

# XVI

CONGRESSO  
DE PRODUÇÃO  
CIENTÍFICA E  
ACADÊMICA



## **Desenvolvimento de plugins de controle para a ferramenta Mosaicode**

Gabriel Lopes Rocha, graduando em Ciência da Computação  
Flavio Luiz Schiavoni, Departamento de Ciência da Computação

### **RESUMO**

Este artigo apresenta uma pesquisa voltada, como indica o título, para o desenvolvimento de plugins de controle para a ferramenta Mosaicode. Durante o percurso entretanto ela tomou outro rumo por se deparar com assuntos de grande relevância para o domínio de Arte Digital. Assim, mantemos a discussão sobre o entendimento e desenvolvimento de aplicações de controle para as Artes Digitais mas voltamos nosso foco para o estudo de IMDs (Instrumentos Musicais Digitais) com o intuito de compreender a importância da expressividade concedida por interfaces de controle alternativas e a importância de estratégias de mapeamento para esse fim. Utilizamos para isto o protocolo MIDI através da biblioteca AlsaMIDI do Linux e a biblioteca joystick.h para a elaboração dos protótipos que aqui serão apresentados para exemplificar essa discussão.

### **INTRODUÇÃO**

O Mosaicode [1] é um ambiente de programação visual voltado para o domínio de Arte Digital. Nele, aplicações são criadas através da montagem de diagramas constituídos por blocos. Cada bloco representa uma funcionalidade. A ferramenta tem como objetivo reunir áreas de atuação dentro deste contexto, entre elas o processamento de áudios e imagens, sensores, controladores e redes de computadores. Com isso, tanto artistas experientes sem habilidade em programação quanto estudantes iniciando no assunto podem explorar os seus recursos. A criação de um conjunto de blocos para a utilização de

# XVI

CONGRESSO  
DE PRODUÇÃO  
CIENTÍFICA E  
ACADÊMICA



controladores em aplicações é motivada pela necessidade de se explorar interfaces que permitam ao usuário maior explorabilidade e expressividade dentro de um contexto artístico. Os dispositivos de entrada tradicionais, teclado e mouse, embora sejam eficazes na maioria das tarefas executadas em computadores pessoais, se mostram ineficazes nesse quesito. Eles são em geral ineficientes em tarefas que exigem controle em tempo real e oferecem poucas possibilidades quanto à captura de gestos[2].

Nesse ponto, para analisar a importância da escolha de interface e método de mapeamento na concepção e execução de performances artísticas de cena híbrida, nos voltamos para o estudo de IMDs. Essa escolha começou no desenvolvimento de um protótipo com o objetivo de desenvolver familiaridade com o protocolo MIDI[6]. Este protocolo visa estabelecer um padrão de comunicação entre instrumentos digitais. O objetivo foi fazer com que um controlador MIDI que possui apenas interface para controles gerasse notas musicais e pudesse ser utilizado como uma interface musical. Isso é feito transformando sinais de controle em sinais de nota. Este processo será explicado na seção PROTOTIPAÇÃO a seguir. Esta tarefa foi simples para os botões que apenas enviavam as informações de “pressionado” ou “solto”, similares às teclas de um teclado. Para os botões de deslizar a ideia de produzir som não é algo tão intuitivo. Assim, apareceu o interesse em se explorar a captura de gestos complexos e os processos de mapeamento. Continuando a pesquisa nesse sentido desenvolvemos conceitos importantes para o contexto de Arte Digital, principalmente em situações que exigem interatividade e processamento em tempo real.

No planejamento de interfaces para controle de apresentações artísticas, quesitos como simplicidade e eficiência se tornam menos importantes em prol da expressividade, explorabilidade e criatividade. Um músico em um show ao vivo, por exemplo, impressiona não só em sua habilidade em tocar seus instrumentos como também na maneira que os toca e a impressão que isso causa em seus espectadores. Desta forma, a interface de



entrada utilizada (o instrumento musical nesse caso) é um fator determinante para a performance. Isso é bastante destacado pelos autores que discorrem sobre a elaboração de IMDs, que dão grande importância tanto para a escolha de interface quanto para a escolha de mapeamento da mesma. Através da análise do processo de criação destes instrumentos portanto pretendemos destacar questões importantes como a captura de gestos e maneiras criativas de transformar estes em sons ou comandos.

### **MUSICAL INSTRUMENT DIGITAL INTERFACE (MIDI)**

Na década de 80 grandes avanços tecnológicos permitiram uma revolução na área de música eletrônica. Nesta época, dispositivos de controle, instrumentos e sintetizadores digitais passaram a se comunicar de forma padronizada, independente de seus fabricantes. Isso se deu devido à um acordo entre eles que resultou na criação de um protocolo único para este contexto. Este protocolo foi denominado Musical Instrument Digital Interface (MIDI)[6]. Existem hoje vários dispositivos que o utilizam, entre eles instrumentos, controladores, sequenciadores e sintetizadores. Esse padrão permite a criação de vários sistemas de intercomunicação entre dispositivos que melhor atendam às necessidades do artista. Instrumentos podem ser utilizados para controlar outros em uma relação de mestre e escravo, com resultados interessantes como uma guitarra causando sons de bateria. Controladores podem ser usados para disparar algumas samples pré gravadas ou para modificar características do sintetizador e assim alterar o som gerado pelo instrumento. As possibilidades são várias e atingidas de maneira relativamente simples.

Dispositivos compatíveis com MIDI possuem as portas MIDI IN, MIDI OUT e possivelmente a MIDI THRU. A comunicação se dá através dessas portas, com a MIDI IN funcionando como porta de entrada e a MIDI OUT como saída. A MIDI TRHU funciona



replicando a mensagem recebida na entrada. Assim sistemas podem ser criados ligando dispositivos em cadeia. Para manter o controle nessas situações, MIDI utiliza um conjunto de 16 canais de comunicação. Cada dispositivo dentro de um sistema pode ser programado para responder apenas a mensagens enviadas a canais específicos. Isto funciona apenas se o dispositivo escravo estiver no modo “OMNI OFF”. Se estiver no modo “OMNI ON”, ele responderá às mensagens enviadas a qualquer um dos canais. Outros dois modos possíveis são “MONO” e “POLY” que indicam se mais de uma nota pode ser tocada ao mesmo tempo ou não.

As mensagens MIDI caem dentro de duas categorias: mensagens de voz e mensagens de modo. As principais mensagens de voz são aquelas que carregam informação sobre uma nota, como se ela deve ser tocada ou interrompida, quando foi tocada e sua altura e intensidade, e as de sinais de controle. O resultado de um sinal de controle depende de como ele foi mapeado dentro de um sistema. Existem também as mensagens de Pitch Bend, para dispositivos que possuam essa possibilidade, e mensagens de mudança de controle, em geral atreladas a características específicas de certos fabricantes. As mensagens de modo servem para alterar os modos descritos no parágrafo anterior.

É importante destacar que as mensagens de nota do MIDI não carregam informações sobre o timbre do som que será tocado por um sintetizador. É dito portanto que essas carregam apenas “informações de performance” entre um dispositivo e outro. Isso tanto traz possibilidades, pois o mesmo conjunto de mensagens resultará em melodias diferentes dependendo da escolha de síntese de som, como dificuldades para a criação de partitura digitais. Dado uma sequência de mensagens MIDI, não existem maneiras de se determinar como uma melodia deve ser tocada sem saber a qual sintetizador estas eram originalmente destinadas.



## INSTRUMENTOS MUSICAIS DIGITAIS (IMDs)

O desenvolvimento de instrumentos musicais digitais foi fortemente influenciado pelos avanços tecnológicos que deram aos computadores pessoais a capacidade processamento de som em tempo real. Isso que permitiu que a produção de música eletrônica antes orientada para a reprodução de faixas pré processadas se voltasse à criação de métodos e dispositivos para serem utilizados em performances ao vivo. Este também foi impulsionado pela disseminação de softwares para a criação de aplicações dentro desse domínio, entre eles o Pure Data, Max/MSP e o Super Collider. Atualmente, a criação de IMDs se concentra em projetos personalizados para atender a necessidades específicas do desenvolvedor, ao invés de se voltar para a produção de instrumentos para serem popularizados e utilizados em massa [5].

Em um instrumento musical acústico a parte de controle é inseparável da produção de som. O timbre gerado está fortemente atrelado à estrutura física deste e assim relações complexas de controle são estabelecidas entre o musicista e o objeto. Essas relações evoluíram com o tempo e geraram formas de interação altamente expressivas[3]. Como você toca um instrumento é tão importante na impressão causada ao espectador da performance quanto a melodia resultante. Para instrumentos musicais digitais não existe uma relação direta entre a captura de gestual (dispositivo de entrada) e a produção de som (sintetizador). Isto sugere uma estrutura intermediária que dite como estas duas partes estão relacionadas.

Portanto, os IMDs podem ser compreendidos à partir de uma estrutura tripartite dividida entre dispositivo de entrada, produção de som e mapeamento. Este último é onde a produção de IMDs se torna mais versátil e desafiadora. Mesmo que apenas essa parte seja alterada, o resultado encontrado é completamente diferente do anterior. Não só isso, a estratégia escolhida para o mapeamento é tão importante que chega a definir a



essência de um IMD. É ela que ditará se a experiência de tocar o instrumento é estimulante e expressiva ou simplória e monótona[4].

Frequentemente quando nos deparamos com a tarefa de definir um mapeamento para um IMD, caímos na armadilha de definir uma relação um-para-um entre os sinais de controle e os parâmetros de síntese. Esta é a maneira mais intuitiva de resolver o problema. Entretanto, tal decisão tende a gerar mapeamentos muito simplórios e conseqüentemente fazer com que a experiência de tocar um IMD seja pouco interessante[4]. Outras estratégias possíveis são as de um-para-muitos e muitos-para-um, onde um único sinal de controle afeta vários parâmetros de síntese e vice-versa. A estratégia que tende a ter melhores resultados quanto à expressividade é a de muitos-para-muitos, uma interrelação complexa entre controle e síntese que mais se assemelha a aquelas estabelecidas em instrumentos acústicos[3].

Uma interpretação possível então é a de que o mapeamento é uma parte integral do IMD[3]. Estendendo essa noção para aplicações gerais no contexto de Arte Digital, a escolha de um controlador apropriado não basta para um bom resultado na interação entre o usuário e o produto estético artístico gerado. Não só isso quanto na elaboração de uma performance de cena híbrida o desenvolvimento da interface é tão importante a afetar a elaboração da própria cena. Como pesquisadores da área de exatas tendemos a tomar decisões frias e lógicas. Isso fica evidente na tendência a gerar mapeamentos simplistas para interfaces de controle [3]. Mas isso pode ser extremamente limitador num contexto de criação artística onde decisões contra intuitivas podem gerar melhores resultados.

## PROTOTIPAÇÃO

# XVI

CONGRESSO  
DE PRODUÇÃO  
CIENTÍFICA E  
ACADÊMICA



Durante os estudos sobre interfaces de controle que culminou na análise do processo de criação de IMDs, um total de três protótipos foram criados. O primeiro trata-se da transformação de um controlador MIDI em um instrumento com a utilização da biblioteca *AlsaMIDI*. Essa biblioteca nativa do sistema operacional Linux reúne um conjunto de funções que permitem o desenvolvimento de um software capaz de fazer a interconexão entre dispositivos MIDI. Mensagens do protocolo podem tanto ser recebidas e tratadas quando criadas e enviadas através deste. Um conjunto pessoal de funções simplificadas foi criado no processo para executar tarefas como enviar um “NOTE ON” ou um “NOTE OFF” de maneira simples. Os outros dois foram criados em cima de uma mesma interface de controle, uma manete USB do modelo Dual Analog, popularmente conhecido por ser utilizado nos videogames Playstation da Sony. Para explorar as possibilidades de se usar um controle de videogame dentro deste contexto, três mapeamentos diferentes foram pensados e desenvolvidos.

Transformar um controlador MIDI em um instrumento exige uma tarefa a princípio simples: receber sinais de controle, interpretá-los e traduzi-los para sinais de nota. Para a síntese de sons, utilizamos os sintetizadores disponíveis no programa LMMS, uma plataforma de produção musical. Para estabelecer a conexão entre esse software e o programa que escrevemos utilizamos o *aconect*, um programa desenvolvido para esse fim. O controlador utilizado foi o nanoKONTROL2 da KORG. O nosso código mapeia os botões presentes na parte direita do controlador e atribui a cada um deles uma nota específica, com os tons aumentando de um em um de cima para baixo e da esquerda para a direita. Isso por si só já despertou o interesse de alunos do curso de Música da UFSJ que enxergaram nele uma maneira inusitada de tocar. Uma mensagem MIDI dispõe de 7 bits para especificar uma nota ou um sinal de controle. Isso nos dá 128 valores possíveis a serem transmitidos. Esses valores são utilizados para determinar seja a nota, a intensidade com que o botão foi pressionado ou o valor de controle. Todos estes



estarão dentro desta mesma escala, de 0 a 127. No caso do controlador, mensagens de controle não incluem intensidade. Botões podem enviar os valores 127 ou 0 caso sejam pressionados ou soltos. Os valores dos sliders variam dentro dessa faixa de acordo com a sua posição.

À medida que os sliders mudam de posição eles vão enviando os valores correspondentes. Para fazer com que eles produzam som, o programa transforma cada um desses valores em valores de nota e envia para o sintetizador. O resultado é similar ao de deslizar a sua mão pelas teclas de um piano. Mas como os sliders podem ser movidos bem mais depressa em uma distância mais curta, o resultado sonoro é bem mais impressionante e causou fascínio naqueles que experimentaram o instrumento. Para ficar mais interessante, ao parar o slider em uma posição diferente de 0 (o mais baixo possível), o som correspondente a ele continua a tocar. Como o controlador inclui vários sliders diferentes, o usuário podia posicioná-los de várias formas para gerar composições de sons distintas.

Para trabalhar com a manete de videogame USB utilizamos a biblioteca joystick.h que possui funções e estruturas para ler os comandos recebidos. Para os botões as informações recebidas são um identificador, um valor que indica se o botão foi pressionado ou solto e o tempo em que ele foi apertado. Para os eixos contínuos, as informações incluem o identificador do eixo, a posição atual e o tempo. A manete oferece quatro eixos contínuos, dois para cada alavanca, e dois eixos discretos correspondentes ao D-Pad. Para facilitar a identificação dos botões utilizaremos os nomes popularizados pelo videogame.

Cada mapeamento aplicado na manete tem um princípio diferente. O primeiro foi planejado para ser simplório. Os botões principais “Triângulo”, “Quadrado”, “Bolinha” e “Xis” foram cada um atribuídos a uma nota. Os botões direcionais no lado esquerdo também receberam o mesmo significado. Com estes oito botões é possível tocar todas as

# XVI

CONGRESSO  
DE PRODUÇÃO  
CIENTÍFICA E  
ACADÊMICA



notas de uma escala. A escala pode variar entre maior ou menor de acordo com o botão “Select”. Entre os botões superiores, “L2” e “R2” descem ou avançam uma oitava. O “R1” e o “L1” descem ou avançam um semitom.

O segundo protótipo trazia a ideia de utilizar um mapeamento mais complexo com o intuito de ampliar a expressividade e a explorabilidade do instrumento. Ele foi concebido a partir de uma máquina de estados onde certos botões determinam o estado atual. Outros botões mudam de mapeamento de acordo com cada um destes estados. Assim, os botões direcionais são usados para determinar os estados, enquanto “Triângulo”, “Quadrado”, “Bolinha” e “Xis” são usados para tocar as notas. Os botões superiores seguem a mesma lógica do mapeamento anterior.

Em ambos os casos as alavancas servem para gerar sinais de controle que podem ser atribuídos a parâmetros específicos do sintetizador utilizado. Tomamos essa decisão para que houvesse mais opções de manipulação dos sons gerados. Os eixos contínuos funcionam nesse caso de maneira similar aos sliders do controlador MIDI ao enviar sinais de controle à medida que se deslocam. Vale ressaltar que este último mapeamento apresentado, embora já tenha gerado um protótipo funcional, ainda está em desenvolvimento.

## CONCLUSÃO

Este artigo apresentou a direção tomada pelo projeto de criar um conjunto de blocos de controle para o Ambiente Mosaiccode durante sua execução. Os assuntos apresentados aqui são de grande relevância dentro do tema de controladores e diversos progressos desta pesquisa foram alcançados. Fomos motivados a ir além do óbvio para entender conceitos como expressividade e explorabilidade, que em produções artísticas e em performances ao vivo são de vital importância. Vale ressaltar que o projeto passou



pela troca de discentes durante sua execução, o que gerou atrasos no andamento da pesquisa. Os estudos aqui apresentados nos motivaram a desenvolver pesquisas futuras sobre o assunto.

### AGRADECIMENTOS

Agradeço ao discente Roney Santos por me dar a oportunidade de continuar a sua pesquisa, a todos os moradores do apartamento 201 da Moradia Universitária por serem meus companheiros leais nessa jornada que é a graduação, em especial a Diego Domingos meu antigo colega de quarto. Agradeço ao Professor Flávio Schiavoni pela paciência e dedicação em minha orientação. Agradeço também à PROAE por me dar o suporte necessário por continuar estudando nesta universidade e também a PROPE / UFSJ pelo apoio financeiro a este projeto.

### REFERÊNCIAS BIBLIOGRÁFICAS

[1] Gonçalves, Luan L. Sound Design com o Mosaiccode: Monografia, Universidade Federal de São João Del Rei, 2017:

[https://acervodigital.ufpr.br/bitstream/handle/1884/25892/dissertacao\\_patricio.pdf?sequence=1](https://acervodigital.ufpr.br/bitstream/handle/1884/25892/dissertacao_patricio.pdf?sequence=1) Acesso: 20/08/2018

[2] Hunt, Andy. Kirk, Ross. Radical user interfaces for real-time control:

<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=794755> Acesso: 20/08/2018

[3] Hunt, Andy. Wanderley, Marcelo M. Kirk, Ross. Towards a Model for Instrumental Mapping in Expert Musical Interaction:

# XVI

CONGRESSO  
DE PRODUÇÃO  
CIENTÍFICA E  
ACADÊMICA



[http://recherche.ircam.fr/anasyn/wanderle/Gestes/Externe/Hunt\\_Towards.pdf](http://recherche.ircam.fr/anasyn/wanderle/Gestes/Externe/Hunt_Towards.pdf)

Acesso:

20/08/2018

[4] Hunt, Andy. Wanderley, Marcelo M. Paradis, Matthew. The importance of parameter mapping in electronic instrument design. Proceedings of the 2002 Conference on New Instruments for Musical Expression (NIME-02), Dublin, Ireland, May 24-26, 2002: <http://designblog.uniandes.edu.co/blogs/dise2626/files/2011/01/Mapping-Hunt-Wanderley.pdf> Acesso: 20/08/2018

[5] Patrício, Eduardo L. Brito. INSTRUMENTOS MUSICAIS DIGITAIS – UMA ABORDAGEM COMPOSICIONAL. Dissertação de Mestrado, Universidade Federal do Paraná, 2010:

[https://acervodigital.ufpr.br/bitstream/handle/1884/25892/dissertacao\\_patricio.pdf?sequence=1](https://acervodigital.ufpr.br/bitstream/handle/1884/25892/dissertacao_patricio.pdf?sequence=1) Acesso: 20/08/2018

[6] Roland, MIDI Guidebook:

<http://www.rolande600.com/download/pdf/midi%20manual%20eng.pdf>

Acesso:

20/08/2018