

PLATAFORMA COMPUTACIONAL PARA PROCESSAMENTO DE SINAIS BIOMÉDICOS

G. M. Turchiello*[#], J. Marino-Neto* e J. L. B. Marques*[#]

(*) Instituto de Engenharia Biomédica (IEB-UFSC), ([#]) Departamento de Engenharia Elétrica e Eletrônica, Universidade Federal de Santa Catarina, Florianópolis-SC, Brasil
e-mail: gilsonturchiello@gmail.com

Resumo: Este trabalho apresenta o desenvolvimento de uma plataforma computacional para processamento digital de sinais biomédicos. Trata-se de uma ferramenta gráfica interativa, dinâmica e de simples utilização que traz blocos funcionais com parâmetros editáveis. A plataforma surge também como aliada nas atividades de ensino, pois pode ser utilizada em disciplinas de processamento digital de sinais, auxiliando na apresentação dos conceitos fundamentais de processamento de sinais.

Palavras-chave: Processamento de sinais, DSP, Sinais biomédicos, Java, Software educacional.

Abstract: *This paper presents the development of a computational platform for digital processing of biomedical signals. This is an interactive graphical tool, dynamic and simple to use that brings functional blocks with editable parameters. The platform also comes as an ally in teaching activities, because it can be used in digital signal processing courses, assisting in the presentation of the fundamental concepts of signal processing.*

Keywords: *Signal processing, DSP, Biomedical signals, Java, Educational software.*

Introdução

O processamento de sinais é a análise ou manipulação de sinais visando extrair informações neles contidas. O Processamento Digital de Sinais ou DSP (*Digital Signal Processing*) é uma das áreas de processamento de sinais usados pela Engenharia Biomédica para tratar sinais de diferentes fontes fisiológicas - os sinais biomédicos ou biosinais [1]. Os sinais biomédicos carregam informações de sistemas biológicos/fisiológicos que são significativas para o diagnóstico médico e para a pesquisa clínica [2][3]. Portanto o processamento destes sinais é essencial para obter as informações que estão, em sua maioria, contaminadas por ruído excessivo originado das mais variadas fontes de interferência.

Contudo, processamento de sinais é uma área complexa que demanda conhecimento de ferramentas matemáticas e computacionais [4]. Evidentemente, dada a grande variedade de algoritmos e técnicas matemáticas existentes, cada uma com suas particularidades, é impossível o domínio de toda a tecnologia DSP [5].

Os *softwares*, como o MATLAB[®] (MathWorks, Inc.), surgem como simplificadores no processo de desenvolvimento de ferramentas de DSP, mas também apresentam barreiras. A necessidade de conhecimento de programação na linguagem suportada pelo *software*, por exemplo. Uma maneira de contornar este problema é fazer uso de programas que contenham *scripts* prontos de ferramentas de DSP específicas para uma determinada área de interesse.

Além disso, a aprendizagem de DSP envolve duas tarefas [5]: aprender os conceitos básicos de DSP e aprender técnicas especializadas para uma determinada área e/ou problema a ser resolvido. Desta forma, um *software* com ferramentas básicas e específicas para processamento de sinais biomédicos surge como aliado no ensino de DSP aplicado à Engenharia Biomédica, pois trata-se de uma área multidisciplinar onde alunos e professores necessitam de uma linguagem de comunicação comum.

Este trabalho apresenta o desenvolvimento de uma plataforma computacional de processamento digital de sinais para uso em ensino e pesquisa em sinais biomédicos. Têm inicialmente como público alvo os alunos de graduação em Engenharia Elétrica e de pós-graduação em Engenharia Biomédica da Universidade Federal de Santa Catarina. Através dela deverá ser possível, por exemplo, projetar filtros digitais, manipular sinais por meio de diferentes técnicas a fim de obter o melhor resultado e desenvolver atividades de apoio no ensino de DSP. Algumas características são desejadas no *software* *i.e.*: agrupar funções em blocos; permitir alteração dos parâmetros característicos das funções; ambiente gráfico para manipulação dos blocos e elaboração de projetos de DSP; interação entre os blocos; processamento automático do projeto de DSP; e importação/exportação de sinais digitais.

Materiais e métodos

A plataforma de processamento digital de sinais biomédicos é um aplicativo *desktop* (funciona em um computador sem depender de um navegador de *internet*) que foi desenvolvido em linguagem Java[™]. Foi criada no ambiente de desenvolvimento integrado Netbeans IDE (versão 7.3) com Java *Standard Edition Development Kit 6* (JDK6, Oracle Corporation).

Os recursos gráficos para exibição dos sinais são obtidos a partir da biblioteca *open-source* JFreeChart,

desenvolvida pela Object Refinery Limited e com licença de *software* livre (LGPL - *Lesser General Public License*). Esta biblioteca Java permite a criação de uma variedade de gráficos interativos ou não e com boa qualidade e interatividade.

A Figura 1 mostra a estrutura geral da plataforma desenvolvida, mostrando o fluxo de dados entre os diferentes elementos.

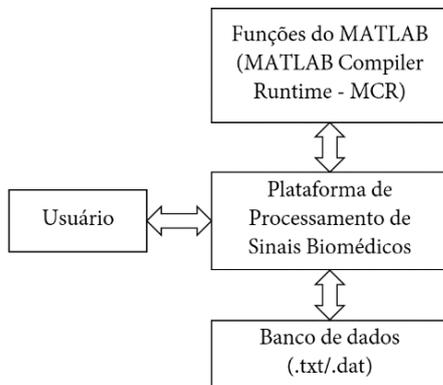


Figura 1: Diagrama simplificado do fluxo de dados da plataforma de processamento digital de sinais biomédicos.

O usuário interage com a interface gráfica da plataforma desenvolvida em ambiente Java. Esta, por sua vez, executa o processamento dos sinais por meio de funções desenvolvidas no *software* MATLAB® e executadas através do *MATLAB Compiler Runtime* (MCR). MCR é uma interface externa que permite que outros programas usem a área de trabalho do MATLAB e executem praticamente todos os pacotes MATLAB.

As funções desenvolvidas na plataforma são representadas como um retângulo, formando um bloco funcional. Estes blocos possuem pinos de entrada ou saída (identificados por triângulos) para conexões com outros blocos. Um triângulo com o vértice voltado para o interior do retângulo do bloco identifica uma entrada enquanto um triângulo com o vértice direcionado para fora do bloco identifica uma saída.

A ligação entre blocos é iniciada através da ação do mouse no pino de interesse do bloco que se quer realizar a ligação. A partir daí segmentos de linhas podem ser inseridos formando uma ligação até o pino de outro bloco. Ligações só podem ser realizadas entre um pino de entrada e outro de saída de dois blocos.

Resumidamente, o desenvolvimento de blocos funcionais iniciou-se pelas operações matemáticas básicas, geradores e importadores de sinais. Posteriormente os blocos para projeto de filtros e o bloco para exibição gráfica dos sinais, finalizando com a ferramenta para análise de sinais de eletrocardiograma (ECG). Para que as características de cada função possam ser alteradas pelo usuário os parâmetros estão acessíveis individualmente em cada bloco. Os blocos são identificados por um nome, um símbolo e uma sequência numérica que é única para cada bloco.

A utilização de blocos na área de desenho da plataforma forma um diagrama em blocos e define um projeto de DSP. Este projeto é reprocessado automaticamente toda vez que alterações são realizadas no mesmo (alteração de parâmetros das funções dos blocos e inserção/exclusão de blocos e ligações). Deste forma, alterações em um bloco são automaticamente percebidas por outros blocos.

Os sinais a serem processados podem ser oriundos de geradores desenvolvidos na própria plataforma ou importados de arquivos. A importação se dá a partir de sinais digitalizados ou criados por outras ferramentas e gravados no computador em arquivos de texto (.txt) ou de dados (.dat). Nos testes realizados foram utilizados essencialmente sinais de ECG gerados/manipulados pelo MATLAB e da base de dados de sinais fisiológicos PhysioNet [6].

A avaliação de usabilidade da plataforma foi realizada, por meio de questionário baseado na versão 7.0 do QUIS (*Questionnaire for User Interaction Satisfaction*), durante as atividades do VII Minicurso de Engenharia Biomédica na Prática (6 a 8 de agosto de 2014). Evento este que é promovido anualmente pelo Instituto de Engenharia Biomédica da Universidade Federal de Santa Catarina (IEB-UFSC) e tem como público alvo alunos de graduação dos cursos de engenharia. Ele possui a finalidade de divulgar a Engenharia Biomédica através de atividades práticas e teóricas ministradas por professores e alunos de pós-graduação do IEB-UFSC. Neste questionário cada item avaliado mede a satisfação do usuário em determinado aspecto do sistema, numa escala de 1 a 9, onde 1 é considerado uma nota ruim e 9 ótima.

Resultados

A Figura 2 mostra a interface principal da plataforma que apresenta uma área de desenho (onde serão inseridos os blocos, formando o projeto de DSP) e outra de seleção de blocos (à esquerda). Há ainda uma barra de menus (superior) e outra de *status* (inferior). Através da barra de menus é possível inserir blocos, efetuar configurações visuais da plataforma e salvar/abrir projetos desenvolvidos na plataforma. A barra de *status* traz informações dos blocos, das operações realizadas na plataforma e orienta o usuário nas ações do mouse nos blocos e ligações. A seguir são descritos os principais blocos disponíveis até o momento na plataforma.

Bloco de Gráficos – Este bloco (Figura 3) pode gerar gráficos de todos os sinais presentes nos demais blocos do projeto e, portanto não possui entradas nem saídas para conexões. A configuração de quais sinais serão mostrados é realizada internamente nas opções deste bloco, onde é possível também definir a espessura da linha, a cor e outras opções individualmente para cada sinal disponível no projeto.

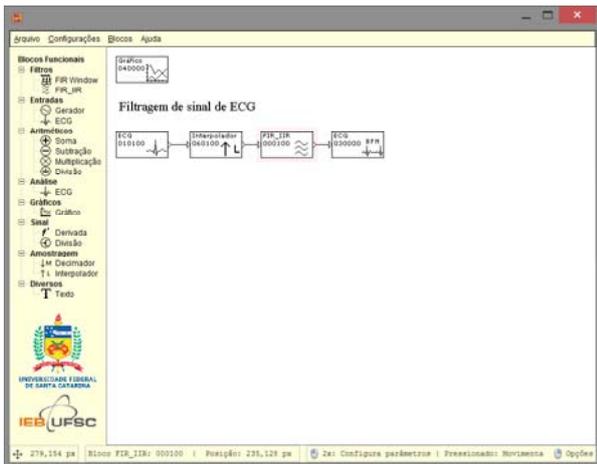


Figura 2: Interface gráfica da plataforma desenvolvida em linguagem Java, exemplificando o uso de blocos funcionais no processo de filtragem de ruídos em sinal de eletrocardiograma.

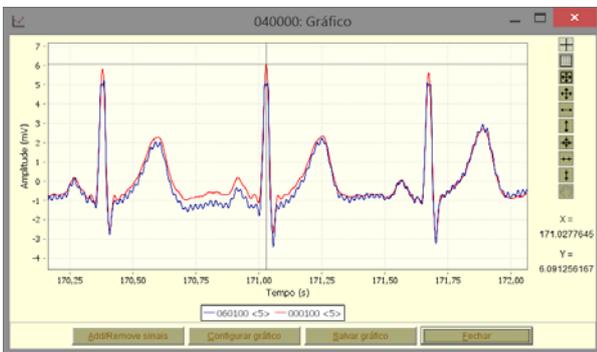


Figura 3: Função do bloco para desenhar os gráficos dos sinais presentes no projeto, exibindo sinais antes (azul) e depois (vermelho) da passagem por um filtro.

Blocos de Filtros – Os blocos de filtros (Figura 4) permitem o projeto de filtros do tipo FIR (Resposta ao Impulso Finita ou *Finite Impulse Response*) e IIR (Resposta ao Impulso Infinita ou *Infinite Impulse Response*). Os filtros FIR podem ser projetados pelo Método da Janela (*Window*) ou *Equiripple*. Já os filtros IIR podem ser *Butterworth*, *Chebyshev* do tipo I e II ou *Elliptic*. Filtros *Notch* também podem ser projetados. Quanto a banda passante, os filtros podem ser do tipo passa-baixas (*lowpass*), passa-altas (*highpass*), passa-banda (*bandpass*) ou rejeita-banda (*bandstop*). Como resultado, este bloco apresenta os coeficientes calculados para as especificações requeridas para o filtro e a resposta de magnitude e fase do mesmo.

Blocos de Análise de Sinais de ECG – Este bloco (Figura 5) tem a finalidade de realizar a análise de um sinal de eletrocardiograma, realizando a marcação dos pontos característicos da onda de ECG. Um gráfico da variação do intervalo entre batimentos (IBI – *Inter-Beat Interval*) ou intervalo RR e dados estatísticos do sinal também são fornecidos.

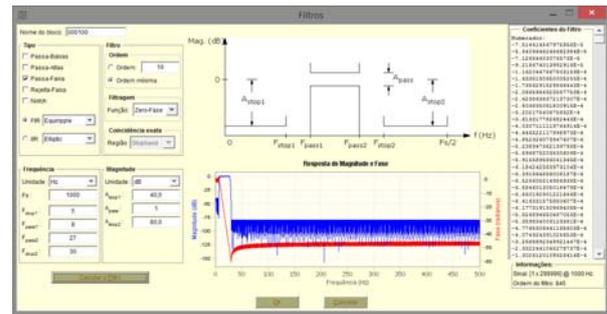


Figura 4: Parâmetros do bloco para projeto de filtros FIR e IIR.

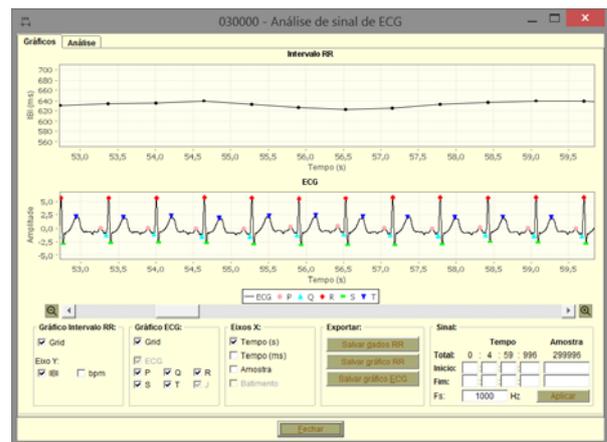


Figura 5: Bloco para análise de sinais de ECG, com marcação de ondas características do ECG e gráfico da variação do intervalo RR.

Blocos aritméticos – Estes blocos permitem operações matemáticas básicas (soma, subtração, multiplicação ou divisão) de um sinal por um valor constante ou até mesmo por outro sinal.

Discussão

A plataforma desenvolvida fornece possibilidades interessantes no ensino de processamento digital de sinais. A plataforma ainda se encontra em desenvolvimento e o número de funções desenvolvidas é limitado, mas já permite seu uso em abordagens de filtragens e análise de sinais.

A plataforma possui uma interface interativa através do mouse, permitindo a inclusão/exclusão de blocos e ligações e a movimentação dos blocos na área de desenho. Os parâmetros das funções podem ser rapidamente alterados individualmente em cada bloco e os demais blocos respondem automaticamente a estas alterações.

Trabalho semelhante é encontrado em [7] onde um aplicativo foi desenvolvido na Universidade do Estado do Arizona para o ensino de DSP em ambiente *Web*. Ele tem sido utilizado principalmente em cursos de graduação em DSP e de comunicação, no ensino presencial e a distância.

No entanto, a plataforma aqui apresentada, diferentemente de [7], é executada no ambiente *desktop* e não necessita de uma conexão com a *internet* para funcionar. Assim, é possível a sua utilização em qualquer momento e local, bem como um melhor tempo de resposta da plataforma visto que não depende do desempenho de uma conexão à *internet*.

Ferramentas como o Simulink® (MathWorks, Inc.) também possuem um ambiente gráfico para criar diagramas de blocos. Simulink é integrado ao MATLAB não funciona independentemente. Logo, para sua utilização, é necessária a instalação do MATLAB e adquirir licença para utilização destas ferramentas.

A validação das funções desenvolvidas na plataforma foi realizada mediante a verificação e comparação dos resultados destas funções com os resultados obtidos com o *software* MATLAB. Os sinais de eletrocardiograma utilizados nos testes foram obtidos no banco de dados PhysioNet que contém uma variedade de sinais biomédicos disponibilizados gratuitamente.

A interação com funções desenvolvidas no MATLAB se mostrou eficiente e expande a capacidade de desenvolvimento de funções de DSP para a plataforma. Contudo, a comunicação entre o aplicativo Java e o MCR não é instantânea, o que ocasiona pequenos atrasos, principalmente na inclusão de blocos.

Conclusão

Este trabalho apresentou os resultados preliminares do desenvolvimento de uma plataforma de processamento digital de sinais que, embora direcionada ao processamento de sinais biomédicos, também possibilita o processamento de outros tipos de sinais. Ela está em constante desenvolvimento e deverá receber novas funções, estendendo sua funcionalidade, tais como análise espectral, transformadas, e ferramentas específicas para processamento e análise de sinais biomédicos.

Onze participantes do minicurso responderam o questionário de avaliação proposto, que resultou em 61% das notas entre 7 e 9 e 34% entre 4 e 6, corroborando para uma boa aceitação da ferramenta. Avaliações semelhantes também poderão ser realizados em disciplinas de processamento de sinais.

A utilização de funções do *software* MATLAB agrega à plataforma maior capacidade de realizar tarefas de processamento de sinais, dada a vasta gama de funções disponíveis no MATLAB. Em contrapartida, é necessária a instalação do MCR para execução da plataforma.

O desenvolvimento do aplicativo em linguagem Java favorece a portabilidade do programa e permite que a plataforma seja executada em qualquer computador com Máquina Virtual Java (JVM – *Java Virtual Machine*) instalada. A plataforma é proprietária e será registrada no INPI (Instituto Nacional de Propriedade Industrial), com distribuição gratuita a usuários.

Agradecimentos

O presente trabalho foi realizado com apoio do CNPq, Conselho Nacional de Desenvolvimento Científico e Tecnológico - Brasil.

Referências

- [1] Muthuswamy J. Biomedical Signal Analysis. In: KUTZ, Myer. Standard Handbook of Biomedical Engineering and Design. New York: McGraw-Hill; 2003. p. 18.1-18.30.
- [2] Bruce EN. Biomedical Signal Processing and Signal Modeling. New York: John Wiley and Sons; 2001.
- [3] Rangayyan RM. Biomedical Signal Analysis: A Case-Study Approach. Wiley-IEEE Press; 2002.
- [4] da Rosa DL. Sistema de Processamento de Sinais Biomédicos: Filtragem de Sinais de Eletroencefalograma [dissertação]. Florianópolis: Universidade Federal de Santa Catarina; 2009.
- [5] Smith SW. The Scientist and Engineer's Guide to Digital Signal Processing. 2ª ed. San Diego: California Technical Publishing; 1999.
- [6] Goldberger AL, *et al.* PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals [internet]. 2000 Jun 13 [cited 2014 Jul 11]; Circulation 101(23):e215-e220. Available from: [http://circ.ahajournals.org/cgi/content/full/101/23/e215]; PMID: 10851218; doi: 10.1161/01.CIR.101.23.e215
- [7] Spanias A, *et al.* Development and Evaluation of a Web-Based Signal and Speech Processing Laboratory for Distance Learning. In: IEEE ICASSP-00; 2000 Jun; Istanbul. 2000. p. 3534-3537.