

# **ESTRATÉGIAS COMPOSICIONAIS NA INTERAÇÃO HOMEM-MÁQUINA**

Cristiano Severo Figueiró, mestre  
[figocris@yahoo.com.br](mailto:figocris@yahoo.com.br)  
Anselmo Guerra de Almeida  
[anselmo@musica.ufg.br](mailto:anselmo@musica.ufg.br)  
Universidade Federal de Goiás

## **Resumo**

Esse artigo é dividido em três partes: na primeira são expostos elementos gerais de reflexão acerca do fazer musical e da composição na contemporaneidade, baseando-se nas idéias sobre a mudança de paradigma da música contemporânea de Edson Zampronha e nas reflexões em relação ao desenvolvimento tecnológico enquanto extensão do pensamento humano de Marshal McLuhan. Na segunda parte são expostos exemplos ilustrativos sobre o desenvolvimento histórico da música interativa tanto no nível da composição quanto no nível da performance e idéias de composição interativa de Todd Winkler. A terceira parte do artigo é destinada à apresentação de um exercício interativo desenvolvido no ambiente Max/MSP, no intuito de exemplificar uma ferramenta interativa.

**Palavras-chaves:** música computacional, composição interativa, tecnologia musical.

## ***Abstract***

*This article is divided in three parts: in the first one are exposed general elements about making music and contemporary composition, based on the Edson Zampronha's ideas on the contemporary music paradigm changes and on the reflections in relation to the technological development while extension of the human thought of Marshal McLuhan. The second part include illustrative examples on the historical development of interactive music, by the compositional and performance levels. There are displayed Todd Winkler's ideas of interactive composition. The third part of the article is destined to the presentation of a developed interactive exercise in the Max/MSP environment, in the intention of exemplify an interactive tool.*

## **I. Reflexões acerca da percepção, performance e composição que circunda a música de caráter interativo**

A performance musical tradicional traz um nível de interação entre *performer* e obra que se chama interpretação. Apesar do fato de nenhuma performance ser igual, podemos identificar, nas interpretações, muitos maneirismos e idiossincrasias próprias dos estilos, constituindo estereótipos bem fundados. Zampronha (2000) descreve dois métodos de como trabalhar a escrita de obras que surgem sob a ótica da percepção ao invés do reconhecimento de estereótipos, que são: a escrita como signo incompleto e a escrita como signo saturado. Estas técnicas acabam exigindo uma maior presença criativa do *performer* e um maior nível de interatividade entre *performer* e obra.

A interatividade na relação do ouvinte com a obra é mais complexa que a interatividade obtida do compositor com o meio e os materiais, e da obra com a sua performance. Depende de outros fatores como o meio pelo qual está sendo transmitida a obra, a bagagem cultural e o nível de percepção cognitiva do ouvinte. Podemos falar da interatividade de uma percepção e/ou reconhecimento participativo ou da interatividade da obra com os parâmetros perceptuais do ouvinte.

A composição baseada no trabalho e processo composicional, que dispensa o papel da performance humana, como por exemplo, alguns casos da música eletrônica, acaba provocando uma espécie de reversão do interesse poético que procura emular o gestual da performance humana. De acordo com Burt (1996), através da composição algorítmica o compositor pode estender a seu “gosto” musical, descobrindo sons e estruturas que ainda não conhecia, sintetizando a estratégia do compositor no sentido de se expressar criando um espaço de diálogo com a subjetividade de seu próprio gosto que é um sentimento não-passivo, que se diferencia da sensação que é modo como nos relacionamos com os objetos externos, sendo portanto, receptiva; ao contrário, o gosto não é uma simples faculdade de registro e análise. Ele possui uma verdadeira virtude criadora, portanto ponto fundamental de auto-análise no trabalho composicional.

No sentido de aumentar o nível de interatividade ao nível composicional, através da automação por algoritmos, podemos acrescentar elementos de performance ao sistema. Segundo Roads (1996):

"Nós distinguimos dois níveis de interação: (1) interação relativamente leve com um "ambiente composicional", baseado em experiência de estúdio, onde existe o tempo para editar e retroceder se necessário; (2) intensa interação em tempo-real

baseada em trabalhos com sistemas de performance de palco, onde a ênfase está no controle de um processo musical em andamento e não existe tempo para edição."

O ambiente de programação *Max* se enquadra na segunda categoria por possibilitar interação em tempo real em vários níveis, tanto na manipulação ou intervenção na performance quanto no nível composicional. Cada *software*, por suas características e especificidades, vai conduzir em algum nível, o pensamento composicional.

Indo radicalmente neste pensamento, podemos citar McLuhan (1964):

" Os efeitos da tecnologia não ocorrem aos níveis das opiniões e dos conceitos: eles se manifestam nas relações entre os sentidos e nas estruturas da percepção, num passo firme e sem qualquer resistência. O artista sério é a única pessoa capaz de enfrentar, impune, a tecnologia, justamente porque ele é um perito nas mudanças da percepção."

Esta colocação nos leva a um nível de reflexão em que colocamos em questão os meios e métodos que utilizamos para nos expressar. Meios e métodos que sempre nos chegam pré-configurados e em que a expressão se torna a transcendência do que é colocado.

## **II.Exemplos históricos do desenvolvimento da interação homem-máquina e estratégias interativas através de técnicas de programação**

No sentido de disponibilizar elementos interativos ao músico, podemos trabalhar diretamente na estrutura do programa a ser desenvolvido. Segundo Almeida (1997):

"Tornar um *sistema interativo* significa dar ao compositor diferentes níveis de acesso, seja no nível timbrístico, no nível de construção de eventos sonoros, ou no nível macro, no planejamento da estratégia composicional. Sob este ponto de vista, o sistema seria um tipo de *assistente musical*, cabendo ao compositor a tarefa de formular o melhor possível o problema a ser resolvido, e escolher o *assistente* mais adequado. Um ambiente interativo de composição musical se assemelha a uma caixa de ferramentas. A flexibilidade destas ferramentas permite aos compositores o domínio dos processos compostionais".

O problema está em como fazer o computador responder aos impulsos que o músico lhe manda. O histórico de pesquisa em música computacional nos leva à peça *Hornpipe* de Gordon Mumma de 1967. Segundo Nyman (1974):

"*Hornpipe* é uma composição para Trompa solo modificada com *console* "cibersônico". O *console* é um tipo de computador analógico que é usado pelo *performer* e preso ao seu cinto (o que possibilita se ver muito claramente como os artefatos ele-

trônicos podem ser literalmente uma extensão do *performer* e seu instrumento). A Trompa em si é modificada com várias palhetas em vez do bocal convencional e com chaves rearranjadas que permitem que o som seja ouvido de diferentes partes do instrumento.”

Podemos analisar através deste pequeno trecho de um texto já relativamente antigo, (em relação as possibilidades das novas tecnologias) que os impulsos que levaram a composição desta peça são praticamente os mesmos que continuam a nos impulsionar a compor música interativa, ou seja, criar um espaço de diálogo entre as estruturas acústicas e eletrônicas.

No seu livro "Composing Interactive Music - Techniques and Ideas Using Max", Todd Winkler nos apresenta uma visão ampla sobre as possibilidades do ambiente MAX. Segundo Winkler (1996):

"A multiplicidade de opções disponíveis para a música interativa deixa a arte de compor mais difícil. As possibilidades técnicas sozinhas, podem ser irresistíveis: sampleamento, síntese e processamento de sinal, estruturas lineares e não-lineares, algoritmos improvisacionais, performance improvisacional, partituras totalmente notadas, resposta computacional seqüenciada, e assim por diante. Se trabalhando sob uma perspectiva de "*top-down* ou *bottom-up*"<sup>1</sup>, o que é necessário é um forte conceito estético ou guia que irá ajudar a definir o propósito, a maneira, os limites, e o campo de ação do trabalho."

Muitos trabalhos interativos caminham no sentido da extensão física e virtual das possibilidades dos instrumentos tradicionais. É o caso da peça *Mirror-Rite* de Jonathan Impett (1996), que é para "meta-trompete", computador e *live-electronics*. A composição não possui notação nem gravação, mas ela se forma como um complexo de estruturas baseadas em regras, transformações e processos sobre a improvisação de um *performer* em tempo-real. Segundo Impett (1996):

"O meta-trompete é um sistema interativo de integração instrumento-interface-composição. [...] Aspectos físicos da performance inerente ao trompete são abstraídos e estendidos para resultarem tanto em parâmetros de controle como em material composicional. O sistema tem um conhecimento de seu passado, estado presente e, por análise, aspectos da natureza de seu comportamento."

---

<sup>1</sup> *top-down* se refere a abordagem estratégica de programação em que se começa a escrever um programa a partir do seu plano e função geral enquanto que *bottom-up* se refere a uma abordagem em que escrevemos vários sub-programas que podemos interligar para desenvolver um maior.

O sistema desta peça se alimenta de dados provindos de sensores no palco, nas chaves do trompete e dos sons emitidos pelo trompete. Estes dados são filtrados e processados em diferentes níveis como a realização dinâmica da forma da composição, parâmetros de síntese e processamento e acionamento de *samples*. O *performer* tem acesso direto ao nível composicional, através do seu comportamento, manuseando dinamicamente a forma e o caráter.

Em relação a peças que proponham a improvisação e ações indeterminadas do computador, Winkler nos fala:

"Este tipo de trabalho tem uma grande amplitude de possibilidades. Na maioria dos exemplos, um *performer* pode tocar qualquer coisa, e o computador irá responder com algo que é inesperado e não repetível, mas que, espera-se, tenha senso para o *performer* e para o público. Com total liberdade vem a possibilidade de se produzir resultados selvagemente randômicos, caóticos e incompreensíveis. A sensibilidade e habilidades do *performer* são mais cruciais para o sucesso deste tipo de trabalho, como também são as capacidades de resposta do *software*."

Independentemente da técnica ou da estratégia composicional utilizada, uma constante é de que a técnica em si orienta ou conduz o processo criativo. Segundo Winkler (1996): "Elas irão inevitavelmente refletir os conceitos musicais e a estrutura ditada pela demanda da composição e a desejada qualidade de interação."

### III.Exercício de interação homem-máquina (*Score-follower*)

Este exercício trata de um *Patch* para MAX/MSP em que se tentou realizar um *score-follower* que acompanhasse um instrumento acústico com o máximo de economia de processamento. O caminho escolhido foi mapear a performance a partir de um microfone, transformar estes dados (freqüências) em dados MIDI e deixar na memória do programa uma seqüência MIDI que sirva de "espelho" para o acompanhamento.

A busca pela interatividade com o *performer* foi a tônica do trabalho. Os dados da performance podem ser mapeados para o acionamento de *samples* que estejam na memória do programa, controle de parâmetros de síntese ou processamento, ou manipulação do nível composicional.

A organização deste *patch* pode ser dividida em quatro partes: escuta, transferência dos dados da performance para dados MIDI, acompanhamento, acionamento de *samples*. A parte da escuta foi resolvida com o uso do objeto **fiddle~**, escrito por Miller Puckette

(1999). Este objeto é um algoritmo que fornece alguns dados sobre o reconhecimento da performance que está na sua entrada de áudio, como *pitch* e amplitude. No caso, foi usado apenas o reconhecimento de *pitch*, que é endereçado para o objeto **midiformat**, que transforma um valor de *pitch* em dado MIDI.

Uma vez que o programa faz o reconhecimento de altura da performance e transforma isso em dado MIDI, podemos enviar este dado para o objeto **follow**, que é um objeto em que podemos sincronizar performances MIDI. Ou seja, o objeto tem uma seqüência MIDI na sua memória e consegue acompanhar uma outra seqüência que fica sendo mandada a ele com base na parte gravada. A cada nota que é reconhecida, o objeto manda uma mensagem que aciona um processo, que no caso é o *playback* de *samples*, e que poderia ser uma série de ações possibilitando diferentes níveis de interatividade, como, por exemplo, o acionamento de estruturas sonoras baseadas no histórico do comportamento da performance, seqüências randômicas com parâmetros variáveis, processamentos variados do sinal de entrada, etc.

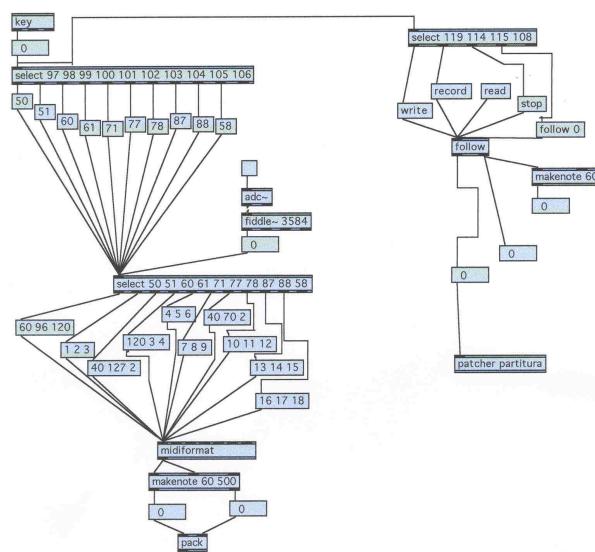


Fig.1: *patch* para MAX que realiza acompanhamento interativo (*Score – Follower*)

## Conclusão

O *score-follower* desenvolvido ainda é uma versão instável, ou seja, ainda não é uma ferramenta com a qual podemos contar massivamente. Mas consiste de uma estrutura básica de um *score-follower*, pronto para desenvolvimento e implementação em composições.

Os principais desenvolvimentos são o trabalho, direto no algoritmo do objeto **fiddle~**, no sentido de estabelecer diferenças entre os timbres dos instrumentos, pois um dos maiores problemas de leitura deste objeto é o fato dele reconhecer muitos componentes parciais de cada timbre, enquanto que o interessante para um *score-follower* seria o reconhecimento apenas do parcial fundamental de cada nota, sendo que para cada timbre ou espectro harmônico diferente ele se comporta de maneira diferente. Outro passo é o de distribuir o processamento, pois, neste caso, o mesmo programa reconhece as alturas e realiza o acompanhamento acionando os *samples*. Se conseguirmos deslocar o reconhecimento para um processador e o acionamento do acompanhamento para outro processador, podemos chegar mais perto de uma situação estável. Poderemos também ampliar a lógica deste *patch*, pois poderá ser incluído, por exemplo, a análise de comportamento da performance de maneira que o *patch* responda de maneiras diferentes de acordo com cada performance.

Para finalizar, podemos concluir que esse processo de pesquisa serve para o aprofundamento da busca por uma identidade artística, através da ampliação de técnicas de programação e da reflexão ao redor do fazer musical contemporâneo e suas relações com as novas tecnologias. Enquanto etapa de formação artística para um compositor, pode-se dizer que esta pesquisa serviu para a ampliação do repertório gestual e idiomático através de possibilidades de *software* e combinação de linguagens.

## Referências Bibliográficas

- ALMEIDA, Anselmo Guerra de. A arte como modelo de interatividade. In: Anais da Anppom, 1999.
- CAESAR, Rodolfo. Novas interfaces e a produção eletroacústica. In: Anais do SBCM, Brasília, 1997.
- MCLUHAN, Marshall. Os meios de comunicação como extensões do homem (Understanding Media). Cultrix, São Paulo, 1993.
- NYMAN, Michael. Experimental music – Cage and beyond. Schirmer Books, New York, 1974.
- OPCODE Systems. MAX Reference Manual. Pablo alto, 1995.
- ROADS, Curtis. Computer music tutorial. MIT Press, Massachussets, 1996.
- WINKLER, Todd. Composing Interactive Music – Techniques and ideas using Max. MIT Press, Massachussets, 1993.
- ZAMPRONHA, Edson. Notação, Representação e Composição – Um novo paradigma da escritura musical. São Paulo, Annablume/Fapesp, 2000.