[i][1]) * sin(points_rads_scaled[
                    i][0])))
        elif k == 2:
            points_cartesian[i].append(int(r * cos(
                points_rads_scaled[i][1]))))
    return points_cartesian

def cluster_center(file_name):
    cluster_center = []
    with open(file_name, 'r') as filehandle:
        for line in filehandle:
            currentPlace = line[:-1]
            cluster_center.append(currentPlace)
    for i in range(len(cluster_center)):
        cluster_center[i] = cluster_center[i].split(", ")
        for j in range(3):
            if j == 0:
                cluster_center[i][j] = cluster_center[i][
                    j].replace("[", "")
            elif j == 2:
                cluster_center[i][j] = cluster_center[i][
                    j].replace("]", "")
                cluster_center[i][j] = float(cluster_center[i][j])
                cluster_center[i][j] = int(cluster_center[i][j])
    return cluster_center

def labels_reader(file_name):
    cluster_center = []
    with open(file_name, 'r') as filehandle:
        for line in filehandle:
            currentPlace = line[:-1]
            cluster_center.append(currentPlace)
    for i in range(len(cluster_center)):
        cluster_center[i] = int(cluster_center[i])
    return cluster_center

def color_generator():

```

```
# set 5 different colors to represent each chord cluster
color_list = [[56, 0, 60], [233, 0, 82], [4, 245, 255], [
    0, 255, 133], [240, 232, 7]]
return color_list
```

## midi\_utilities.py

```
# midi_utilities.py
```

```
def midi_reader(file_name):
    midi_list = []
    with open(file_name, 'r') as filehandle:
        for line in filehandle:
            currentPlace = line[:-1]
            midi_list.append(currentPlace)
    for i in range(len(midi_list)):
        midi_list[i] = midi_list[i].split(", ")
        if len(midi_list[i]) == 1:
            midi_list[i][0] = midi_list[i][0].replace("]", "")
        for j in range(len(midi_list[i])):
            if j == 0:
                midi_list[i][j] = midi_list[i][j].replace("[", "")
            elif j == len(midi_list[i]) - 1:
                midi_list[i][j] = midi_list[i][j].replace("]", "")
    return midi_list
```

## draw\_utilities.py

```
# draw_utilities.py

# to draw all the points that constitute the sphere
def draw_sphere(points_sphere):
    strokeWeight(1.5)
    stroke(0)
    for i in range(len(points_sphere)):
        point(points_sphere[i][0],
              points_sphere[i][1],
              points_sphere[i][2])

# to draw the reference lines and points;
def draw_references(r, f_phases, pitches_cartesian, pitches_names):
    strokeWeight(1)
    stroke(0)
    line(0, 0, r - 25, 0, 0, r + 50)
    line(0, 0, -1 * (r - 25), 0, 0, -1 * (r + 50))
    line(r - 25, 0, 0, r + 50, 0, 0)
    line(-1 * (r - 25), 0, 0, -1 * (r + 50), 0, 0)
    line(0, r - 25, 0, 0, r + 50, 0)
    line(0, -1 * (r - 25), 0, 0, -1 * (r + 50), 0)
    textSize(r/30)
    textAlign(CENTER)
    fill(0)
    text("phi" + f_phases[17] + " -pi", 0, -10, r + 55)
    text("phi" + f_phases[17] + " pi", 0, 10, -1 * (r + 55))
    text("phi" + f_phases[15] + " 0", -1 * (r + 55), -10, 0)
    text("phi" + f_phases[15] + " pi/-pi", r + 60, -10, 0)
    text("phi" + f_phases[15] + " -pi/2", 0, r + 70, 0)
    text("phi" + f_phases[15] + " pi/2", 0, -1 * (r + 70), 0)
    textSize(r/24)
    textAlign(CENTER)
    fill(0)
    for i in range(len(pitches_cartesian)):
        text(pitches_names[i],
             pitches_cartesian[i][0],
             pitches_cartesian[i][1],
             pitches_cartesian[i][2])

# to draw the cluster centers
def draw_cluster_centers(r, cluster_center, cluster_letters,
                        color_list, point_counter):
    strokeWeight(r/40)
```

```

textSize(r/20)
textAlign(CENTER)
for i in range(len(cluster_center)):
    stroke(color_list[i][0], color_list[i][1], color_list[i][2])
    fill(color_list[i][0], color_list[i][1], color_list[i][2])
    point(cluster_center[i][0], cluster_center[i][1],
cluster_center[i][2])
    if point_counter == 0:
        text("cluster center " + cluster_letters[i],
            cluster_center[i][0],
            cluster_center[i][1] + (r/13),
            cluster_center[i][2])

# to draw only the last n points and lines in the spherical space
def draw_n_paths(r, n, points_rads, points_cartesian, color_list,
labels_list, point_counter):
    textAlign(LEFT)
    textSize(r/12)
    if 0 < point_counter <= n:
        for i in range(point_counter):
            fill(color_list[labels_list[i]][0],
                color_list[labels_list[i]][1],
                color_list[labels_list[i]][2])
            text(str(i+1),
                points_cartesian[i][0],
                points_cartesian[i][1] - r/40,
                points_cartesian[i][2])
            strokeWeight(r/85)
            stroke(color_list[labels_list[i]][0],
                color_list[labels_list[i]][1],
                color_list[labels_list[i]][2])
            point(points_cartesian[i][0],
                points_cartesian[i][1],
                points_cartesian[i][2])
            if i >= 1:
                strokeWeight(1)
                line(points_cartesian[i - 1][0],
                    points_cartesian[i - 1][1],
                    points_cartesian[i - 1][2],
                    points_cartesian[i][0],
                    points_cartesian[i][1],
                    points_cartesian[i][2])
    elif n < point_counter <= len(points_cartesian):
        for i in range(n):
            fill(color_list[labels_list[point_counter - 1 - i]][0],
                color_list[labels_list[point_counter - 1 - i]][1],
                color_list[labels_list[point_counter - 1 - i]][2])

```

```

text(str(point_counter - i),
     points_cartesian[point_counter - 1 - i][0],
     points_cartesian[point_counter - 1 - i][1] - r/40,
     points_cartesian[point_counter - 1 - i][2])
strokeWeight(r/85)
stroke(color_list[labels_list[point_counter - 1 - i]][0],
       color_list[labels_list[point_counter - 1 - i]][1],
       color_list[labels_list[point_counter - 1 - i]][2])
point(points_cartesian[point_counter - 1 - i][0],
      points_cartesian[point_counter - 1 - i][1],
      points_cartesian[point_counter - 1 - i][2])
#to draw the last n-1 lines
for i in range(n-1):
    strokeWeight(1)
    stroke(color_list[labels_list[point_counter - 1 - i]][0],
          color_list[labels_list[point_counter - 1 - i]][1],
          color_list[labels_list[point_counter - 1 - i]][2])
    line(points_cartesian[point_counter - i - 2][0],
         points_cartesian[point_counter - i - 2][1],
         points_cartesian[point_counter - i - 2][2],
         points_cartesian[point_counter - i - 1][0],
         points_cartesian[point_counter - i - 1][1],
         points_cartesian[point_counter - i - 1][2])
elif point_counter > len(points_cartesian):
    for i in range(len(points_cartesian)):
        fill(color_list[labels_list[i]][0],
            color_list[labels_list[i]][1],
            color_list[labels_list[i]][2])
        text(str(i+1), points_cartesian[i][0],
            points_cartesian[i][1] - r/40,
            points_cartesian[i][2])
        strokeWeight(r/85)
        stroke(color_list[labels_list[i]][0],
              color_list[labels_list[i]][1],
              color_list[labels_list[i]][2])
        point(points_cartesian[i][0],
             points_cartesian[i][1],
             points_cartesian[i][2])
# to draw the phase coordinates alongside the points;
textSize(r/20)
fill(0)
if point_counter <= len(points_cartesian):
    if point_counter > 0 and point_counter < 10:
        text("(" + str(round(points_rads[point_counter - 1][
            0], 2)) + ", " + str(round(points_rads[
            point_counter - 1][1], 2)) + ")",
            points_cartesian[point_counter - 1][0] + 32,

```

```

        points_cartesian[point_counter - 1][1] - 28,
        points_cartesian[point_counter - 1][2])
elif point_counter >= 10:
    text(" (" + str(round(points_rads[point_counter - 1][
        0], 2)) + ", " + str(round(points_rads[
        point_counter - 1][1], 2)) + ")",
        points_cartesian[point_counter - 1][0] + 65,
        points_cartesian[point_counter - 1][1] - 28,
        points_cartesian[point_counter - 1][2])

# to draw software name;
def software_name(r, f_phases):
    textSize(r/27)
    textAlign(LEFT)
    fill(0)
    text("spherical phase space",
        width/6.5,
        height/12)
    text("phi " + f_phases[15] + "/" + f_phases[17],
        width/6.5,
        height/12 + 50)
    text("v_0.2",
        width/6.5,
        height/12 + 100)

# to draw paths representation;
def draw_paths_representation(r, cluster_center, color_list,
cluster_letters, point_counter, path_hud):
    textSize(r/20)
    textAlign(CENTER)
    for i in range(len(cluster_center)):
        fill(color_list[i][0], color_list[i][1], color_list[i][2])
        if i == 0:
            text(cluster_letters[i], width/5, height/2 - 100)
        elif i == 1:
            text(cluster_letters[i], width/5 - 125, height/2 + 100)
        elif i == 2:
            text(cluster_letters[i], width/5 + 125, height/2 + 100)
    textSize(r/30)
    fill(0)
    if point_counter > 1:
        strokeWeight(path_hud[0][0])
        stroke(0, path_hud[1][0])
        line(width/5 - 15, height/2 - 110, width/5 - 125, height/2 + 65)
        if path_hud[1][0] != 0:
            text(str(path_hud[2][0]), width/5 - 95, height/2 - 15)
        strokeWeight(path_hud[0][1])

```

```

stroke(0, path_hud[1][1])
line(width/5 - 109, height/2 + 88, width/5 + 103, height/2 + 88)
if path_hud[1][1] != 0:
    text(str(path_hud[2][1]), width/5, height/2 + 120)
strokeWidth(path_hud[0][2])
stroke(0, path_hud[1][2])
line(width/5 + 125, height/2 + 65, width/5 + 15, height/2 - 110)
if path_hud[1][2] != 0:
    text(str(path_hud[2][2]), width/5 + 95, height/2 - 15)

# to draw cluster counter representation;
def draw_cluster_counter_representation(r, points_cartesian,
labels_list, cluster_center, cluster_letters, color_list,
point_counter):
    textSize(r/20)
    textAlign(RIGHT)
    if point_counter >= 0 and point_counter <= len(points_cartesian):
        list_I = []
        list_II = []
        list_III = []
        for i in range(point_counter):
            if labels_list[i] == 0:
                list_I.append("true")
            elif labels_list[i] == 1:
                list_II.append("true")
            elif labels_list[i] == 2:
                list_III.append("true")
        for i in range(len(cluster_center)):
            fill(color_list[i][0], color_list[i][1], color_list[i][2])
            if i == 0:
                text(cluster_letters[i] + " - " + str(len(list_I)),
                    width/1.285, height/5)
            elif i == 1:
                text(cluster_letters[i] + " - " + str(len(list_II)),
                    width/1.285, height/5 + 45)
            elif i == 2:
                text(cluster_letters[i] + " - " + str(len(list_III)),
                    width/1.285, height/5 + (2*45))
    else:
        list_I = []
        list_II = []
        list_III = []
        for i in range(len(points_cartesian)):
            if labels_list[i] == 0:
                list_I.append("true")
            elif labels_list[i] == 1:
                list_II.append("true")

```

```
elif labels_list[i] == 2:
    list_III.append("true")
for i in range(len(cluster_center)):
    fill(color_list[i][0], color_list[i][1], color_list[i][2])
    if i == 0:
        text(cluster_letters[i] + " - " + str(len(list_I)),
              width/1.285, height/5)
    elif i == 1:
        text(cluster_letters[i] + " - " + str(len(list_II)),
              width/1.285, height/5 + 45)
    elif i == 2:
        text(cluster_letters[i] + " - " + str(len(list_III)),
              width/1.285, height/5 + (2*45))
```

## kmeans.py

```
# kmeans.py

import kmeans_utilities as kmu
from sklearn.cluster import KMeans

def main():

    r = 700
    file_name = "phases_skk_III_4_5.txt"
    clusters_number = 3
    kmu.set_directory()
    point_rads = kmu.point_rads(file_name)
    points_cartesian = kmu.points_cartesian(point_rads, r)
    X = kmu.points_cartesian_array(points_cartesian)

    # to determine the optimal number of k
    sum_squared_distances = []
    K = range(1,7)
    for k in K:
        km = KMeans(n_clusters=k)
        km = km.fit(X)
        sum_squared_distances.append(km.inertia_)

    # Initializing KMeans
    kmeans = KMeans(n_clusters = clusters_number)
    kmeans = kmeans.fit(X)
    labels = kmeans.predict(X)
    C = kmeans.cluster_centers_

    #kmu.cluster_center_exporter(C, "III")
    #kmu.labels_exporter(labels, "III")

    kmu.plot_optimal_k(K, sum_squared_distances)
    kmu.plot_graph(X, C)

if __name__ == "__main__":
    main()
```

## kmeans\_utilities.py

```
# kmeans_utilities.py

import os
import pathlib
import math
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

def set_directory():
    os.chdir(pathlib.Path(__file__).parent.absolute())

def point_rads(file_name):
    point_rads = []
    with open(file_name, 'r') as filehandle:
        for line in filehandle:
            # remove linebreak which is the last character of the string
            currentPlace = line[:-1]
            # add item to the list
            point_rads.append(currentPlace)
    for i in range(len(point_rads)):
        point_rads[i] = point_rads[i].split(", ")
        for j in range(2):
            if j == 0:
                point_rads[i][j] = point_rads[i][j].replace("[", "")
            elif j == 1:
                point_rads[i][j] = point_rads[i][j].replace("]", "")
                point_rads[i][j] = float(point_rads[i][j])
    return point_rads

def points_cartesian(point_rads, r):
    points_rads_scaled = []
    points_cartesian = []
    for i in range(len(point_rads)):
        points_rads_scaled.append([])
        points_cartesian.append([])
        for j in range(2):
            # to define a range between 0 and 2*PI
            if j == 0:
                points_rads_scaled[i].append(point_rads[i][j] + math.pi)
            elif j == 1:
                points_rads_scaled[i].append((point_rads[i][j]
```

```

        j] + math.pi)/2)
for k in range(3):
    if k == 0:
        points_cartesian[i].append(int(r * math.sin(
            points_rads_scaled[i][1]) * math.cos(
                points_rads_scaled[i][0])))
    elif k == 1:
        points_cartesian[i].append(int(r * math.sin(
            points_rads_scaled[i][1]) * math.sin(
                points_rads_scaled[i][0])))
    elif k == 2:
        points_cartesian[i].append(int(r * math.cos(
            points_rads_scaled[i][1])))
return points_cartesian

def points_cartesian_array(points_cartesian):
    X = np.array(points_cartesian)
    return X

def cluster_center_exporter(cluster_list, movement):
    new_list = []
    for i in range(len(cluster_list)):
        new_list.append([])
        for j in range(3):
            new_list[i].append(cluster_list[i][j])
    with open("cluster_center_" + str(
        movement) + "_movement" + ".txt", 'w') as filehandle:
        for listitem in new_list:
            filehandle.write('%s\n' % listitem)

def labels_exporter(labels_list, movement):
    with open("labels_" + str(
        movement) + "_movement" + ".txt", 'w') as filehandle:
        for listitem in labels_list:
            filehandle.write('%s\n' % listitem)

def plot_optimal_k(K, sum_squared_distances):
    plt.plot(K, sum_squared_distances, 'bx-')
    plt.xlabel('k')
    plt.ylabel('Sum_of_squared_distances')
    plt.title('Elbow Method For Optimal k')
    plt.show()

def plot_graph(X, C):
    plt.rcParams['figure.figsize'] = (16, 9)
    fig = plt.figure()
    ax = Axes3D(fig)

```

```
ax.scatter(X[:, 0], X[:, 1], X[:, 2])
ax.scatter(C[:, 0], C[:, 1], C[:, 2],
marker='*', c='#050505', s=1000)
plt.show()
```

**Anexo 2: Partitura do 3º andamento do Op. 19  
de Arnold Schönberg**

Nota – edição própria no âmbito deste trabalho

## Sechs kleine Klavierstücke, Op. 19

### III.

Arnold Schoenberg - 1911

Sehr langsame 

Piano



*f*

*pp*

*pp*

2

*pp*

4

*p*

6

*pp*

*ppp*

## **Anexo 3: Partitura do quinteto de cordas**





♩ = 152

3

25

*nat.*  
*s. sor.* *simile*

*f p f p f p f p f*

*nat.*  
*s. sor.* *simile*

*f p f p f p f p f*

*pizz.*  
*s. sor.*

*f* *ff*

*pizz.*

*f* *ff*

♩ = 52

32

*ff* *pp* *f* *pp*

*ff* *pp* *f* *pp*

*arco* *poco vib.* *s. t.* *s. p.*

*ff* *pp* *f* *pp*

*arco* *s. t.* *s. t.* *s. p.*

*ff* *pp* *f* *pp*

4

**molto accel.**

$\text{♩} = 152$

Musical score for measures 36-40. The score consists of five staves. The first four staves are for string quartet parts (Violin I, Violin II, Viola, and Violoncello), and the fifth is the Bass line. Each string part begins with a *s.t.* (sul tasto) marking and a *pizz.* (pizzicato) instruction. The dynamics are marked as *mf* (mezzo-forte) and *f* (forte). The text *al niente* is written above the first four staves. The Bass line starts with *pp < p* (pianissimo to piano) and *pizz.* markings. The music features a complex rhythmic pattern with many sixteenth notes and rests.

Musical score for measures 41-45. The score consists of five staves. The first four staves are for string quartet parts, and the fifth is the Bass line. The first three staves have *ff* (fortissimo) dynamics. The last two staves have *fff* (fortississimo) dynamics. The text *arco* (arco) is written above the staves. The music features a complex rhythmic pattern with many sixteenth notes and rests, including triplets.

## **Anexo 4: Material Harmónico**

# Material Harmónico

## Secção A

### Tricordes:

Musical staff showing seven measures of chords. The chords are labeled 1 through 7 below the staff.

1 A    2 B    3 B    4 A    5 B    6 B    7 A

10

### Tetracordes:

Musical staff showing five measures of chords. The chords are labeled 8 through 12 below the staff.

8 B    9 C    10 C    11 C    12 B

17

### Pentacordes:

Musical staff showing six measures of chords. The chords are labeled 13 through 18 below the staff.

13 B    14 A    15 B    16 A    17 B    18 B

### Heptacordes:

Musical staff showing five measures of chords. The chords are labeled 19 through 23 below the staff.

19 C    20 A    21 C    22 C    23 C

### Clusters:

Musical staff showing three measures of chords. The chords are labeled 24, 25, and 26 below the staff.

24 B    25 A    26 C

## Secção B

38

Musical staff showing five measures of chords. The chords are labeled 27 through 31 below the staff.

27 A    28 C    29 B    30 A    31 A

## **Anexo 5: Partitura do quinteto de cordas com anotações**

# I. Quinteto para cordas

Escrito com recurso à EFF  
Junho 2020

Samuel Pereira

\*números são referência  
à numeração do ficheiro  
material harmónico

♩ = 52  
s. t.  
c. sor.  
senza vib.

2 19 20

s.p. s.t.

nat. 5  
s. sor.

Violino I

Violino II

Viola I

Viola II

Violoncelo

*pp* *f* *pp* *pp* *p*

al niente

al niente

al niente

al niente

senza vib.

senza vib.

senza vib.

senza vib.

senza vib.

nat. s. sor.

poco vib.\*

poco vib.\*

poco vib.\*

*p*

\*poco vibrato especifica um vibrato muito ligeiro  
\*\*molto vibrato especifica um vibrato com amplitude de sensivelmente 1/4 de tom

7

10 16

poco vib. senza vib.

molto vib.\*\* s.p. , senza vib.  
nat.

nat. s. sor.

molto vib.\*\* s.p. , senza vib.  
nat.

senza vib. gliss. ,

molto vib.\*\* s.p. , senza vib.  
nat.

senza vib. ,

molto vib.\*\* s.p. , senza vib.  
nat.

senza vib. ,

molto vib.\*\* s.p. , senza vib.  
nat.

senza vib. ,

molto vib.\*\* s.p. , senza vib.  
nat.

senza vib. ,

molto vib.\*\* s.p. , senza vib.  
nat.

*p* *mf* *p* *mf* *p* *mf* *p* *mf* *p* *mf* *p*





4

**molto accel.**

♩ = 152

36

*s.t.* *pizz.* *mf* *f*

*s.t.* *pizz.* *mp* *f*

*s.t.* *pizz.* *mf* *f*

*s.t.* *pizz.* *p* *mp*

*pizz.* *pp* *p*

29 28

41

27 29 28

*arco* *fff* *fff* *fff* *fff*

*fff* *fff* *fff* *fff* *fff*