

SOAL: Ferramenta para análise musical no ambiente Open Music

André Lira Rolim^{1*}, Bruno Jefferson^{1*}, Didier Guigue^{2†}

1-Departamento de Informática – Universidade Federal da Paraíba - Campus 1 -
João Pessoa – Paraíba – Brasil

Andrefloyd10@yahoo.com.br, bruno@lavid.ufpb.br

2- Departamento de Música – Universidade Federal da Paraíba - Campus 1 -
João Pessoa – Paraíba - Brasil

dguigue@cchla.ufpb.br

Resumo. Este artigo apresenta uma biblioteca de funções que implementamos com a linguagem LISP e CLOS, e que se encontra integrada ao ambiente OpenMusic. A SOAL (Sonic Object Analysis Library) é destinada à análise de arquivos MIDI de música do Séc. XX, dentro de certas premissas analíticas que iremos também brevemente explicitar.

Abstract. This paper shortly presents a library of functions, implemented in LISP and CLOS, we have developed as part of the OpenMusic environment. The goal of SOAL - SonicObjectAnalysisLibrary - is to analyze MIDI files of 20th Century music, according to some premisses we will also briefly explain.

1.Introdução

O suporte teórico deste projeto de biblioteca de funções é uma abordagem da música não-tonal do Séc. XX a partir do conceito de *objeto sonoro*. As premissas e a metodologia foram extensivamente descritas e exemplificadas em publicações anteriores [Guigue 1997, Guigue 2005a, Guigue 2005b]. De forma resumida, podemos sintetizar que, no domínio da música acústica, o que chamamos de *objeto sonoro* pode ser definido como a combinação e interação de componentes musicais primários (*classes de notas*) com secundários. Estes são componentes de ordem estatístico e/ou relativo, tal como intensidades, densidades, e, geralmente falando, componentes que quantificam ou qualificam as modalidades de preenchimento do “espaço” e do “tempo” musicais. É uma estrutura de nível intermediário, entre o nível elementar e atômico das classes de notas e o nível superior da macro-estrutura. A forma como esses objetos são interligados pode vir a ser um importante vetor da estruturação da música do século 20, quando o timbre se tornou tão ou mais funcional quanto a organização das alturas [Guigue 2005b]. SOAL é distribuída aos membros do Forum *OpenMusic*¹.

*Bolsista CNPq do programa PIBIC da UFPB, processo 109368/2003-7.

† Projeto de pesquisa financiada pelo CNPq.

¹ Mais informações: <http://forumnet.ircam.fr/> (menu *OpenMusic*).

Também é disponibilizada gratuitamente no site do GMT na UFPB, onde se desenvolve este trabalho [<http://www.cchla.ufpb.br/gmt/> (menu “software”)].

2. Estrutura da biblioteca

SOAL é dividida em duas pastas principais que agrupam as funções essenciais de análise dos arquivos MIDI: a pasta *Achronic Analysis*, que apenas considera as informações de notas (*pitches*) e intensidades (*velocity*) dos arquivos analisados, e a pasta *Diachronic Analysis*, que processa todas as informações relativas as modalidades de distribuição no tempo dos eventos (*notas*).

Ambas as pastas são divididas em duas sub-pastas para organizar as funções de acordo com seu objetivo principal, que é, seja de ordem quantitativo (primeira sub-pasta de cada pasta, medindo o âmbito, espacial ou temporal, do objeto analisado), seja de ordem qualitativa (segunda sub-pasta), qualificando as modalidades de distribuição dos eventos de acordo com diversos paradigmas.

SOAL disponibiliza também outras pastas: uma, chamada *piano specific*, contém uma seleção de funções otimizadas para música de piano; outra, *stats & utils*, oferece uma coleção de funções mais genéricas para propósitos de análise estatística e matemática, incluindo entre outras, vários modelos de cálculo do desvio padrão otimizado para dados musicais. Finalmente, fica a função de leitura dos arquivos MIDI, e cujas informações em saída são no formato que todas as demais funções de *SOAL* esperam receber em entrada.

3. Utilizando a SOAL

A função que extrai dados dos arquivos MIDI para serem utilizados nas demais funções da biblioteca é chamada de *multi-midi-reader*, e desde a versão 1.2 agora distribuída, suporta a leitura recursiva de um número teoricamente infinito de arquivos MIDI como entrada². O *multi-midi-reader* extrai quatro tipos de dados, a saber: *notes* (notas), *onsets* (disparos), *durations* (durações) e *velocities* (velocidades).

De posse dessas informações, o usuário pode então utilizar as várias funções da biblioteca, podendo até agregar funções da linguagem LISP e demais funções definidas no *OpenMusic*, aproveitando as facilidades oferecidas pela interface deste ambiente. A figura 1 mostra um *patch* típico utilizando *SOAL*³. Nela o *multi-midi-reader* extrai informações de dois arquivos MIDI, onde as suas saídas estão ligadas a outras duas funções da *SOAL*, *file-duration* e *smaller-impulse*, que calculam o tempo e o menor impulso de cada arquivo, respectivamente. O exemplo mostra também que algumas funções de *SOAL* chamam outras funções, a exemplo da *events-density* aqui representada, que calcula a densidade relativa de eventos em relação a duração completa dos arquivos MIDI (cf. a documentação on-line na mesma figura). Vemos que *events-density* recebe os resultados das funções *file-duration* e *smaller-impulse* como entrada. Como todas as operações padrão em *OpenMusic*, o resultado das avaliações de *SOAL* é mostrado na janela LISP chamada “Listener”.

² A limitação do número de arquivos depende da limitação de hardware disponível.

³ A biblioteca é distribuída com alguns patches demonstrativos.

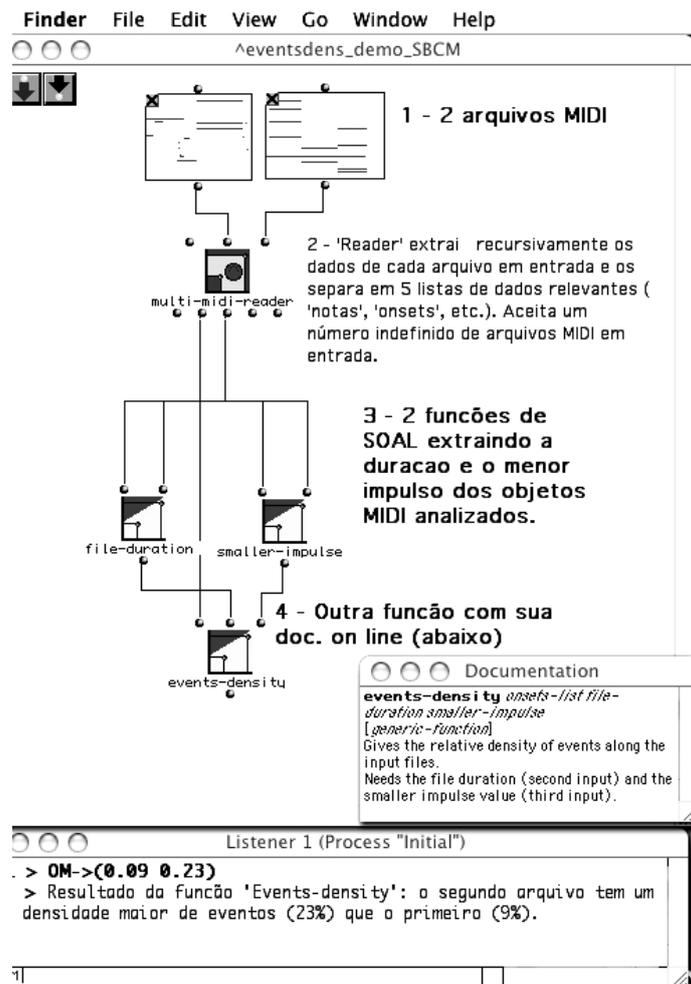


Figura 1: Um exemplo de aplicação da SOAL: o cálculo da densidade de eventos de dois arquivos MIDI utilizando 4 funções da biblioteca.

4.Implementando a SOAL

O uso de *LISP* na implementação da *SOAL* proporcionou grande suporte na implementação da *SOAL*. Visto que a necessidade de constantes cálculos matemáticos e ou estatísticos sob um ou mais conjuntos de dados de mesma natureza, nos obrigam a manipular inúmeras listas de dados, a linguagem *LISP* oferece grandes facilidades nas operações das mesmas. [Touretzky 1990]

O código fonte da biblioteca *SOAL* é organizado em oito arquivos: *Soal.lisp*, *OM-Methods.lisp*, *space.lisp*, *stats.lisp*, *time.lisp*, *piano-specific.lisp*, *constants.lisp* e *aux.lisp*, de modo que, como todos esses arquivos são empacotados no ambiente *Open Music*, qualquer função em um arquivo pode referenciar outra função contida em outro. Além dessas funções declaradas nos arquivos, a *SOAL* pode utilizar funções que *OpenMusic* implementa, fazendo referencia direta a elas, como no exemplo de código fonte mostrado na figura 2, onde temos uma função usada no *multi-midi-reader*, a *OperaMidiFiles*, que utiliza a função do Open Music *mf-info* para obter informações do primeiro elemento do parâmetro *midiFiles*.

```
(defun OperaMidiFiles(midiFiles)
  (first (mf-info (first midiFiles))))
```

Figura 2: declaração da função *OperaMidiFiles*, que retorna a primeira informação obtida pela função do Open Music, *mf-info*, referente ao primeiro arquivo conectado ao leitor *multi-midi-reader*.

5. Conclusões e trabalhos futuros

Neste curto texto, foi apresentada uma ferramenta computacional construída para fins de análise musical no ambiente *Open Music*. A construção da biblioteca *SOAL* se justifica pela inexistência de ferramentas automatizadas de avaliação (principalmente estatística) da estrutura de objetos musicais dentro deste ambiente especializado para a composição erudita. Ele interessa não somente ao analista, ao estudioso ou ao musicólogo, como também ao próprio compositor, que pode assim avaliar imediatamente as qualidades estruturais dos objetos gerados a partir de determinado processo composicional em fase de execução ou de experimentação.

SOAL se encontra em contínuo desenvolvimento. Em suas próximas versões serão trabalhadas a extração e a implementação de funções em cima dos assim chamados “controles contínuos” [MIDI 1996] assim como o desenvolvimento de outras funções de análise acrônica e/ou diacrônica dos eventos.

Referencias

Guigue, D. (1997). “Une Étude ‘pour les Sonorités Opposées’. Pour une analyse ‘orientée objets’ de l’œuvre pour piano de Debussy et de la musique du XX siècle”. Villeneuve d’Ascq : Presses Universitaires du Septentrion, 560 p. Acessível em forma condensada no servidor do GMT [<http://www.cchla.ufpb.br/gmt/> (menu “papers”)].

Guigue, D. (2005a). Sonic Object Analysis Library — OpenMusic Tools for analyzing musical objects structure. Electronic documentation of *SOAL*. Paris: IRCAM, 27 p. Também disponível no servidor do GMT [<http://www.cchla.ufpb.br/gmt/> (menu “software”)].

Guigue, D., Onofre, M. F., Rolim, A. (2005b). *SOAL* for music analysis: a study case with Berio’s *Sequenza IV*, 12^a Journées d’Informatique Musicale – JIM 2005. Paris: Actes des JIM 2005, p. 13-18. [<http://jim2005.mshparisnord.net/articles.htm>].

MIDI Manufacturers Association (1996). Complete MIDI 1.0 Detailed Specification, Version 96.1.

Touretzky, D. S. (1990). COMMON LISP: A Gentle Introduction to Symbolic Computation. Redwood City: The Benjamin/Cummings Pub. Co.