

MuseCode: uma abordagem aritmética à música tonal

José Carlos Ramalho

University of Minho
Braga, Portugal

jcr@di.uminho.pt

Resumo Hoje em dia, a utilização de software musical para apoio à composição, à produção de partituras e à análise musical é incontornável. Por outro lado, as notações musicais além de complexas, diferem muito uma das outras; é muito difícil para não dizer impossível, garantir a interoperabilidade entre diferentes aplicações de manipulação de música. Apesar de haver algumas iniciativas, ainda não emergiu um formato como norma.

Houve ainda outras motivações na génese deste trabalho. Muitos dos exercícios que se fazem em teoria musical levam-nos a realizar cálculos algo complicados: cálculo de intervalos, construção de acordes, construção das mais variadas escalas, transposição de partituras para instrumentos transpositores, construção de cadências para jazz ou clássico, etc. A complexidade inerente a estes cálculos existe devido à não linearidade das notas na escala musical que todos conhecemos, as oito notas de uma oitava estão separadas por 5 intervalos de um tom e dois intervalos de meio tom que conforme a escala em questão aparecem em determinadas posições dessa escala. Estes dois intervalos de meio tom invalidam a possibilidade de se usar a nota musical como referência para cálculo, ou se o fizermos, vamos ter de andar a contar sempre os tons e os meio tons.

Este trabalho teve então como principal objectivo definir um referencial linear no qual fosse possível representar de forma unívoca todas as notas musicais com todas as possíveis alterações que estas possam a vir a ter (em música: sustenidos e bemóis). Este referencial deveria permitir realizar todos aqueles cálculos enunciados acima através de uma operação aritmética simples, como a adição ou a subtração.

1 Introdução

Quem estuda música, é, muitas vezes, confrontado com esta questão: *A música está nas notas ou entre as notas?*

Questão esta que representa um óptimo pontapé de saída para a motivação por detrás deste trabalho. Como noutras áreas e contextos nós somos sensíveis às variações. Assim, é simples concluir que para um ser humano a música se situa entre as notas e não nas notas. O que altera a importância relativa das coisas: o intervalo entre duas notas passa a ser mais importante que as notas em si. Por inerência o cálculo/identificação de um intervalo assume uma grande importância. Ora, o cálculo de um intervalo deveria ser uma operação muito simples, afinal é a distância entre dois pontos. Devia ser simples mas não é. Basta observar o tempo de resposta de um aluno de música. E não

é simples porque o referencial musical, ao contrário de muitos outros referenciais abstractos (por exemplo, os números inteiros) não é uniforme, os elementos que constituem o seu domínio de aplicação não estão à mesma distância uns dos outros.

Na figura 1, apresenta-se a escala normalmente usada como referência na música ocidental: a escala de dó maior. Esta escala, como todas as escalas maiores, tem sete intervalos. Cinco são intervalos de um tom, os outros dois de meio tom (estão assinalados na figura). Estes intervalos de meio tom para que uma escala maior soe como deve, ou como estamos habituados auditivamente, têm de estar situados entre a terceira e a quarta nota da escala e, entre a sétima e a oitava.

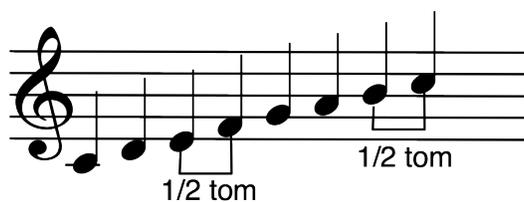


Figura 1: Escala modelo de dó maior

Normalmente, é esta a escala que é apresentada para introduzir conceitos e matérias como a construção de intervalos, construção de acordes, construção de outras escalas, transposição de música, etc. E como vamos estando habituados a esta escala, calcular intervalos a partir da nota dó é fácil. Mas se em vez de dó for a partir de fá já é um pouco mais complicado. E se mudarmos a tonalidade, se em vez de dó maior tivermos lá maior ou fá sustenido maior, a complexidade aumenta. E se formos para uma tonalidade menor, ou para um outro modo, ou ainda para o jazz, as coisas que eram tão simples na escala modelo deixam de ser simples e passam a complexas podendo estar na origem de muitos erros (se isto não acontecesse não haveria necessidade de fazer avaliação à teoria musical que aqui se está a discutir).

O problema que se está aqui a levantar é o de como criar um referencial uniforme para as notas musicais que nos permita raciocinar em termos de intervalos através de operações aritméticas muito simples como a adição e a subtração (coisa que já fazemos no conjunto dos números naturais, ou no dos inteiros, ou em qualquer outro domínio matemático abstracto). O exemplo apresentado é um pouco simples, pois a ideia é entender o objectivo à representação de qualquer nota mesmo que alterada (uma nota pode ser alterada meio tom para cima, com um sustenido, ou meio tom para baixo, com um bemol).

Na secção seguinte, descreve-se o processo que levou à criação do Musecode. A seguir, numa nova secção, apresentam-se algumas aplicações da teoria desenvolvida na resolução de alguns problemas. Por fim, numa secção final, tecem-se algumas conclusões e enumeram-se algumas aplicações informáticas que esta teoria torna possível de uma forma muito simples.

2 MuseCode

Estudaram-se várias notações musicais em uso hoje em dia no universo digital: MusicXML, MusicTeX, Lilypond, ABC. Cada uma delas tem a sua maneira de representar as notas, todas são complexas e difíceis de manipular (a música é algo difícil devido às várias dimensões em que se pode expandir). Mas uma delas, a MusicXML, deu a ideia de base.

2.1 Codificação das notas

Em MusicXML uma tonalidade é representado por um número inteiro. Esse número corresponde ao número de alterações que têm de ser introduzidas para manter os intervalos entre as notas como na escala modelo. Por exemplo, a escala apresentada na figura 1, de dó maior, não necessita de nenhuma alteração pois os intervalos já estão com o valor correcto (ou não seria esta a escala modelo). Esta tonalidade, que é dó maior, seria representada pelo valor 0. Mas se pensarmos noutra tonalidade, por exemplo mi maior (figura 2).

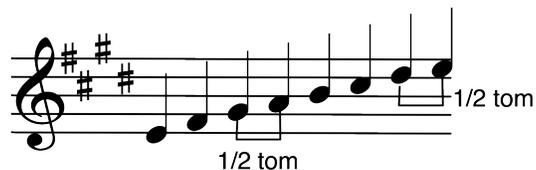


Figura 2: Escala de mi maior

Nesta escala, para mantermos os intervalos de uma escala maior temos de introduzir 4 alterações de expansão (sustenidos). Assim, em MusicXML representaríamos esta tonalidade pelo valor +4.

Em MusicXML usa-se esta forma de representação para as tonalidades mais usuais. Neste trabalho, o que se fez foi caracterizar todas as tonalidades possíveis e associar o valor característico da tonalidade à nota base dessa tonalidade. As tonalidades mais usuais são fáceis de caracterizar mas há algumas um pouco mais complexas. Iremos apresentar alguns exemplos dessas.

Aplicando esta metodologia às tonalidades que têm como nota base as notas conhecidas de toda a gente: dó, ré, mi, fá, sol, lá e si, obtemos a tabela 1.

O único valor que sai do padrão é o da nota Fá: -1. Isto deve-se ao facto de na escala de Fá maior ser necessário introduzir uma alteração de contracção para manter a relação de intervalos de uma escala maior. Na metodologia aplicada temos o valor +1 por cada alteração de expansão e o valor -1 por cada alteração de contracção.

Como cada nota pode sofrer uma alteração de expansão (sustenido) ou de contracção (bemol) este trabalho não ficaria completo se não caracterizássemos as tonalidades

Nota	Valor
Dó	0
Ré	+2
Mi	+4
Fá	-1
Sol	+1
Lá	+3
Si	+5

Tabela 1: Valores das notas

Nota	Valor	Nota	Valor	Nota	Valor
Dó b	-7	Dó	0	Dó #	+7
Ré b	-5	Ré	+2	Ré #	+9
Mi b	-3	Mi	+4	Mi #	+11
Fá b	-8	Fá	-1	Fá #	+6
Sol b	-6	Sol	+1	Sol #	+8
Lá b	-4	Lá	+3	Lá #	+10
Si b	-2	Si	+5	Si #	+12

Tabela 2: Valores das notas

derivadas daquelas alterações. Desta forma chegou-se à tabela 2 que apresenta uma representação uniforme de todas as notas possíveis numa frase musical.

Vamos ver agora como é que se chegou a determinados valores. Por exemplo, o Fá bemol tem o valor -8 . Podemos visualizar a escala de Fá bemol maior na figura 3.

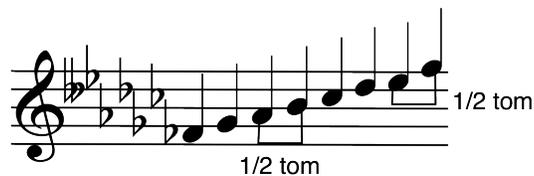


Figura 3: Escala de fá bemol maior

Na escala de fá bemol maior, começamos a escala em fá bemol e para manter a relação de intervalos de uma escala maior somos obrigados a introduzir mais sete bemóis. Precisamos de colocar dois bemóis na nota si, o que se designa por duplo bemol. No fim, a contagem dá um total de 8 bemóis e daí o valor -8 que se associou a esta tonalidade e à sua nota de base.

Vamos ver agora a escala de si sustenido maior que apresenta o valor de $+12$.

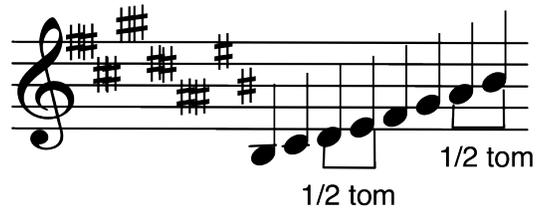


Figura 4: Escala de si sustenido maior

Podemos considerar que esta escala tem 12 alterações de expansão, pois tem sustenidos em todas as notas (7) e cinco deles são duplos ($7 + 5 = 12$). Isto acontece, porque para começar a escala na nota si e manter a relação de intervalos de uma escala maior, esta é a única forma de fazê-lo.

Neste ponto, temos então representações numéricas para todas as notas, inclusive para alterações de expansão ou de contracção. O domínio para representação de notas apresentado na tabela 2 pode ser representado matematicamente por: $[-8, +12]$. Trata-se de um domínio discreto composto por 21 valores inteiros correspondentes às 21 notas.

Vamos ver como é que se pode usar esta representação das notas para obter uma representação de intervalos (note que havendo uma representação numérica os objectos em causa podem materializar-se no universo digital, ou seja, podemos criar abstrações deles em programas).

2.2 Codificação de Intervalos

O objectivo inicial era conseguir calcular um intervalo com uma simples operação aritmética. Vamos então fazê-lo.

Um intervalo entre duas notas pode ser representado pelo valor de uma subacção: subtraindo ao valor da nota destino o valor da nota de origem. Por exemplo, se quisermos calcular o intervalo entra a nota dó e a nota fá (um intervalo que tem o nome de quarta perfeita) vamos calcular a expressão: $-1 - 0$. O resultado é -1 e será este o valor que iremos usar para representar o intervalo de quarta perfeita. Uma questão pode colocar-se neste ponto: Estará o referencial criado para as notas tão bem construído que permita generalizar o cálculo de um intervalo?

Depois de alguns testes e raciocínio constatou-se que sim. Atentemos nos seguintes exemplos para o mesmo intervalo, ou seja, assumindo que o valor -1 representará sempre o intervalo de quarta perfeita (tabela 3).

Do exercício apresentado na tabela 3, podemos concluir que -1 , no referencial que se criou, representa o intervalo de quarta perfeita. Após esta conclusão podemos expandir este raciocínio aos outros intervalos criando uma representação numérica e única para cada um deles. Assim, seguindo este raciocínio e olhando para o domínio matemático das notas ($[-8, +12]$) podemos inferir que o domínio matemático dos intervalos pode ser definido como: $[-20, +20]$.

Nota origem	Nota destino	$N_{orig} - N_{dest}$
Ré	Sol	$1 - 2 = -1$
Mi	Lá	$3 - 4 = -1$
Fá	Si b	$-2 - (-1) = -1$
Sol	Dó	$0 - 1 = -1$
Lá	Ré	$2 - 3 = -1$
Si	Mi	$4 - 5 = -1$

Tabela 3: Intervalos de quarta perfeita

É claro que alguns dos valores deste intervalo correspondem a situações que não fazem sentido no universo musical, ou seja, nem todos os valores deste domínio terão valores no contradomínio dos intervalos.

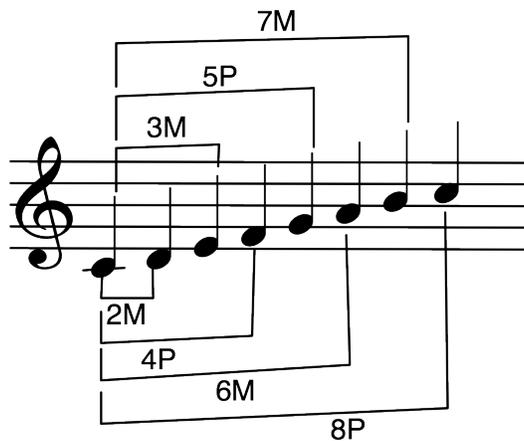


Figura 5: Intervalos a partir de Dó na escala modelo

Vamos olhar agora para a escala modelo de dó maior e usar o mesmo raciocínio para calcular valores para todos intervalos a partir da nota dó (figura 5). Como a nota dó tem o valor 0 os cálculos a fazer estão naturalmente simplificados. Na tabela 4 estão os cálculos e valores calculados para estes intervalos o que nos irá fornecer o esqueleto base para os restantes intervalos que interessam numa perspectiva de análise e representação musical.

Usando esta metodologia podemos calcular os valores das representações dos outros intervalos usados normalmente na construção e análise musical: intervalos menores, diminutos e aumentados.

A tabela 5 apresenta o cálculo das representações para os restantes intervalos.

Nota origem	Nota destino	$N_{orig} - N_{dest} = Valor$	Descrição
Dó	Ré	$2 - 0 = 2$	Segunda Maior (2M)
Dó	Mi	$4 - 0 = 4$	Terceira Maior (3M)
Dó	Fá	$-1 - 0 = -1$	Quarta Perfeita (4P)
Dó	Sol	$1 - 0 = 1$	Quinta Perfeita (5P)
Dó	Lá	$3 - 0 = 3$	Sexta Maior (6M)
Dó	Si	$5 - 0 = 5$	Sétima Maior (7M)
Dó	Dó	$0 - 0 = 0$	Oitava Perfeita (8P)

Tabela 4: Intervalos a partir de dó

Nota origem	Nota destino	$N_{orig} - N_{dest} = Valor$	Descrição
Dó	Ré b	$-5 - 0 = -5$	Segunda menor (2m)
Dó	Mi b	$-3 - 0 = -3$	Terceira menor (3m)
Dó	Fá b	$-8 - 0 = -8$	Quarta diminuta (4d)
Dó	Fá #	$6 - 0 = 6$	Quarta Aumentada (4A)
Dó	Sol b	$-6 - 0 = -6$	Quinta diminuta (5d)
Dó	Sol #	$8 - 0 = 8$	Quinta Aumentada (5A)
Dó	Lá b	$-4 - 0 = -4$	Sexta menor (6m)
Dó	Si b	$-2 - 0 = -2$	Sétima menor (7m)
Dó	Dó #	$7 - 0 = 7$	Meio tom cromático (m.t.cr.)

Tabela 5: Intervalos menores, diminutos, aumentados, etc

Nas tabelas apresentadas (tabela 4 e tabela 5), podemos observar que intervalos inversos têm valores inversos. É o caso da 2m e da 7M (−5 e 5 respectivamente), da 2M e da 7m (2 e −2 respectivamente), da 4A e 5d (6 e −6 respectivamente) e todas as outras inversões. Este facto vem validar o referencial para a representação musical que se propõe neste trabalho.

O MuseCode fica completo com a criação das tabelas de aplicação. Como iremos ver mais à frente, nos cenários de aplicação, para usarmos esta representação de forma expedita precisamos de dois índices invertidos para as notas e de outros dois para os intervalos. No caso das notas, precisamos de um índice de notas que nos mapeie cada nota no seu valor e a correspondente tabela invertida que a partir do valor nos dá a nota. Para os intervalos vamos precisar também das duas tabelas.

Apresentam-se a seguir as quatro tabelas para suportar a consulta de notas e intervalos e respectivos valores.

T1: Notas		T2: Valores		T3: Intervalos		T4: Valores	
Nota	Valor	Valor	Nota	Intervalo	Valor	Valor	Intervalo
Dó b	-7	-8	Fá b	2m	-5	-8	4d
Dó	0	-7	Dó b	2M	+2	-7	m.t.cr.
Dó #	+7	-6	Sol b	3m	-3	-6	5d
Ré b	-5	-5	Ré b	3M	+4	-5	2m
Ré	+2	-4	Lá b	4d	-8	-4	6m
Ré #	+9	-3	Mi b	4P	-1	-3	3m
Mi b	-3	-2	Si b	4A	+6	-2	7m
Mi	+4	-1	Fá	5d	-6	-1	4P
Mi #	+11	0	Dó	5P	+1	0	uníssonos ou 8P
Fá b	-8	+1	Sol	5A	+8	+1	5P
Fá	-1	+2	Ré	6m	-4	+2	2M
Fá #	+6	+3	Lá	6M	+3	+3	6M
Sol b	-6	+4	Mi	7m	-2	+4	3M
Sol	+1	+5	Si	7M	+5	+5	7M
Sol #	+8	+6	Fá #	8P	0	+6	4A
Lá b	-4	+7	Dó #	uníssonos	0	+7	m.t.cr.
Lá	+3	+8	Sol #	m.t.cr.	+7 ou -7	+8	5A
Lá #	+10	+9	Ré #	2A	+9	+9	2A
Si b	-2	+10	Lá #				
Si	+5	+11	Mi #				
Si #	+12	+12	Si #				

Nos cenários de aplicação, que iremos apresentar a seguir, referir-nos-emos a estas tabelas como T1, T2, T3 e T4.

3 Cenários de aplicação

A teoria apresentada na secção anterior e que está resumida e concentrada nas tabelas T1, T2, T3 e T4, pode agora ser aplicada num conjunto muito diversificado de proble-

mas de construção e de análise musical que vão da construção de escalas à improvisação ou à transposição.

3.1 Construção de escalas

Uma escala é normalmente apresentada como um conjunto de intervalos. E quando, a um aluno de música é solicitado que construa uma determinada escala, este começa pela nota base e vai usando a sequência de intervalos da escala em questão para ir descobrindo as restantes notas.

Usando o MuseCode podemos simplificar este processo reduzindo-o a uma simples sequência de somas. Por exemplo, as escalas maiores são caracterizadas pela seguinte sequência de intervalos: $[2M, 2M, 2m, 2M, 2M, 2M, 2m]$ (esta era a escala apresentada na figura 1 e que tem dois meios tons entre a terceira e quarta nota e entre a sétima e a oitava nota). Usando o MuseCode podemos representar uma escala maior pelo seguinte vector (aplicação da tabela T3): $[2, 2, -5, 2, 2, 2, -5]$. Se alguém nos pedisse a escala de lá bemol maior a única coisa que é preciso fazer é pegar no valor da nota base (tabela T1: Lá b = -4) e somar os valores do vector de uma escala maior para ir obtendo o vector com as notas da escala: $[-4, -2, 0, -5, -3, -1, +1, -4]$. Para obtermos a sequência de notas aplicamos a tabela T2 a este vector e obtemos: [Lá b, Si b, Dó, Ré b, Mi b, Fá, Sol, Lá b].

Esta metodologia pode ser aplicada a outros tipos de escalas e a título de exemplo apresentamos na tabela 6 os vectores característicos para outras escalas comuns no estudo clássico de música e uma no estudo de jazz.

Escala	Seq. Intervalos	Vector
Maior	$[2M, 2M, 2m, 2M, 2M, 2M, 2m]$	$[2, 2, -5, 2, 2, 2, -5]$
Menor natural	$[2M, 2m, 2M, 2M, 2m, 2M, 2M]$	$[2, -5, 2, 2, -5, 2, 2]$
Menor harmónica	$[2M, 2m, 2M, 2M, 2m, 2A, 2m]$	$[2, -5, 2, 2, -5, 9, -5]$
Menor melódica ascendente	$[2M, 2m, 2M, 2M, 2M, 2M, 2m]$	$[2, -5, 2, 2, 2, 2, -5]$
Menor melódica descendente	$[2M, 2M, 2m, 2M, 2M, 2m, 2M]$	$[2, 2, -5, 2, 2, -5, 2]$
Pentatónica	$[2M, 2M, 3m, 2M, 3m]$	$[2, 2, -3, 2, -3]$
Hexáfona	$[2M, 2M, 2M, 2M, 2M, 2M]$	$[2, 2, 2, 2, 2, 2]$
Pentatónica de blues	$[3m, 2M, m.t.cr., 2m, 3m, 2M]$	$[-3, 2, +7, -5, -3, 2]$

Tabela 6: Construção de escalas

Na tabela 7 mostram-se exemplos de aplicação na construção de escalas específicas.

Mesmo que pareça complexo este método tem uma grande vantagem, é matematicamente correcto e produz resultados fiáveis. Além disso, a ideia é colocar o computador a fazer estes cálculos.

Designação	Nota+Vector	Escala
Ré M	[2, 4, 6, 1, 3, 5, 7, 2]	[Ré, Mi, Fá #, Sol, Lá, Si, Dó #, Ré]
Si m nat.	[5, 7, 2, 4, 6, 1, 3, 5]	[Si, Dó #, Ré, Mi, Fá #, Sol, Lá, Si]
Si m harm.	[5, 7, 2, 4, 6, 1, 10, 5]	[Si, Dó #, Ré, Mi, Fá #, Sol, Lá #, Si]
Si b blues	[-2, -5, -3, 4, -1, -4, -2]	[Si b, Ré b, Mi b, Mi, Fá, Lá b, Si b]

Tabela 7: Construção de escalas – exemplos

3.2 Construção de acordes

Um acorde musical designa um conjunto de notas que pode ser tocado em simultâneo, no caso de um instrumento harmónico como o piano ou a guitarra, ou em sequência (arpejo), no caso dos instrumentos melódicos como os sopros.

Em MuseCode, cada acorde é também representado por uma sequência ou vector de valores. Aplicando um raciocínio semelhante ao que fizemos para as escalas obtemos as tabelas 8 e 9.

Acorde	Seq. Intervalos	Vector
(M)aior	[3M, 3m]	[4, -3]
(m)enor	[3m, 3M]	[-3, 4]
(d)iminuto	[3m, 3m]	[-3, -3]
(A)umentado	[3M, 3M]	[4, 4]
Maior 1ª inversão (M6)	[3m, 4P]	[-3, -1]
Maior 2ª inversão (M6/4)	[4P, 3M]	[-1, 4]
menor 1ª inversão (m6)	[3M, 4P]	[4, -1]
menor 2ª inversão (m6/4)	[4P, 3m]	[-1, -3]
7ª da dominante	[3M, 3m, 3m]	[4, -3, -3]

Tabela 8: Construção de acordes

Nota base	Designação	Vector	Acorde
Ré	M	[2, 6, 3]	[Ré, Fá #, Lá]
Ré	m	[2, -1, 3]	[Ré, Fá, Lá]
Ré	d	[2, -1, -4]	[Ré, Fá, Lá b]
Ré	A	[2, 6, 10]	[Ré, Fá #, Lá #]

Tabela 9: Construção de acordes – exemplos

3.3 Transposição

Nem todos os instrumentos estão afinados na mesma nota base. Quer-se dizer, por exemplo, que quando se toca um Lá no piano o som é diferente do som ouvido quando a mesma nota é tocada num clarinete, numa trompa ou num saxofone. À adaptação que é preciso fazer à música para que ela possa ser tocada e ouvida da mesma maneira chama-se transposição (por exemplo, quando se quer tocar uma partitura de piano num clarinete).

Este problema agrava-se numa orquestra onde se juntam vários instrumentos, alguns dos quais transpositores. Se o compositor quiser fazer passar determinada frase musical por vários naipes de instrumentos vai ter de transpôr essa frase para o naipe em causa.

O MuseCode também pode ser aplicado para resolver rapidamente estas situações sem margem para erro. Consideremos o seguinte cenário: queremos tocar uma partitura de flauta (que afina em Dó como a maior parte dos instrumentos) num clarinete (que está afinado em Si bemol). O Problema resolve-se transpondo cada nota uma 2M para cima. Usando o MuseCode significaria simplesmente somar o valor de 2M, que é +2, ao valor de cada uma das notas obtendo desta forma as novas notas a serem tocadas.

A título de exemplo vamos transpôr o arpejo de dó maior ([Dó, Mi, Sol, Dó]) no piano para outros instrumentos, tabela 10.

Instrumento	Nota de afinação	Intervalo Relativo a Dó	Vector	Arpejo
Clarinete	Si b	+2M - +2	$[0, 4, 1, 0] + 2 = [2, 6, 3, 2]$	[Ré, Fá #, Lá, Ré]
Saxofone	Mi b	-3m - -3	$[0, 4, 1, 0] - (-3) = [3, 7, 4, 3]$	[Lá, Dó #, Mi, Lá]
Trompa	Fá	-4P - -1	$[0, 4, 1, 0] - (-1) = [1, 5, 2, 1]$	[Sol, Si, Ré, Sol]

Tabela 10: Transposições

4 Conclusão

Neste artigo, apresentou-se o MuseCode que foi criado com o objectivo de ser um referencial matemático uniforme para a representação das notas musicais e de qualquer intervalo entre estas. Os resultados obtidos pela experimentação permitiram validar a abordagem e provar a sua consistência em termos de teoria musical. Podemos ainda referir, que a título experimental, o MuseCode já foi usado por alunos dum conservatório para confirmarem as soluções de problemas de teoria musical e aqui mais uma vez provou ser sólido e de fácil aplicação (é claro que a sua utilização foi proibida porque invalidaria a colocação de qualquer questão sobre construção de escalas e acordes, cálculo de intervalos, transposições, etc, que costumam formar o corpo das questões em teoria musical; qualquer pessoa munida das tabelas T1 a T4 e que saiba aplicá-las resolveria qualquer um destes problemas mesmo sem saber a teoria subjacente). Pode portanto, neste contexto, ser usado como instrumento válido de prova e verificação.

Num contexto informático, as suas aplicações são imensas. Passamos a citar algumas que serão alvo de desenvolvimento nos próximos tempos.

Gerador de partituras para treino É um facto de que já há muitos livros com escalas e arpejos para serem praticados mas muitas vezes não encontramos o que queremos ou então o que queremos está repartido por várias secções ou livros; brevemente começaremos a desenvolver uma aplicação Web suportada pelo MuseCode onde o utilizador poderá seleccionar escalas, arpejos, cadências e gerar partituras à medida com este material; a geração de partituras e de ficheiros áudio MIDI poderá ser feita gerando código para o Lilypond ou ABC;

Criação de um site para treino auditivo Aqui o MuseCode será usado como representação interna de notas e intervalos e irá permitir a geração aleatória de exercícios;

Improvisação Este poderia ter sido um dos cenários de aplicação apresentados; não o foi devido à sua complexidade; quem estuda jazz rapidamente se depara com a necessidade de improvisar; a improvisação segue algumas regras, há escalas que devem e podem ser usadas numa determinada tonalidade, arpejos com e sem alterações, etc; a improvisação faz-se criando frases de 12 compassos combinando pseudo-aleatoriamente alguns daqueles elementos; usando o MuseCode podemos codificar todos os elementos característicos de uma tonalidade (que podem depois ser facilmente transpostos para outra como já vimos) e partindo desta base gerar cadências de improvisação.

Começando a desenvolver estes projectos com certeza mais ideias irão surgir pois a música é multidimensional e o seu estudo pode ser aprofundado em muitas direcções.

Acknowledgments

Queria aqui deixar expresso o meu agradecimento pessoal ao professor José Nuno Oliveira do departamento de informática da universidade do minho, com o qual tive longas e animadas conversas que me apontaram a direcção e que estiveram na génese deste trabalho.