

DESENVOLVIMENTO E IMPLEMENTAÇÃO DE UMA LINGUAGEM DE DESCRIÇÃO DE ALTO-NÍVEL, EXTENSÍVEL E CONECTÁVEL, PARA COMPOSIÇÃO MUSICAL COM SISTEMAS DE AFINAÇÃO¹

Pedro Kröger
kroeger@pedrokroeger.net
Universidade Federal da Bahia

Resumo

Os programas de computador para sistemas de afinação são voltados sobretudo para geração e análise de escalas, não sendo suficientemente práticos para a composição musical. Ferramentas eletrônicas do compositor moderno como Csound e Max, apesar de permitirem o uso de qualquer sistema de afinação, só o fazem de forma direta, manipulando as frequências. Acreditamos ser necessária a criação de uma linguagem de alto nível e escalável onde possa se descrever música no domínio do sistema de temperamento de modo a explorar os ricos recursos sem detrimento da "musicalidade". Esse artigo descreve o estado de uma pesquisa para o desenvolvimento e implementação de uma linguagem de descrição de alto-nível, expansível e conectável, para composição com sistemas de afinação. Foi desenvolvido um sistema preliminar e funcional capaz de permitir a experimentação necessária para criar uma linguagem de composição utilizando a metodologia de desenvolvimento *bottom-up*.

Palavras-chave: Informática em Música, Sistemas de temperamento, Composição

Abstract

Computer programs for tuning system are dedicate for analysis and generation of scales, not beeing practical enough for musical composition. Electronic tools of the modern composer such as Csound and Max, although powerful and capable of using any tuning system, only can do it directly, manipulating frequency numbers. It is necessary the development of a high-level, scalable language that allows the description of music in the domain of tuning systems. This paper describes the state of a research for the development and implementation of a high-level and expansible language for composition using different tuning systems. A working prototype that allows the necessary experimentation to create a compositional language was developed using the bottom-up development methodology.

¹ Esse trabalho foi desenvolvido com o auxílio do CNPq, com uma bolsa de pesquisa na modalidade "Desenvolvimento Científico e Regional".

1 Introdução

O conhecimento de sistemas de afinação é vital para o estudo da música ocidental. Acompanhar a história dos sistemas de afinação é acompanhar a própria história da música, sendo a base para o estudo da harmonia e contraponto.

Durante quase dois mil e quinhentos anos o estudo de sistema de afinação ocupou uma posição dominante seja em tratados teóricos (Rameau 1971), de instrumentos (Barbieri 1991), de canto (Stevin 1955), de composição (Hindemith 1942), ou mesmo matemáticos (Crocker 1964)

Contudo atualmente o estudo dos sistemas de afinação é sub-considerado, relegado à uma introdução básica e visto como algo “esotérico” e pouco prático. Todavia, acreditamos que não se pode falar de harmonia e contraponto sem falar de sistemas de afinação, já que as definições de consonância e dissonância através do tempo estão intimamente ligadas à maneira como os intervalos são afinados. Por exemplo, as terças eram sentidas como consonâncias secundárias (ou mesmo dissonâncias) para os pitagóricos. Com o tempo procurou-se terças mais consonantes, como as terças do sistema justo.

Alguns teóricos defendem a idéia que o acréscimo acumulativo do uso de dissonâncias, que culminam no cromatismo germânico e conseqüentemente no dodecafonismo está vinculado em parte pela adoção do sistema de temperamento igual. Em outros sistemas de afinação como o mesotônico os semitons não tem o mesmo tamanho, portanto cada intervalo tem sua cor característica. No sistema de afinação pitagórica, por exemplo, as terças maiores são “maiores” que o mesmo intervalo no temperamento igual² e as terças menores são muito “menores” no sistema mesotônico³. Isso permite um uso bastante refinado de dissonâncias sem necessitar de estruturas cordais necessariamente complexas.

Acreditamos ser necessária a criação de uma linguagem de alto nível e escalável (o sistema de afinação pode ser modificado a qualquer momento) onde possa se descrever música no domínio do sistema de temperamento de modo a explorar os ricos recursos sem detrimento da “musicalidade”. Por exemplo, poder-se-ia escrever um acorde de do maior e toca-lo em diferentes níveis do sistema de limite-5. Esse sistema deve ser expansível, para que o usuário possa adaptá-lo de acordo com suas necessidades. Finalmente, esse programa

2 A terça maior tem 407.82 cents no sistema pitagórico, 400 no temperamento igual, e 386.31 no temperamento igual
3 As terças menores tem 300, 294, e 281 cents no temperamento igual, sistema pitagórico, e temperamento justo, respectivamente.

deverá ter ligações com programas já existentes, para uma mais completa utilização de recursos composicionais.

Esse artigo descreve o estado da pesquisa cujo objetivo é o desenvolvimento e implementação de uma linguagem de descrição de alto-nível, expansível e conectável, para composição com sistemas de afinação.

2 Metodologia

2.1 Revisão de literatura

Primeiramente foi conduzida uma revisão de literatura relativa ao assunto. Materiais bibliográficos foram adquiridos como o livro *The Just Intonation Primer* de David Doty e mais de 40 números do *1/1: The Journal of the Just Intonation Network*. Além disso, foram baixados os históricos das listas de discussão Harmonic Entropy, MakeMicroMusic, The Tuning List, e Tuning Math. Algumas dessas listas, como The Tuning List tem mais de oito anos de existência e mais de 50000 mensagens arquivadas. Nesses grupos membros da comunidade trocam idéias sobre as pesquisas e idéias mais recentes sobre sistemas de afinação.

Essas listas de discussão mencionadas são hospedadas no Yahoo! Groups que permite um acesso limitado ao histórico das mensagens. Para fazer uma pesquisa completa e avançada esses históricos foram baixados para um computador local e programas específicos para lidar com um grande número de mensagens, como Lurker⁴ e Gnus⁵ foram utilizados. Essa deveria ser uma tarefa trivial, e de fato o é se a lista de discussão utiliza programas de código aberto como o MailMan. Mas como o Yahoo! Groups é baseado em acesso através da visualização de anúncios, o acesso aos históricos é limitado. Para baixar o histórico utilizamos o programa yahoo2mbox⁶ que simula o acesso ao histórico do Yahoo! Groups como se feito por um navegador. Como o Yahoo! Groups tem um limite de *download* nós desenvolvemos um programa na linguagem de programação Python⁷ para reconectar depois de certo tempo. Esse programa faz a autenticação no grupo em questão, começa a baixar o histórico através do yahoo2mbox e repete o processo depois de algum tempo caso a conexão tenha sido interrompida pelo servidor. Esse processo levou cerca de quatro dias não consecutivos.

4 <http://lurker.sourceforge.net>.

5 <http://www.gnus.org>.

6 <http://www.tt-solutions.com/en/products/yahoo2mbox>.

7 <http://www.python.org>.

Foi organizado e catalogado o material bibliográfico disponível em formato digital, primariamente através da internet.

No total foram obtidos 240 artigos e mais de 75000 mensagens de listas de discussão.

2.2 Implementação de rotinas para dados musicais

Foi concluída a implementação das rotinas para dados musicais. Foram implementadas rotinas como:

`name2number` converte a representação com nome de notas para numérica.

`number2note` converte a representação numérica para com nome de notas.

`setsystem` determina o sistema numérico a ser usado.

`module` retorna o módulo de um número de acordo com o sistema numérico (12 ou 96).

`transpose` transpõe uma lista de notas para determinada altura.

`retrograde` retrograda uma lista de notas.

`inversion` inverte uma lista de notas em relação a um eixo.

`complementation` retorna o complemento de um conjunto.

`exclude-repetitions` exclui as notas repetidas em uma lista de notas.

`ordenate` ordena um conjunto de notas.

`intervals` retorna os intervalos que constituem um conjunto de notas.

`normal-form` retorna a forma normal de um conjunto.

`prime-form` retorna a forma prima de um conjunto.

`rotate` retorna a rotação de uma lista de notas de acordo com um índice dado.

`matrix` retorna a matriz serial de um conjunto de 12 notas.

2.3 Definição e implementação da linguagem

O projeto inicial contemplava uma fase para a definição da linguagem (Kröger 2003) e outra separada para sua implementação. Esse é um procedimento tipicamente *top-down*. No decorrer do trabalho decidimos mudar essa abordagem e desenvolver a linguagem através de uma perspectiva *bottom-up* (a seção 4 contém uma descrição mais elaborada).

Dessa maneira foi criado código na linguagem de programação Scheme⁸ para servir como um ambiente para a experimentação com diversas possibilidades com sistemas de afinação. As possibilidades mais desejadas e/ou mais úteis tornar-se-iam a própria linguagem final.

⁸ <http://www.schemers.org>.

Foram implementados procedimentos baseados nas operações listadas em (Doty 2002):

add-intervals soma dois ou mais intervalos representados por frações.

invert-fraction inverte uma fração. Se a entrada é $2/3$ a saída será $3/2$.

subtract-intervals subtrai dois ou mais intervalos representados por frações.

complement retorna o complemento de uma fração ou de uma lista de frações.

ratio->freq converte o valor de fração para hertz em função a uma frequência base.

ratio->cents converte o valor de fração para cents em função a uma frequência base.

ratio->harmonic converte o valor de fração para o número do harmônico em função a uma frequência base.

ratio->sub-harmonic converte o valor de fração para o número do sub-harmônico em função a uma frequência base.

make-chord constrói um acorde de três notas.

export-csound exporta uma lista de frações para o formato de partitura do Csound.

play-fraction toca uma fração dada.

play-chord toca um acorde dado.

Outras operações sugeridas em textos como (Monzo 1998, Secor 2003, Terpstra 1994, Canright 1985) começaram a ser acrescentadas ao sistema. Começou a ser investigado o problema do mapeamento de diferentes sistemas de temperamento para escalas de diferentes tamanhos. Essa fase não chegou a ser concluída devido ao interrompimento do projeto.

3 Resultados alcançados

Os seguintes resultados foram alcançados:

- histórico facilmente acessível das listas de discussão *Harmonic Entropy*, *MakeMicroMusic*, *The Tuning List*, e *Tuning Math*. Um total de mais de 370 megas e quase 76000 mensagens
- lista bibliográfica pertinente ao assunto
- coleção bibliográfica de 96 artigos em formato digital e 144 publicados em papel
- sistema preliminar e funcional capaz de permitir a experimentação necessária para criar uma linguagem de composição

4 Discussão

Como foi mencionado na seção 2.3 o projeto inicial contemplava a definição e a implementação da linguagem em duas fases distintas. Contudo, durante a execução do projeto achamos mais adequado criar desde o início um sistema que permitisse experimentar com diversas possibilidades de definição para a linguagem.

Um dos princípios básicos em desenvolvimento de software é que um programa grande deve ser dividido em partes menores. A abordagem tradicional é chamada de design *top-down*. Cada maior funcionalidade do programa é dividida em sub-rotinas, que por sua vez são sub-divididas em sub-rotinas até “alcançar o nível certo de granularidade” (Graham 1933). Uma outra abordagem é conhecida como design *bottom-up*, onde a linguagem de programação é modificada para se adequar ao problema. As vantagens dessa abordagem é que ela faz com que a linguagem de programação faça a maior parte do trabalho, promove reutilização de código, torna os programas mais fáceis de ler, e ajuda a clarificar as idéias sobre o design do programa (Graham 1933).

Dessa maneira o sistema foi feito em *camadas*, onde a primeira camada implementa operações básicas para lidar com notas, a segunda lida com conceitos básicos de sistemas de afinação, a terceira com conceitos específicos de sistemas de afinação, e assim por diante. Assim é criada uma *barreira de abstração* entre cada camada de modo que cada camada pode ser modificada, desenvolvida e depurada sem detrimento das outras.

Achamos que esse é uma metodologia poderosa para desenvolver software, sobretudo para pesquisa científica porque permite ter *working code*⁹ desde cedo além de ajudar em uma melhor compreensão prática do problema em mãos.

5 Conclusão

Foi desenvolvido um sistema inicial para permitir a pesquisa e a composição com sistemas de afinação. Como esse sistema foi feito de forma incremental (i.e. *bottom-up*) novos recursos podem facilmente ser acrescentados. Esse sistema inicialmente continha operações básicas de manipulação de notas envolvendo afinações (tipicamente frações). Em seguida ele foi ampliado para conter novas operações e formas de representação. Pretendemos continuar o desenvolvimento desse sistema e em breve disponibilizá-lo na internet.

⁹ Código-fonte em um programa funcionando, em oposição à uma lista de especificações.

6 Referências

- BARBIERI, P. Violin intonation: a historical survey. *Early Music*, v. 19, n. 1, p. 69–88, 1991.
- CANRIGHT, D. Rational notation. *1/1: the Journal of the Just Intonation Network*, v. 1, n. 1, 1985.
- ROCKER, R. L. Pythagorean mathematics and music. *Journal of Aesthetics and Art Criticism*, v. 22, n. 3, p. 325–335, 1964.
- DOTY, D. B. *The Just Intonation Primer*. [S.l.]: The Just Intonation Network, 2002.
- GRAHAM, P. *On Lisp: Advanced Techniques for Common Lisp*. [S.l.]: Prentice Hall, 1933.
- HINDEMITH, P. *Craft of Musical Composition*. [S.l.]: European Amer Music Dist Corp, 1942.
- KRÖGER, P. Desenvolvimento e implementação de uma linguagem de descrição de alto-nível, extensível e conectável, para composição musical com sistemas de afinação. Projeto de pesquisa na modalidade Desenvolvimento Científico Regional. 2003.
- MONZO, J. L. *JustMusic: A New Harmony Representing Pitch as Prime Series*. [S.l.]: Publicado pelo autor, 1998.
- RAMEAU, J.-P. *Treatise on Harmony*. [S.l.]: Dover, 1971. Traduzido por Philip Gossett.
- SECOR, G. *The miracle temperament, decimal keyboard, and sagittal notation*. No prelo a ser publicado por Xenharmonikon. 2003.
- STEVIN, S. *Vande spiegheling der singconst*. In: FOKKER, A. (Ed.). *Principal Works of Simon Stevin*. Amsterdam: Swets and Zeitlinger, 1955. v. 5, p. 413–464. Trad. Adriaan Fokker.
- TERPSTRA, S. More notas on noation. *1/1: the Journal of the Just Intonation Network*, v. 8, n. 3, p. 1, 1994.