

UNIVERSIDADE PRESBITERIANA MACKENZIE
ESCOLA DE ENGENHARIA
ENGENHARIA ELÉTRICA

GUSTAVO DE MELO VALEIRA

MODELAGEM E SIMULAÇÃO DE INSTRUMENTOS MUSICAIS
UTILIZANDO FILTROS DIGITAIS

São Paulo
2007

GUSTAVO DE MELO VALEIRA

MODELAGEM E SIMULAÇÃO DE INSTRUMENTOS MUSICAIS UTILIZANDO
FILTROS DIGITAIS

Trabalho de Graduação Interdisciplinar
apresentado ao curso de Engenharia Elétrica, da
Escola de Engenharia da Universidade
Presbiteriana Mackenzie, como requisito parcial
à obtenção do grau de Bacharel em Engenharia.

ORIENTADOR: PROFESSOR DOUTOR MARCIO EISENCRAFT

São Paulo
2007

AGRADECIMENTOS

Ao meu orientador Marcio Eisencraft pela paciência, ajuda, incentivo e comprometimento nas orientações, que me guiaram na realização deste trabalho e no artigo publicado no SBrT 2007.

Aos meus pais, Eurico e Fátima, e meu irmão Rafael, pessoas importantíssimas na minha vida, pela ajuda, compreensão e apoio neste trabalho e no estudo.

Aos meus amigos e professores pelo apoio e incentivo durante e fora das aulas.

“As coisas que se harmonizam em tom, vibram em conjunto.”

Autor desconhecido

RESUMO

Com o avanço da tecnologia, maiores facilidades são encontradas para a geração de sons semelhantes aos criados pelos instrumentos musicais. Uma das formas de se criar notas musicais é utilizando filtros digitais usados, por exemplo, em teclados. Neste trabalho é projetado um filtro digital que captura as características mais importantes do corpo de um instrumento musical. Para tornar o trabalho mais acessível é feita uma revisão bibliográfica sobre processamento de sinais e filtros. A qualidade de uma nota musical gerada pelo filtro é determinada comparando-se o seu espectro em frequência com o da nota produzida pelo instrumento musical.

Palavras-chave: Filtro digital. Processamento digital de sinais. Modelagem de instrumentos musicais.

ABSTRACT

With the advance of technology, it has become easier to generate sounds similar to the created by musical instruments. One of the forms to create musical notes is using digital filters used, for instance, in keyboards. In this work a digital filter is projected to capture the most important characteristics of a musical instrument's body. To make the work self-contained it contains a bibliographical revision on signal processing and filters. The quality of a musical note generated by filters is determined comparing its frequency spectrum with the one of the note produced by musical instruments.

Keywords: Digital filters. Digital signal processing. Musical instruments models.

LISTA DE ILUSTRAÇÕES

Gráfico 1 - Sinal de tempo contínuo correspondente a nota Si gravada através de um microfone.	13
Gráfico 2 - Sinal de tempo discreto $y[n] = e^{-0.1n} \sin(n\pi/5)$	14
Gráfico 3 - Exemplos de sinais de tempo discreto: a) sinal impulso, b) sinal degrau, c) e d) sinais exponenciais.	15
Gráfico 4 - Sinal senoidal $x[n] = \cos(2\pi n/12)$. Neste caso, $\Omega = 2\pi/12$ e $\theta = 0$	15
Gráfico 5 - a) Sinal impulso atrasado em quatro amostras; b) sinal degrau adiantado em seis amostras.	16
Gráfico 6 - Dente de serra com período T_0	17
Gráfico 7 - Espectro em frequência do dente de serra com frequência ω_0	18
Gráfico 8 - Série de Fourier do dente de serra de 200 Hz com 1, 2, 10 e 10000 harmônicas.	19
Gráfico 9 - Sinal amostrado a $f_s = 120 \text{ Hz} < 2f = 220 \text{ Hz}$	20
Diagrama 1 - Diagrama de blocos simplificado de um sistema de tempo discreto.	21
Gráfico 10 - a) Sinal original $x[n] = \sin(\pi n/10)$; b) Dizimação $x_d[n] = x[2n]$; c) Interpolação $x_i[n] = x[n/2]$	24
Gráfico 11 - RDC correspondente ao círculo unitário.	26
Gráfico 12 - Resposta em frequência do sistema da Eq. (22): a) em módulo; b) em fase.	28
Gráfico 13 - Localização dos pólos e zeros no plano z complexo.	30
Diagrama 2 - Filtros em cascata.	31
Diagrama 3 - Sistema $H[z]$ em cascata com o sistema $1/H[z]$	31
Gráfico 14 - Localização no círculo unitário de $p_{1,2} = a \angle \pm \Omega_p$ e $z_{1,2} = b \angle \pm \Omega_z$	33
Gráfico 15 - Módulo da resposta em frequência para: a) $b = 0.4$, b) $b = 0.6$, c) $b = 0.9$, d) $b = 1.0$, e) $b = 1.2$ e f) $b = 2.0$	34
Gráfico 16 - a) Ganho mínimo e b) banda variando o zero b . Foi usado $a = 0.2$ e $\Omega = \pi/6$...	35
Gráfico 17 - Módulo da resposta em frequência para: a) $a = 0.4$, b) $a = 0.6$, c) $a = 0.9$, d) $a = 1.0$	36
Gráfico 18 - a) Ganho máximo e b) banda variando o pólo a . Foi usado $b = 0.2$ e $\Omega = \pi/6$..	36
Gráfico 19 - Módulo da resposta em frequência do filtro inverso: a) instável; b) estável.	37
Gráfico 20 - Módulo da resposta em frequência: a) do corpo do violão; b) do filtro inicial....	41
Gráfico 21 - Módulo da resposta em frequência do corpo do violão e do filtro que está sendo projetado $H(f) = 1$	42
Gráfico 22 - Módulo da resposta em frequência do corpo do violão, do filtro inicial e do filtro projetado.	44
Gráfico 23 - Módulo da resposta em frequência do corpo do violão e do filtro modificado. ..	44
Gráfico 24 - Módulo da resposta em frequência do corpo do violão, do filtro anterior e do filtro anterior em cascata com o projetado no Gráfico 23.	45
Gráfico 25 - Módulo da resposta em frequência: a) do corpo do violão; b) do filtro projetado.	46
Gráfico 26 - Localização dos pólos e zeros do filtro no círculo unitário do plano z.	46
Gráfico 27 - Resposta ao impulso normalizada: a) do filtro criado; b) do corpo do violão. c) Erro.	48

Diagrama 4 - Sistema para gerar a excitação $s_{exc}(t)$	49
Diagrama 5 - Sistema para obter $y_{vsol}(t)$	49
Gráfico 28 - a) Nota Sol gerada pelo violão; b) Entrada necessária para gerar na saída do filtro a nota Sol do violão.	50
Gráfico 29 - Sinais em função do tempo: a) Dente de serra de 196 Hz $f_{serra1}(t)$; b) Resultado $y_1(t)$ da filtragem do dente de serra.	52
Gráfico 30 - Espectro em frequência normalizado: a) do sinal dente de serra filtrado $y_1(t)$; b) da nota Sol do violão. São indicadas as posições das harmônicas nos espectros.	52
Gráfico 31 - Módulo da resposta em frequência do corpo do violão e do novo filtro projetado $H_{f2}(f)$	53
Gráfico 32 - Resposta ao impulso normalizada: a) do filtro $H_{f2}(z)$; b) do corpo do violão. c) Erro.	54
Gráfico 33 - Espectro em frequência normalizado: a) do dente de serra filtrado pelo novo filtro; b) da nota Sol gerada pelo violão. São indicadas as posições das harmônicas nos espectros.	55
Gráfico 34 - Sinal: a) excitação da corda; b) nota Sol artificial; c) nota Sol gerada pelo violão.	56
Gráfico 35 - Espectro em frequência normalizado: a) $H_{y3}(f)$; b) $H_{vsol}(f)$. São indicadas as posições das harmônicas nos espectros.	57
Gráfico 36 - a) Nota Ré gerada pelo processo de interpolação e dizimação da nota Sol do violão e b) seu respectivo espectro em frequência normalizado.	60
Gráfico 37 - Módulo da resposta em frequência das 5 cordas do baixo acionadas separadas (Sol, Ré, Lá, Mi e Si) e acionadas juntas (5 cordas).	62
Gráfico 38 - a) Módulo da resposta em frequência do baixo e da somatória das respostas de cada corda; b) Resposta impulsiva do baixo ao acionar as 5 cordas.	62
Gráfico 39 - Módulo da resposta em frequência: a) das 5 cordas do baixo; b) do filtro projetado.	63
Gráfico 40 - Espectro em frequência da nota Si produzida: a) pelo filtro; b) pelo baixo. São indicadas as posições das harmônicas nos espectros.	64
Gráfico 41 - Espectro em frequência da nota Sol produzida: a) pelo filtro; b) pelo baixo. São indicadas as posições das harmônicas nos espectros.	65
Gráfico 42 - Módulo da resposta em frequência: a) do corpo do baixo ao acionar a corda Si; b) do filtro projetado.	66
Gráfico 43 - Resposta ao impulso: a) $h_{si}(t)$ da corda Si do baixo; b) $h_{Fsi}(t)$ do filtro $H_{Fsi}(f)$	66
Gráfico 44 - a) Nota Si $y_{si}(t)$ gerada pelo baixo; b) Sinal filtrado $y_{Fsi}(t)$	67
Gráfico 45 - Espectro em frequência: a) do dente de serra filtrado $y_{Fsi}(t)$; b) da nota Si do baixo $y_{si}(t)$. São indicadas as posições das harmônicas nos espectros.	68

SUMÁRIO

1	INTRODUÇÃO	9
2	SINAIS E SISTEMAS DE TEMPO DISCRETO.....	12
2.1	SINAIS	12
2.2	SÉRIE DE FOURIER.....	16
2.3	AMOSTRAGEM.....	19
2.4	SISTEMAS DE TEMPO DISCRETO	20
2.4.1	Invariância no tempo	21
2.4.2	Linearidade	21
2.4.3	Resposta ao impulso	22
2.4.4	Dizimação ou compressão no tempo	23
2.4.5	Interpolação	23
2.5	TRANSFORMADA Z.....	24
2.5.1	Propriedades	26
2.5.2	Função de transferência.....	27
2.5.3	Resposta em frequência.....	27
3	FILTROS DIGITAIS	29
3.1	PÓLOS E ZEROS	29
3.2	FILTROS EM CASCATA	30
3.3	CONDIÇÕES PARA ESTABILIDADE.....	31
3.4	FREQÜÊNCIAS DE TEMPO CONTÍNUO E DISCRETO.....	32
3.5	ORDEM.....	32
3.6	VARIAÇÃO DA LOCALIZAÇÃO DOS PÓLOS E ZEROS	33
3.7	SOLUÇÃO PARA FILTROS INSTÁVEIS.....	37
4	SIMULAÇÕES	38
4.1	SINTETIZADORES.....	38
4.2	SIMULAÇÃO DE UM VIOLÃO	39
4.2.1	Projeto do corpo do violão	39
4.2.2	Comparação entre o filtro e o corpo do violão.....	47
4.2.3	Excitação do filtro: sinal produzido pelo violão	48
4.2.4	Excitação do filtro: dente de serra	50
4.2.5	Projeto modificado do corpo do violão	53
4.2.6	Excitação da corda	55
4.2.7	Mudança de tom	58
4.3	SIMULAÇÃO DE UM BAIXO	60
4.3.1	Espectro em frequência.....	61
4.3.2	Projeto do filtro.....	63
4.3.3	Projeto do filtro para a corda Si.....	65
4.3.4	Dente de serra como excitação do filtro.....	67
5	CONCLUSÕES.....	69
	REFERÊNCIAS	72
	APÊNDICE A – FUNÇÕES DO MATLAB.....	73
	APÊNDICE B – ARTIGO PUBLICADO NOS ANAIS DO XXV SIMPÓSIO BRASILEIRO DE TELECOMUNICAÇÕES (SBrT 2007)	84

1 INTRODUÇÃO

Neste trabalho é feito uma pesquisa sobre como a produção de som dos instrumentos musicais pode ser comparada com a dos filtros digitais, baseado em (STONICK; BRADLEY, 1996) e principalmente em (BRADLEY; CHENG; STONICK, 1995; KARJALAINEN; VÄLIMÄKI; JÁNOZY, 1993). A fim de torná-lo acessível a um público maior é feita uma revisão bibliográfica sobre processamento de sinais, Transformada Z e filtros digitais.

Filtro é um sistema que transforma um sinal de entrada em um novo sinal. Filtros seletivos em frequência eliminam componentes que estão em uma determinada faixa de frequência chamada *faixa de rejeição* e deixam inalteradas as que estão em outra faixa, a *faixa de passagem* (HAYKIN; VEEN, 1999).

Um filtro pode ser caracterizado por uma equação de diferenças e por meio desta é possível definir a sua ordem. O maior atraso de amostras que essa equação contiver corresponde a ordem do filtro (LATHI, 1998). Por exemplo, um filtro com entrada $x[n]$ e saída $y[n]$ representado por

$$y[n] = \frac{x[n] + 2x[n-2]}{3} \quad (1)$$

é de 2ª ordem pois, o maior atraso que esta equação apresenta é de duas amostras ($x[n-2]$).

O uso de filtros digitais para a síntese de notas musicais tem uma série de vantagens práticas. Entre elas, destaca-se a facilidade em se mudar as características naturais de um instrumento, permitindo maiores variações do som gerado (STONICK; BRADLEY, 1996). Além disso, podem-se simular diversos instrumentos musicais usando o mesmo equipamento, bastando para isso ajustar os coeficientes dos filtros para cada caso.

É interessante escolher filtros de menor ordem possível, já que assim há

processamento menor, um algoritmo mais simples de ser executado e, conseqüentemente, um produto mais barato.

A modelagem de filtros digitais para simular instrumentos musicais é interessante, pois junta duas áreas, a Engenharia com a Música, que para muitas pessoas é um lazer, divertimento.

Sintetizadores simulam vários instrumentos musicais, como baixo, violão e guitarra. Para isso, usam filtros que criam uma determinada nota dependendo do instrumento selecionado pelo usuário (STONICK; BRADLEY, 1996). Notas são sons produzidos por instrumentos musicais e recebem nomes de acordo com sua frequência fundamental. As sete notas principais são: Dó, Ré, Mi, Fá, Sol, Lá e Si. Reproduzir uma nota através de um filtro digital é interessante, pois com isso é possível simular diversos instrumentos em um só.

O som produzido por uma corda de um instrumento musical é alterado devido às características acústicas do instrumento. Neste trabalho é utilizado o programa Matlab[®] para criar e testar filtros digitais de 2ª ordem em cascata que sintetizem sons semelhantes aos gerados por um dado instrumento. A semelhança é analisada comparando o espectro em frequência da nota produzida pelo filtro com o da nota real. Os instrumentos musicais utilizados são o violão e baixo.

Esta monografia é composta por cinco capítulos descritos sucintamente a seguir.

No Capítulo 2 são abordados alguns aspectos da teoria de processamento digital de sinais, que são importantes para tornar mais acessível o restante deste trabalho. Entre outros aspectos, discute-se sinais, Série de Fourier, amostragem, sistemas de tempo discreto e a Transformada Z.

No Capítulo 3 são abordados filtros digitais. Discutem-se, entre outros tópicos, aspectos como os pólos e zeros, ordem do filtro e condições para estabilidade.

No Capítulo 4 é mostrado passo a passo o projeto de um filtro digital correspondente ao corpo de um violão. Além disso, são apresentados os resultados da simulação desse violão, por meio de dados e funções fornecidas em (STONICK; BRADLEY, 1996). Também é simulado outro instrumento musical, o baixo. São usadas gravações do baixo Washburn® XB125, realizadas pelo autor.

No Capítulo 5 são traçadas as conclusões do trabalho.

No Apêndice A são apresentados os códigos das funções utilizadas no Matlab®.

No Apêndice B é mostrado o artigo de iniciação científica publicado nos anais do XXV Simpósio Brasileiro de Telecomunicações (SBrT 2007), realizado entre 3 e 6 de setembro de 2007. Também é mostrado o pôster apresentado neste Simpósio.

2 SINAIS E SISTEMAS DE TEMPO DISCRETO

O processamento de sinais consiste na análise, síntese ou transformação de um sinal que carrega algum tipo de informação (GIROD; RABENSTEIN; STENGER, 2003). Ele pode ser classificado em analógico ou digital.

O processamento analógico de sinais é realizado por dispositivos analógicos, como, por exemplo, diodos, resistores, capacitores e indutores. Já o digital é realizado por somadores, multiplicadores e memória e por isso é realizado através de operações numéricas (HAYKIN; VEEN, 1999).

Neste capítulo são abordados conceitos de Processamento Digital de Sinais, que são utilizados ao longo dos demais capítulos, tendo em vista tornar o trabalho mais acessível. Como existem muitas referências sobre o assunto, e.g. (BELLANGER, 2000, DINIZ; SILVA; NETTO, 2004, GIROD; RABENSTEIN; STENGER, 2003, HAYKIN; VEEN, 1999, LATHI, 1998, MITRA, 2001, OPPENHEIM; SCHAFER; BUCK, 1999, PROAKIS; MANOLAKIS, 1996), aqui é apresentado um breve resumo.

2.1 SINAIS

Sinais são funções de uma ou mais variáveis que fornecem informações sobre um determinado fenômeno (LATHI, 1998; PROAKIS; MANOLAKIS, 1996). Podem ser de tempo contínuo ou de tempo discreto. Um sinal é de tempo contínuo se a sua amplitude varia continuamente com o tempo t . Um exemplo de sinal de tempo contínuo $y(t)$, é mostrado no Gráfico 1. Ele corresponde ao som produzido quando se acionou a corda mais grave do baixo modelo Washburn® XB-125.

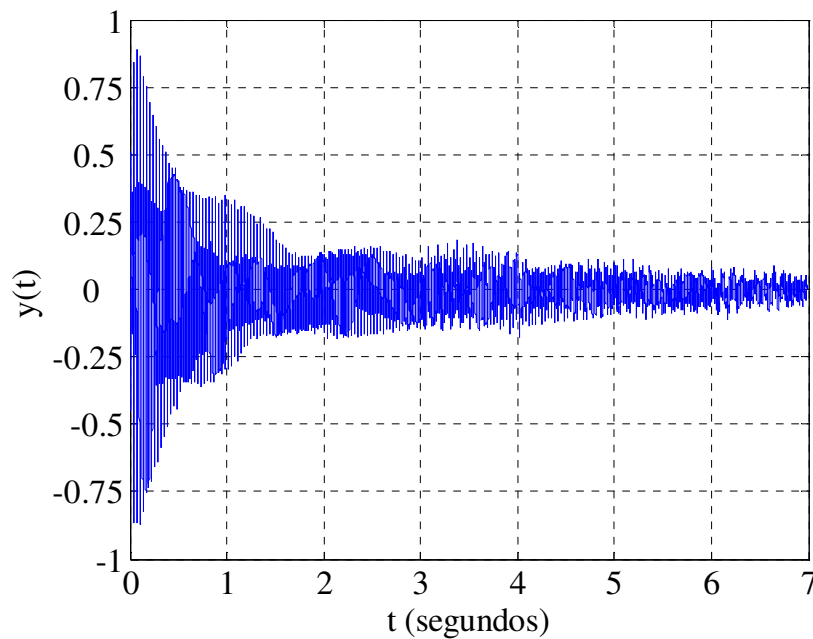


Gráfico 1 - Sinal de tempo contínuo correspondente a nota Si gravada através de um microfone.

Um sinal de tempo discreto pode ser definido em instantes isolados, separados geralmente de maneira uniforme (LATHI, 1998). Para indicar que um sinal é de tempo discreto, usa-se aqui como variável independente n ao invés de t . Além disso, usam-se colchetes ao invés de parênteses. Por exemplo, $x(t)$ é um sinal de tempo contínuo e $x[n]$ um sinal de tempo discreto. A representação gráfica de um sinal de tempo discreto é exemplificada no Gráfico 2.

Define-se a seguir a notação para alguns sinais de tempo discreto que são essenciais para o estudo de processamento de sinais.

O *sinal impulso* é notado por $\delta[n]$ e definido por:

$$\delta[n] = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases} \quad (2)$$

O *sinal degrau* é notado por $u[n]$ e definido por:

$$u[n] = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases} \quad (3)$$

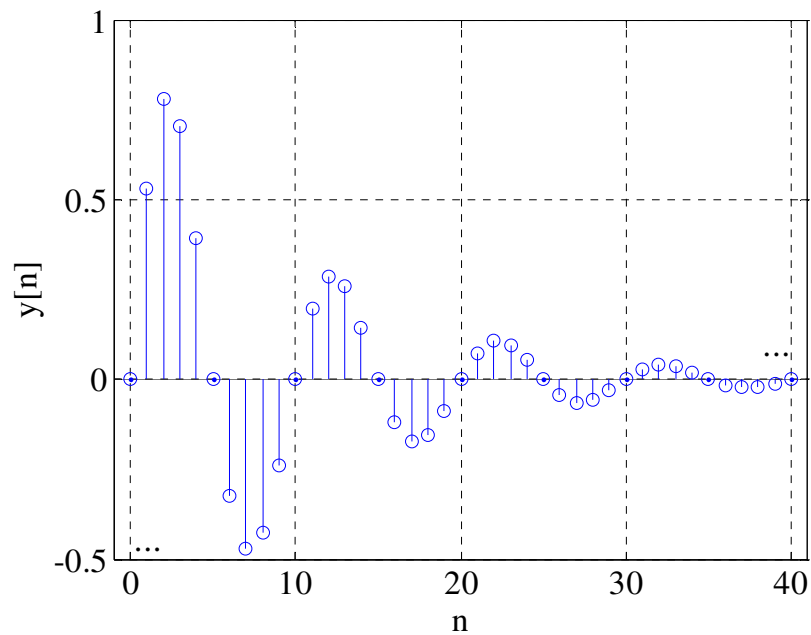


Gráfico 2 - Sinal de tempo discreto $y[n] = e^{-0.1n} \sin(n\pi/5)$.

Os *sinais exponenciais* são a multiplicação de um sinal degrau por uma exponencial, ou seja, $a^n u[n]$ (JACKSON, 1999), sendo a um número real. No Gráfico 3 são mostrados exemplos desses três sinais fundamentais.

Um sinal senoidal de tempo discreto é expresso por:

$$x[n] = A \cos(\Omega n + \theta). \quad (4)$$

Na Eq. (4), Ω é a frequência de tempo discreto, θ é o deslocamento em fase e A é a amplitude máxima do sinal. Um exemplo de sinal senoidal é mostrado no Gráfico 4.

Os sinais podem ser deslocados no tempo. Um sinal deslocado em m amostras é representado por $f[n - m]$. Quando $m < 0$ o sinal é dito *adiantado* e quando $m > 0$ o sinal é dito *atrasado*. No Gráfico 5 estão dois exemplos de sinais deslocados no tempo.

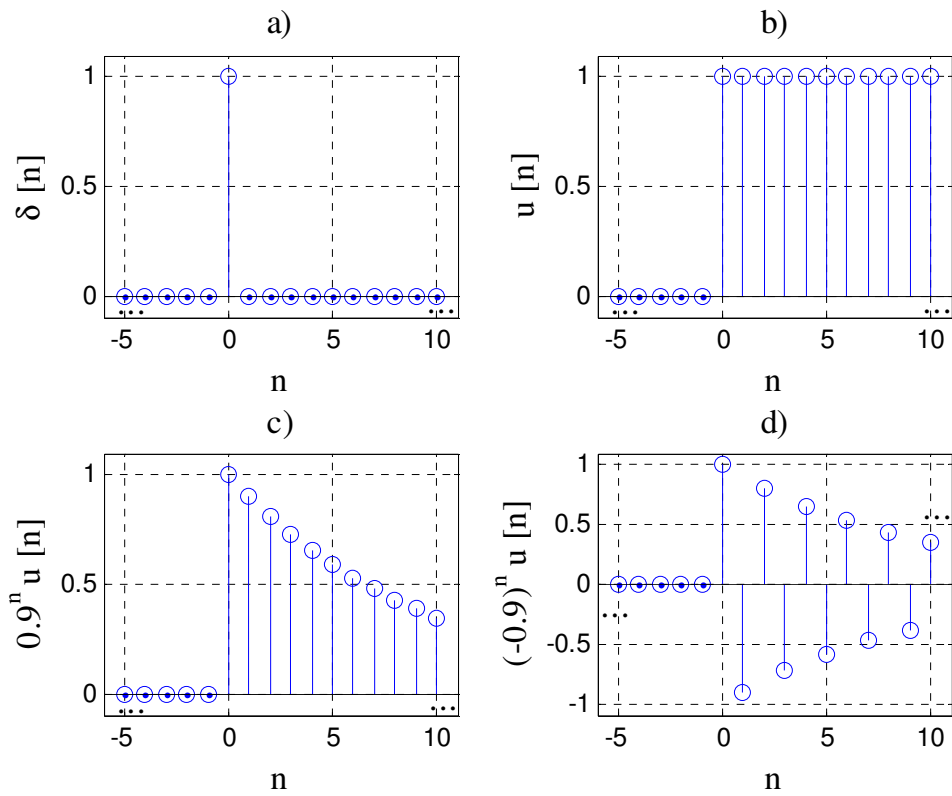


Gráfico 3 - Exemplos de sinais de tempo discreto: a) sinal impulso, b) sinal degrau, c) e d) sinais exponenciais.

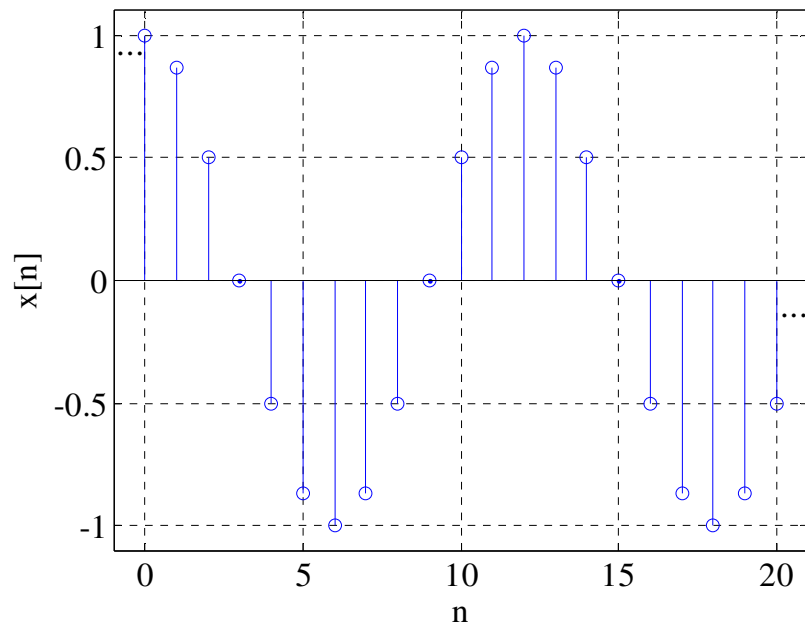


Gráfico 4 - Sinal senoidal $x[n] = \cos(2\pi n/12)$. Neste caso, $\Omega = 2\pi/12$ e $\theta = 0$.

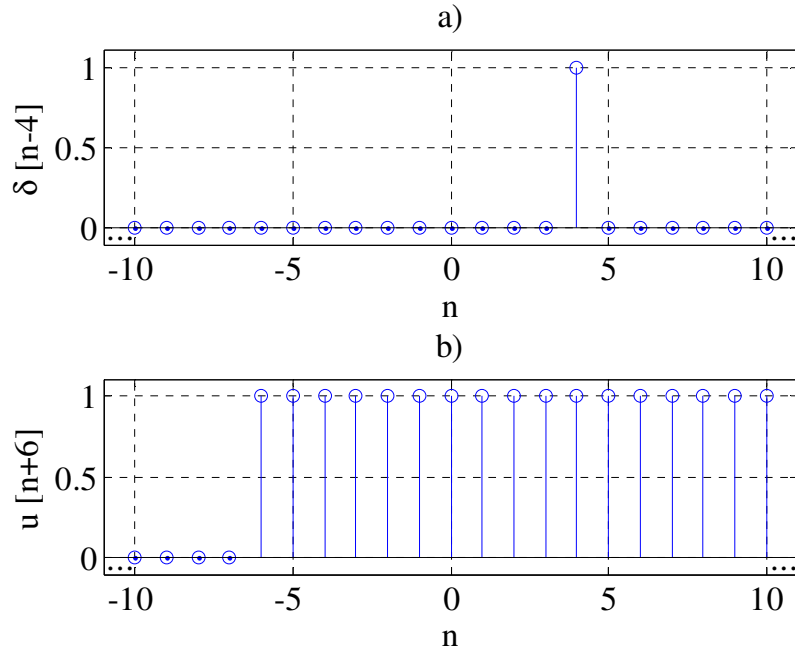


Gráfico 5 - a) Sinal impulso atrasado em quatro amostras; b) sinal degrau adiantado em seis amostras.

2.2 SÉRIE DE FOURIER

Uma senóide de tempo contínuo com frequência $n\omega_0$, sendo n inteiro, é conhecida como a n -ésima harmônica da senóide de frequência ω_0 . Pela Série Trigonométrica de Fourier, um sinal $f(t)$ periódico definido em qualquer intervalo de tempo, sendo $T_0 = 2\pi/\omega_0$ o seu período, pode ser escrito como (LATHI, 1998):

$$f(t) = a_0 + \sum_{n=1}^{\infty} (a_n \cos \omega_0 n t + b_n \sin \omega_0 n t) \quad (5)$$

Os coeficientes são calculados pelas equações seguintes (LATHI, 1998), sendo o intervalo da integral t_1 até $t_1 + T_0$ o período da função.

$$a_0 = \frac{1}{T_0} \int_{t_1}^{t_1+T_0} f(t) dt \quad (6)$$

$$a_n = \frac{2}{T_0} \int_{t_1}^{t_1+T_0} f(t) \cos n\omega_0 t dt \quad (7)$$

$$b_n = \frac{2}{T_0} \int_{t_1}^{t_1+T_0} f(t) \sin n\omega_0 t dt \quad (8)$$

O sinal dente de serra $y_{ds}(t)$, que é usado neste trabalho, tem o aspecto mostrado no Gráfico 6. A função $f(t)$ do dente de serra, para o intervalo $0 < t < T_0$, pode ser escrita como:

$$f(t) = \frac{2}{T_0} t - 1 \quad (9)$$

Realizando os cálculos para achar a Série de Fourier do dente de serra, obteve-se os coeficientes com os seguintes valores:

$$a_0 = 0, a_n = 0, b_n = -\frac{2}{\pi n} \quad (10)$$

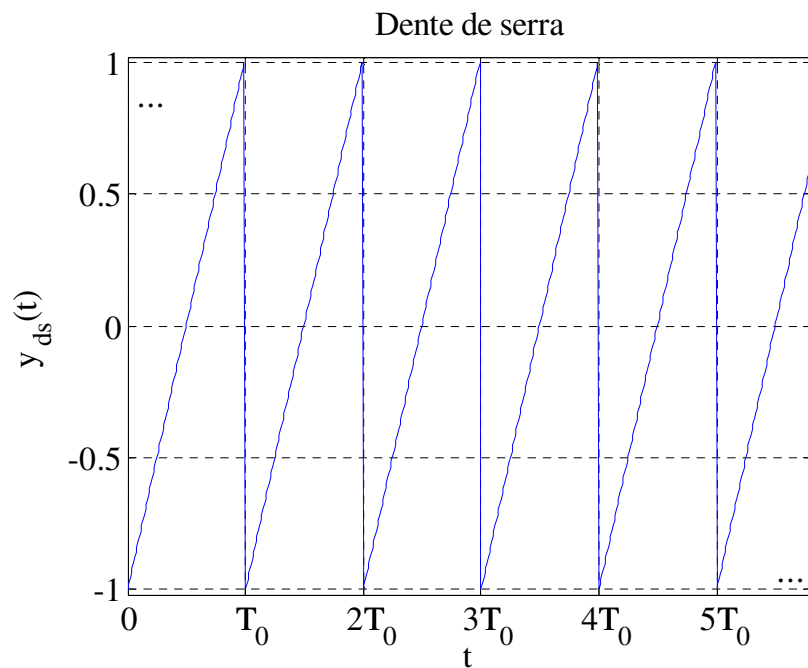


Gráfico 6 - Dente de serra com período T_0 .

Então a Série de Fourier do dente de serra é:

$$f(t) = \sum_{n=1}^{\infty} -\frac{2}{\pi n} \sin \omega_0 n t \quad (11)$$

Observando essa Série verifica-se que as harmônicas caem com n , ou seja, a segunda harmônica tem metade da amplitude da fundamental, a terceira tem um terço e assim por diante. A amplitude normalizada A_N máxima do espectro é igual a 1, pois $A_N = A(n\omega_0)/A_{máx}$, sendo n inteiro e maior que 1, $A(n\omega_0)$ a amplitude de cada harmônica e $A_{máx}$ a maior amplitude existente no espectro.

O espectro em frequência normalizado do dente de serra em função de ω_0 é mostrado no Gráfico 7.

No Gráfico 8 é mostrado a Série de Fourier do dente de serra de $f_0 = \omega_0/2\pi = 200$ Hz com $n=1, 2, 10$ e 10000 harmônicas.

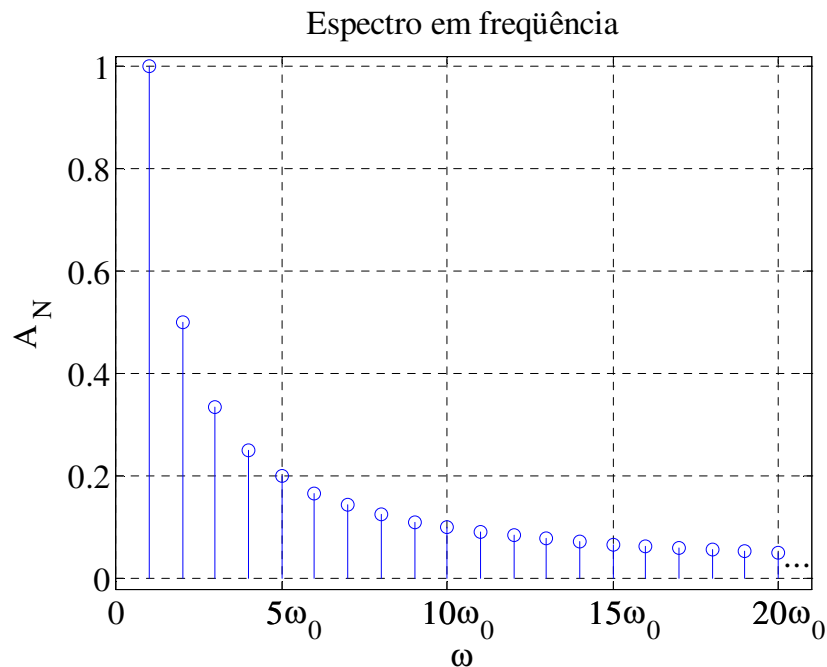


Gráfico 7 - Espectro em frequência do dente de serra com frequência ω_0 .

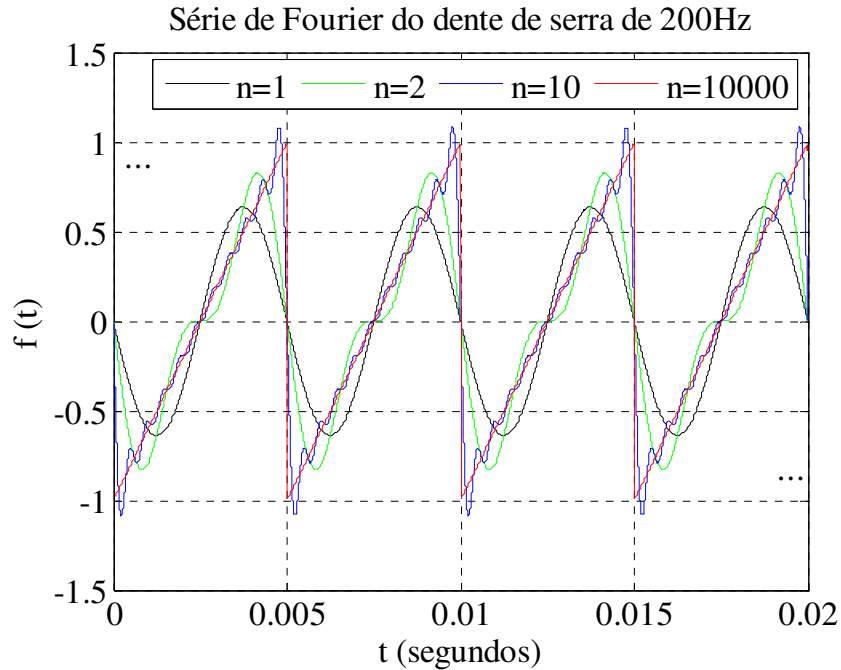


Gráfico 8 - Série de Fourier do dente de serra de 200 Hz com 1, 2, 10 e 10000 harmônicas.

2.3 AMOSTRAGEM

Amostragem é a conversão de um sinal de tempo contínuo em tempo discreto. É realizada tomando-se amostras do sinal de tempo contínuo em intervalos de tempo discretos.

Existem várias maneiras de amostrar um sinal de tempo discreto (PROAKIS; MANOLAKIS, 1996). Neste trabalho, limita-se a abordagem a amostragem uniforme, que consiste em coletar a amplitude do sinal a cada T segundos. O tempo T é chamado de *período de amostragem* e $f_s = 1/T$ é chamado de *freqüência de amostragem*. O Gráfico 9 mostra um sinal senoidal de $f = 110$ Hz com $f_s = 120$ Hz.

Para não haver perda de informação durante a amostragem é necessário que $f_s \geq 2f_{\max}$ (PROAKIS; MANOLAKIS, 1996), ou seja, a freqüência de amostragem deve ser no mínimo duas vezes a máxima freqüência f_{\max} existente no sinal a ser amostrado.

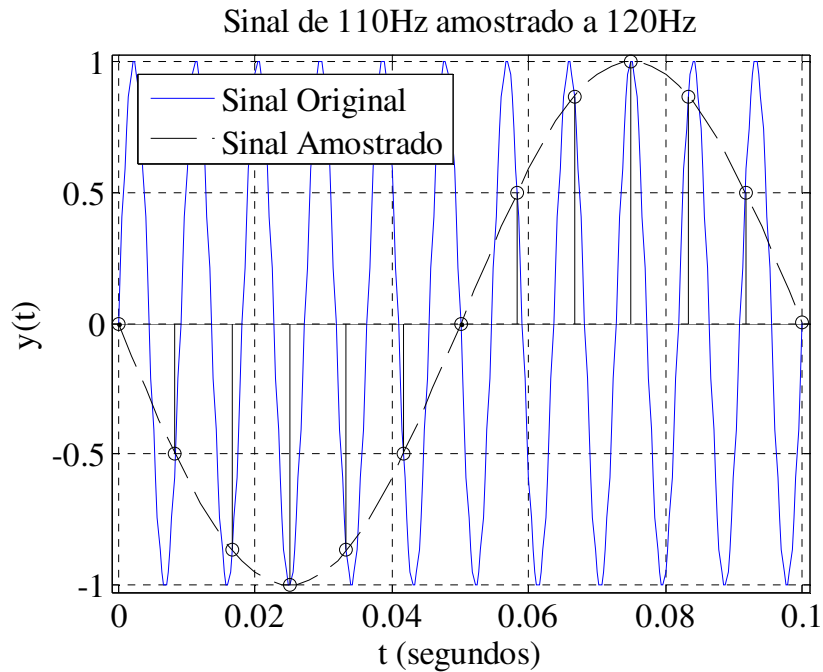


Gráfico 9 - Sinal amostrado a $f_s = 120 \text{ Hz} < 2f = 220 \text{ Hz}$.

No Gráfico 9 verifica-se que por a frequência de amostragem ser menor que duas vezes a frequência máxima do sinal, a característica do sinal é perdida. O sinal original era de 110Hz e o sinal reconstruído após a amostragem tem frequência $1/0.1s = 10 \text{ Hz}$.

2.4 SISTEMAS DE TEMPO DISCRETO

Um sistema de tempo discreto é um dispositivo que transforma um sinal de tempo discreto em outro sinal. O sinal que entra é chamado de *excitação* e o sinal de saída é a *resposta do sistema*.

No Diagrama 1 é mostrado o diagrama de blocos simplificado de um sistema de tempo discreto (PROAKIS; MANOLAKIS, 1996).

Os sistemas de tempo discreto podem ser classificados como invariantes ou variantes no tempo e lineares ou não-lineares.

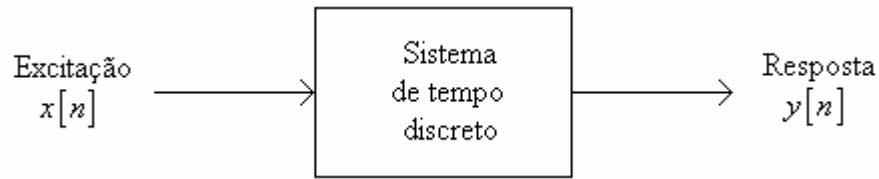


Diagrama 1 - Diagrama de blocos simplificado de um sistema de tempo discreto.

2.4.1 Invariância no tempo

Um sistema é chamado invariante no tempo quando um deslocamento no tempo do sinal de entrada provoca um deslocamento idêntico no sinal de saída. Supondo que o sinal $y[n]$ é a resposta de um sistema a uma entrada $x[n]$, se essa entrada for atrasada de k amostras, então a resposta terá que ser $y[n-k]$ para que este sistema seja invariante no tempo. Se não se comportar dessa maneira, o sistema é chamado de variante no tempo (PROAKIS; MANOLAKIS, 1996, HAYKIN; VEEN, 1999).

2.4.2 Linearidade

Um sistema é linear se este atende aos princípios da superposição e da homogeneidade. Caso contrário o sistema é chamado de não-linear.

Suponha um sistema que tenha uma resposta $y_1[n]$ para uma entrada $x_1[n]$ e uma resposta $y_2[n]$ a outra entrada $x_2[n]$. Quando a soma $x_1[n] + x_2[n]$ é a entrada desse sistema, então a resposta deve ser $y_1[n] + y_2[n]$ para que o sistema respeite o princípio da superposição.

O princípio da homogeneidade é respeitado quando, por exemplo, a saída de um sistema é $ay_1[n]$ para uma entrada $ax_1[n]$, em que a é uma constante real (PROAKIS;

MANOLAKIS, 1996).

Os sistemas de tempo discreto lineares e invariantes no tempo podem ser descritos através de equações de diferenças. Considerando $y[n]$ a saída do sistema e $x[n]$ a excitação, um sistema de tempo discreto pode ser descrito como (PROAKIS; MANOLAKIS, 1996):

$$y[n] = -\sum_{k=1}^N a_k y[n-k] + \sum_{k=0}^M b_k x[n-k] \quad (12)$$

A menos que mencionado em contrário, consideram-se aqui apenas Sistemas Lineares e Invariantes no tempo (LIT).

2.4.3 Resposta ao impulso

A resposta ao impulso $h[n]$ é a saída de um sistema LIT quando a entrada é um impulso. Esta resposta caracteriza totalmente o sistema, pois com ela é possível calcular a resposta do sistema a qualquer entrada através da soma de convolução (HAYKIN; VEEN, 1999). É por esse motivo que os sistemas LIT são muito utilizados.

A saída de um sistema LIT é a superposição ponderada de respostas ao impulso deslocadas no tempo. Isso é chamado de soma de convolução e a sua equação é mostrada na Eq. (13) (HAYKIN; VEEN, 1999). O símbolo matemático que representa a soma de convolução é o $*$.

$$y[n] = x[n] * h[n] = \sum_{k=-\infty}^{\infty} x[k] h[n-k] \quad (13)$$

2.4.4 Dizimação ou compressão no tempo

Dizimação consiste em tomar amostras a cada a amostras de um sinal de tempo discreto, sendo a inteiro. Um sinal $f_d[n] = f[2n]$ tem somente as amostras pares de $f[n]$. Esse processo também é chamado de compressão no tempo (LATHI, 1998). No Gráfico 10(b) é mostrada a dizimação do sinal do Gráfico 10(a).

2.4.5 Interpolação

Interpolação consiste em adicionar amostras a um sinal. Um sinal $f_i[n] = f[n/2]$ tem um número maior de amostras que o sinal original. Já que não existe amostra em índices fracionários, as amostras que são adicionadas têm um valor estimado a partir das demais. Na prática essas amostras podem ser iguais a média dos valores anteriores e posteriores a essa amostra (LATHI, 1998). Se $f[0] = 2$ e $f[1] = 3$, então $f_i[1] = f[0.5] = 2.5$. Um exemplo de interpolação é mostrado no Gráfico 10(c).

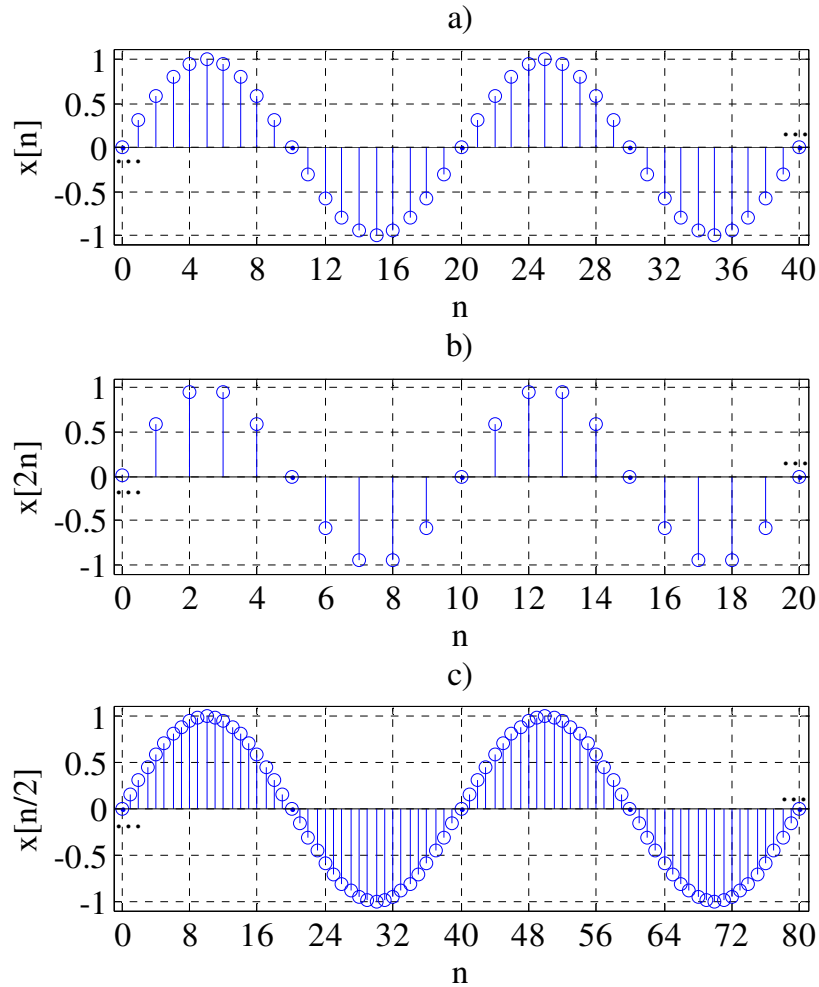


Gráfico 10 - a) Sinal original $x[n] = \sin(\pi n/10)$; b) Dizimação $x_d[n] = x[2n]$; c) Interpolação

$$x_i[n] = x[n/2].$$

2.5 TRANSFORMADA Z

A Transformada Z transforma equações de diferenças em equações algébricas, facilitando a análise do sistema de tempo discreto. A análise no domínio da frequência de um sistema de tempo discreto é baseada no fato de que a resposta de um sistema LIT a um exponencial z^k é a mesmo exponencial multiplicado por uma constante (HAYKIN; VEEN, 1999). Com isso a entrada é expressa pela soma de exponenciais z^k e a resposta do sistema será a soma das respostas do sistema para todos os exponenciais. A Transformada Z permite

representar um sinal e um sistema como uma soma de exponenciais z^k (LATHI, 1998).

A Transformada Z de um sinal $x[n]$ é definida como:

$$X(z) = \sum_{n=-\infty}^{\infty} x[n] z^{-n} \quad (14)$$

A transformada direta desse sinal pode ser expressa também como

$$x[n] \xrightarrow{Z} X(z) \quad (15)$$

A Transformada Z existe somente se a Eq. (14) converge. A região de convergência (RDC) de $X(z)$ é a faixa de valores de z na qual $X(z)$ possui um valor finito (HAYKIN; VEEN, 1999).

O número complexo z é geralmente representado por exponenciais complexas da forma $re^{j\Omega}$, com $j = \sqrt{-1}$, e é localizado em um plano complexo, denominado z . No eixo horizontal tem-se a parte real $\text{Re}(z)$ e no eixo vertical, a parte imaginária $\text{Im}(z)$. Então r corresponde a uma distância r da origem e Ω a um ângulo a partir do eixo horizontal.

O conceito de RDC pode ser melhor entendido com um exemplo. Seja $x[n] = u[n]$ e $z = re^{j\Omega}$, então através da Eq. (14) chega-se a:

$$X(re^{j\Omega}) = \sum_{n=-\infty}^{\infty} u[n] r^{-n} e^{-j\Omega n} = \sum_{n=0}^{\infty} (re^{-j\Omega})^n = 1 + (re^{-j\Omega}) + (re^{-j\Omega})^2 + (re^{-j\Omega})^3 + \dots \quad (16)$$

A Eq. (16) é uma série geométrica de tamanho infinito e sabe-se que (PROAKIS; MANOLAKIS, 1996)

$$1 + a + a^2 + a^3 + a^4 + a^5 + \dots = \frac{1}{1-a}, |a| < 1 \quad (17)$$

e, conseqüentemente, para que a Eq. (16) convirja é necessário que $|re^{-j\Omega}| < 1$. A Transformada Z existirá neste exemplo somente se $|r| < 1$ e por isso a região de convergência corresponderá a um círculo unitário, conforme mostrado no Gráfico 11.

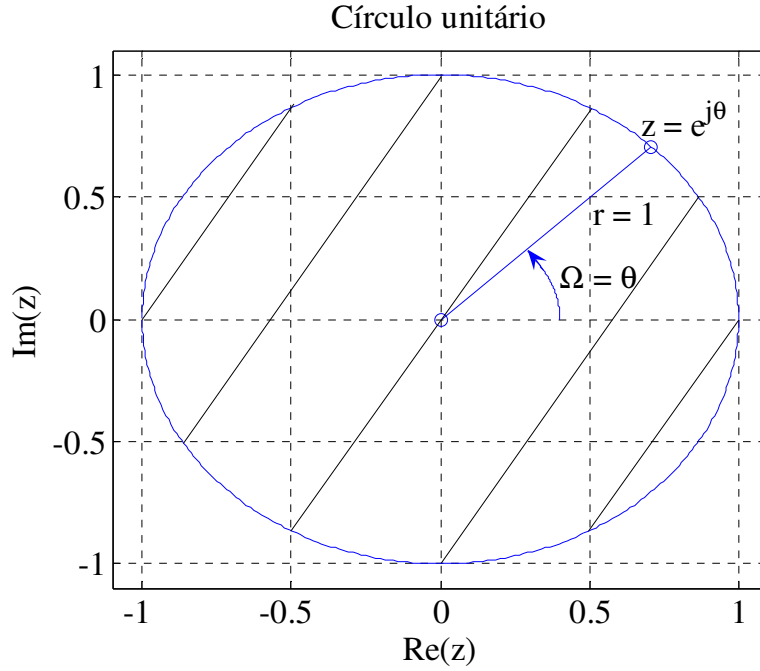


Gráfico 11 - RDC correspondente ao círculo unitário.

2.5.1 Propriedades

A propriedade da linearidade diz que a Transformada Z de uma soma de sinais é igual a soma das transformadas individuais (HAYKIN; VEEN, 1999), ou seja,

$$ax[n] + by[n] \xrightarrow{Z} aX(z) + bY(z), \quad (18)$$

sendo a e b constantes reais.

A propriedade do deslocamento no tempo diz que o deslocamento de um sinal no tempo é igual a Transformada Z do sinal multiplicada por z elevado ao valor do deslocamento (HAYKIN; VEEN, 1999). Assim,

$$x[n-d] \xrightarrow{Z} X(z) z^{-d} \quad (19)$$

A transformada da convolução de dois sinais é igual a multiplicação das Transformadas Z dos sinais (HAYKIN; VEEN, 1999).

$$x[n] * y[n] \xrightarrow{Z} X(z) Y(z) \quad (20)$$

2.5.2 Função de transferência

Sejam $X(z)$, $Y(z)$ e $H(z)$, respectivamente, a Transformada Z do sinal de entrada, de saída e da resposta ao impulso. Pela propriedade da convolução, a resposta do sistema em função da entrada e da resposta ao impulso é dada por (HAYKIN; VEEN, 1999):

$$Y(z) = H(z) X(z) \rightarrow H(z) = \frac{Y(z)}{X(z)} \quad (21)$$

Deste modo, a *função de transferência* $H(z)$ relaciona a entrada e a saída de um sistema.

2.5.3 Resposta em frequência

A resposta em frequência de um sistema $H(z)$ é obtida substituindo-se z por $e^{j\Omega}$ na Eq. (21). Constata-se que $e^{j\Omega}$ tem periodicidade igual a 2π (PROAKIS; MANOLAKIS, 1996), o que implica que é necessário especificá-la apenas para uma volta da circunferência unitária.

Comumente se representa a resposta em frequência através de dois gráficos: um do módulo e outro da fase.

Por exemplo, um sistema com função de transferência

$$H[z] = \frac{2 - 2z^{-1}}{5} \quad (22)$$

tem resposta em frequência

$$H(e^{j\Omega}) = \frac{2 - 2e^{-j\Omega}}{5} = \frac{2}{5} e^{-\frac{j\Omega}{2}} \left(e^{\frac{j\Omega}{2}} - e^{-\frac{j\Omega}{2}} \right) = \frac{2}{5} e^{-\frac{j\Omega}{2}} 2j \sin\left(\frac{\Omega}{2}\right) = \frac{4}{5} e^{-\frac{j\Omega}{2} + \frac{j\pi}{2}} \sin\left(\frac{\Omega}{2}\right) \quad (23)$$

O módulo $\left|H(e^{j\Omega})\right|$ e a fase $\angle H(e^{j\Omega})$ da resposta em frequência são:

$$\begin{aligned}\left|H(e^{j\Omega})\right| &= \frac{4}{5} \sin \frac{\Omega}{2} \\ \angle H(e^{j\Omega}) &= e^{\frac{-j}{2}(\Omega-\pi)}\end{aligned}\quad (24)$$

O Gráfico 12 mostra o módulo e a fase da resposta em frequência deste sistema. A frequência está normalizada, ou seja, a frequência de tempo discreto está dividida por π .

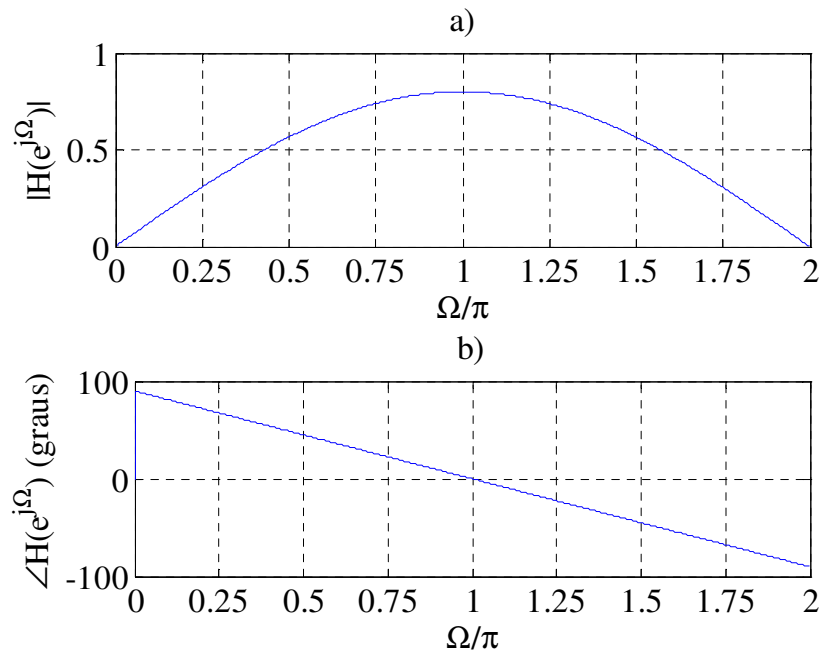


Gráfico 12 - Resposta em frequência do sistema da Eq. (22): a) em módulo; b) em fase.

3 FILTROS DIGITAIS

Neste capítulo são abordadas as características dos filtros digitais que são utilizadas ao longo do próximo capítulo para realizar as simulações. É utilizado também o Matlab[®] para mostrar o que acontece quando se varia as características do filtro.

Filtros digitais são sistemas de tempo discreto que transformam um sinal. Existem os filtros seletivos em frequência, que deixam passar determinadas faixas de frequência e removem outras faixas de frequência (HAYKIN; VEEN, 1999). Se forem conhecidas as faixas de frequência desejadas na saída de um filtro, é possível projetar um filtro com essas características.

Um filtro digital pode ser expresso por uma equação de diferenças. Para transformar para o domínio da Transformada Z , usa-se as propriedades discutidas na Seção 2.5.1. Por exemplo, para transformar um filtro dado por

$$y[n+2] - 0.71y[n+1] + 0.25y[n] = 2x[n+1] \quad (25)$$

para o domínio Z é necessário aplicar as propriedades da linearidade e do deslocamento no tempo, Eqs. (18) e (19). Então, a equação do filtro no domínio Z fica

$$Y(z)z^2 - 0.71Y(z)z^1 + 0.25Y(z) = 2X(z)z^1 \quad (26)$$

e, conseqüentemente, a sua função de transferência é:

$$H(z) = \frac{2z}{z^2 - 0.71z + 0.25} \quad (27)$$

3.1 PÓLOS E ZEROS

Pólos de $H(z)$ são valores de z tais que $H(z) = \infty$. E os zeros são valores de z tais que $H(z) = 0$.

Uma função de transferência pode ser descrita como (LATHI, 1998):

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_M z^{-M}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_N z^{-N}} \quad (28)$$

Os zeros na Eq. (28) são as raízes do numerador e os pólos são as raízes do denominador.

Por exemplo, os pólos da função de transferência da Eq. (27) são $p_1 = 0.3550 + j0.3521$, $p_2 = 0.3550 - j0.3521$ e o zero é $z = 0$. A localização dos pólos e zeros no plano z complexo é apresentada no Gráfico 13. Os pólos são representados por um “x” e o zero por um “o”.

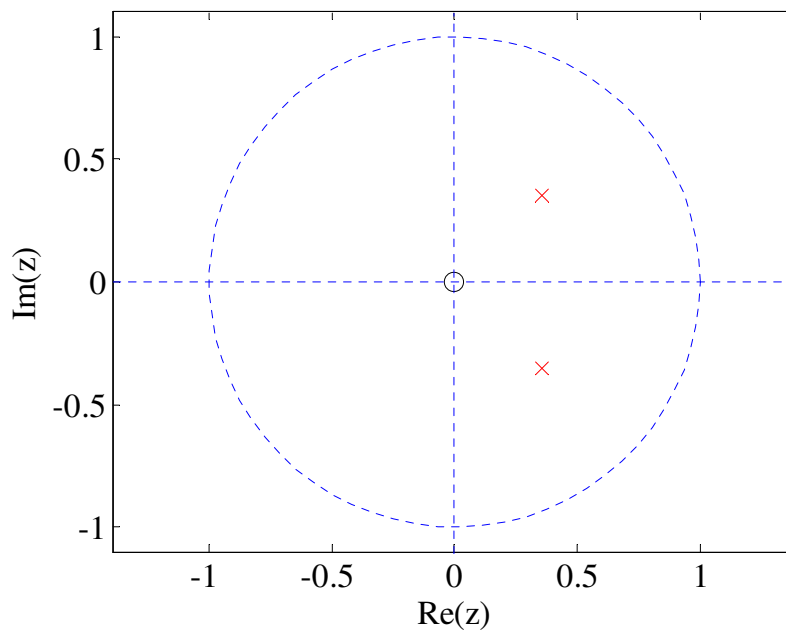


Gráfico 13 - Localização dos pólos e zeros no plano z complexo.

3.2 FILTROS EM CASCATA

Filtros digitais podem ser interligados em cascata por alguns motivos (STONICK; BRADLEY, 1996). Já que a precisão de um computador é finita, podem ocorrer diferenças entre as constantes do filtro desejado e as representadas pelo computador,

provocando diferenças significativas em filtros de ordem elevada (STONICK; BRADLEY, 1996). É por isso que nas simulações utilizam-se filtros de segundo grau em cascata.

Considerando dois filtros LIT $H_1(z)$ e $H_2(z)$ em cascata, a função de transferência do sistema total é $H(z) = H_1(z)H_2(z)$. O Diagrama 2 mostra como este sistema é representado em forma de diagrama de blocos.

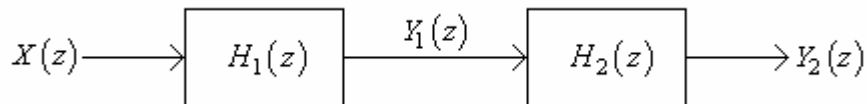


Diagrama 2 - Filtros em cascata.

3.3 CONDIÇÕES PARA ESTABILIDADE

Quando um pólo p_k está dentro da circunferência unitária ($|p_k| < 1$), ele contribui com um termo exponencialmente decrescente para a resposta ao impulso. Uma resposta impulsiva estável não contém termos exponenciais crescentes, pois assim a resposta ao impulso não seria finita. Isso implica que os pólos têm que estar dentro da circunferência unitária para que o filtro seja estável (HAYKIN; VEEN, 1999).

No sistema composto por dois filtros, conforme visto no Diagrama 3, deseja-se saber quais são as condições para que ambos os filtros sejam estáveis. Para que este sistema seja estável, é necessário que todos os pólos estejam dentro da circunferência unitária. Por isso, tanto os pólos como os zeros de $H(z)$ tem que ter módulo menor do que um.

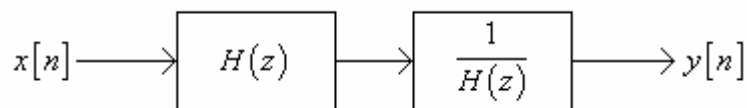


Diagrama 3 - Sistema $H(z)$ em cascata com o sistema $1/H(z)$.

3.4 FREQUÊNCIAS DE TEMPO CONTÍNUO E DISCRETO

A relação entre as frequências de tempo contínuo f e tempo discreto F é $F = f/f_s$, em que f_s é a frequência de amostragem. Já que $\omega = 2\pi f$ e $\Omega = 2\pi F$, então:

$$\omega = \Omega f_s \quad (29)$$

No círculo unitário, a frequência de amostragem f_s corresponde ao $\Omega = 2\pi$. Se a frequência de amostragem for de 44.1 kHz, então as frequências de tempo contínuo correspondentes a $\Omega_1 = 0.5\pi$ e $\Omega_2 = 2\pi/100$ são $f_1 = 11025 \text{ Hz}$ e $f_2 = 441 \text{ Hz}$, respectivamente. Verifica-se com isso que é importante o conhecimento da frequência de amostragem para o projeto de filtros, pois com ela é possível saber que valores de Ω correspondem à frequência f de projeto que se deseja cortar ou passar.

3.5 ORDEM

A ordem de um filtro corresponde a maior potência de z , considerando a menor potência igual a zero. Por exemplo, o filtro

$$H[z] = \frac{2 + z^{-1} - z^{-2} + 2z^{-3}}{1 - 2z^{-1} + 3z^{-2} + z^{-3}} \times \frac{z^3}{z^3} = \frac{2 - z + z^2 + 2z^3}{1 + 3z - 2z^2 + z^3} \quad (30)$$

é de 3ª ordem.

A função de transferência $H(z)$, sendo $z = e^{j\Omega}$, de um filtro de 2ª ordem pode ser expressa pela Eq. (31), em que cada parênteses corresponde a uma função de 1ª ordem.

$$H(e^{j\Omega}) = \frac{(1 - bz^{-1}e^{j\Omega_z})(1 - bz^{-1}e^{-j\Omega_z})}{(1 - az^{-1}e^{j\Omega_p})(1 - az^{-1}e^{j\Omega_p})} \quad (31)$$

Os módulos dos zeros e dos pólos são b e a , respectivamente. Nessa função de transferência os dois pólos $p_{1,2} = a\angle\pm\Omega_p$ são conjugados e os dois zeros $z_{1,2} = b\angle\pm\Omega_z$ também. No Gráfico 14 é apresentado a localização dos pólos e zeros no plano z complexo, mostrando onde estão localizados cada um dos parâmetros.

Aplicando a distributiva e simplificando a Eq. (31) chega-se a função de transferência mostrada na Eq. (32), que será usada na próxima seção.

$$H[z] = \frac{1 - 2bz^{-1}\cos\Omega_z + b^2z^{-2}}{1 - 2az^{-1}\cos\Omega_p + a^2z^{-2}} \quad (32)$$

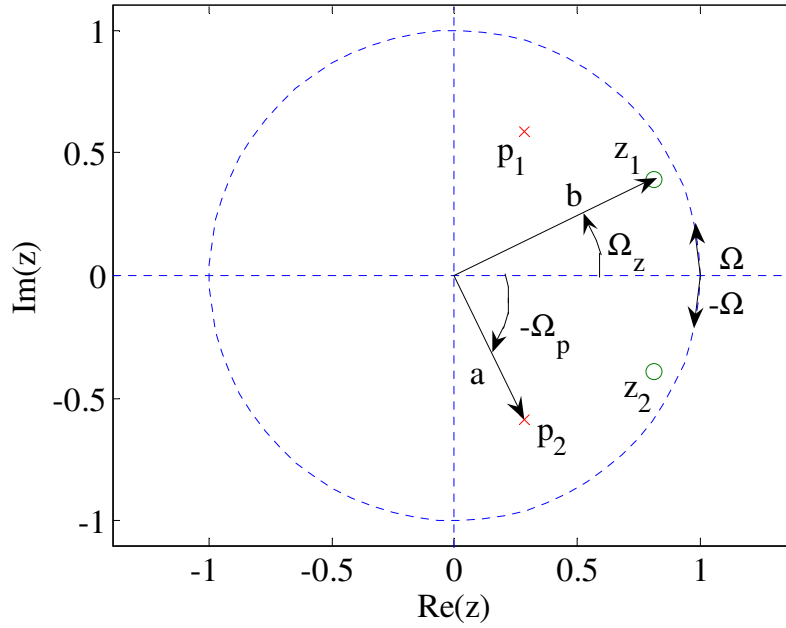


Gráfico 14 - Localização no círculo unitário de $p_{1,2} = a\angle\pm\Omega_p$ e $z_{1,2} = b\angle\pm\Omega_z$.

3.6 VARIAÇÃO DA LOCALIZAÇÃO DOS PÓLOS E ZEROS

A alteração provocada pela variação dos pólos e zeros é analisada a seguir primeiramente mantendo-se fixo os pólos e variando-se os zeros. Tanto os pólos como os zeros estão na mesma linha radial, ou seja, $\Omega_p = \Omega_z = \pi/6$. O módulo dos zeros varia de

$b = 0.4$ até $b = 2$ e o módulo do pólo se manteve em $a = 0.2$.

A resposta em frequência variando b é mostrada no Gráfico 15.

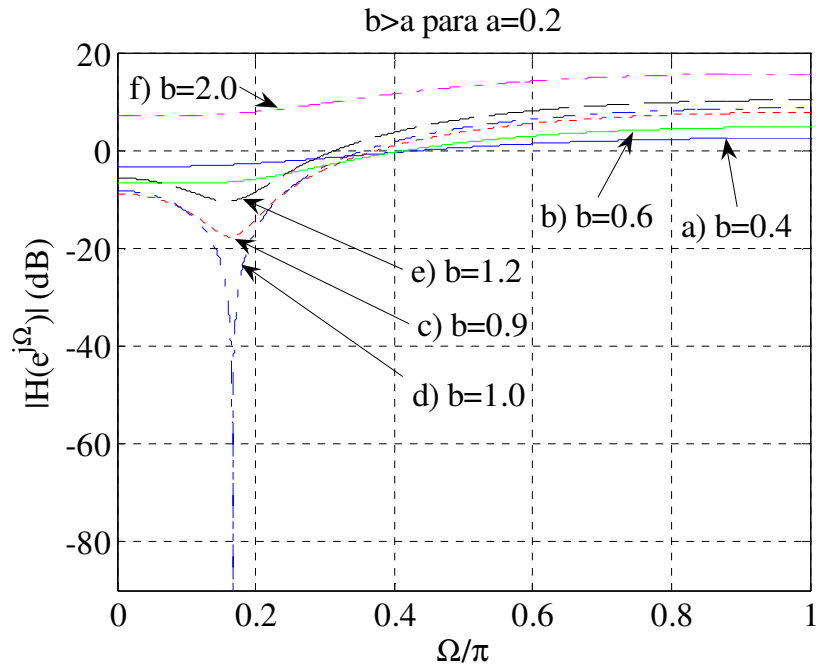


Gráfico 15 - Módulo da resposta em frequência para: a) $b = 0.4$, b) $b = 0.6$, c) $b = 0.9$, d) $b = 1.0$, e) $b = 1.2$ e f) $b = 2.0$.

Analisando o Gráfico 15, verifica-se que o filtro rejeita-faixa. Neste caso de vale anti-ressonante, para existir ganho mínimo é necessário existir duas frequências que tenham 3 dB a mais que o ponto de ganho mínimo. Cada uma dessas duas frequências se localiza a direita e a esquerda do ponto de ganho mínimo. A banda é a diferença entre essas duas frequências. Para $b = 0.9$, por exemplo, o ganho mínimo é -18 dB e por isso a banda está compreendida entre as duas frequências que têm ganho igual a -15 dB.

No Gráfico 15, verifica-se que começa a existir ganho mínimo a partir de um pouco mais de $b = 0.6$ e este diminui até $b = 1.0$. Depois o ganho mínimo vai aumentando até não existir mais ganho mínimo. Com relação a banda, esta diminui a partir de um pouco mais de $b = 0.6$ e volta a aumentar a partir de $b = 1.0$. Isto pode ser melhor observado no Gráfico 16, em que consta o comportamento do ganho mínimo G do vale anti-ressonante e da banda

B_ω com a variação de b para $a=0.2$ e $\Omega = \Omega_p = \Omega_z = \pi/6$. Valor igual a zero significa que não existe banda e ganho mínimo.

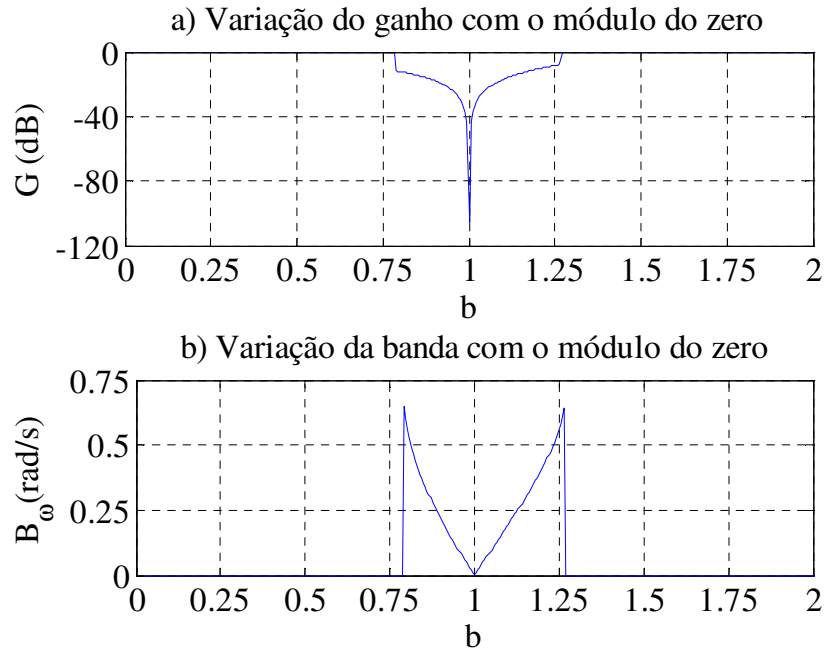


Gráfico 16 - a) Ganho mínimo e b) banda variando o zero b . Foi usado $a=0.2$ e $\Omega = \pi/6$.

A seguir, analisa-se como a variação do pólo afeta a resposta em frequência. Fixou-se o zero em $b=0.2$ e os pólos variaram de $a=0.4$ até $a=1.0$. O valor de a não pode ser maior que um porque desse modo o filtro fica instável. A resposta em frequência mostrada no Gráfico 17 corresponde a $a=0.4$, 0.6 , 0.9 e 1.0 com $b=0.2$ e $\Omega_F = \pi/6$. O Gráfico 18 mostra a variação da banda B_ω e do ganho máximo G do vale ressonante com a , para $b=0.2$ e $\Omega = \Omega_p = \Omega_z = \pi/6$.

No Gráfico 17 verifica-se que o filtro é do tipo passa-baixas. Neste caso de pico ressonante, para existir ganho máximo é necessário existir duas frequências que tenham 3dB a menos que o ponto de ganho máximo. Até $a=0.6$ não existe ganho máximo nem banda. Já a partir de um pouco menos de $a=0.9$ o ganho máximo aumenta, até chegar a $a=1.0$. A banda faz o processo inverso, diminui com o aumento de a . Isto pode ser melhor

observado no Gráfico 18.

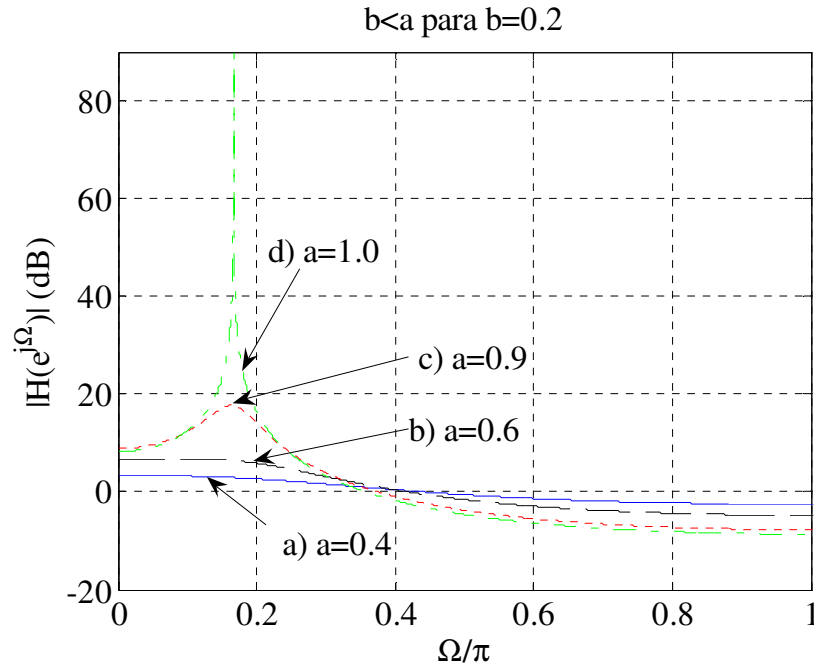


Gráfico 17 - Módulo da resposta em frequência para: a) $a = 0.4$, b) $a = 0.6$, c) $a = 0.9$, d) $a = 1.0$.

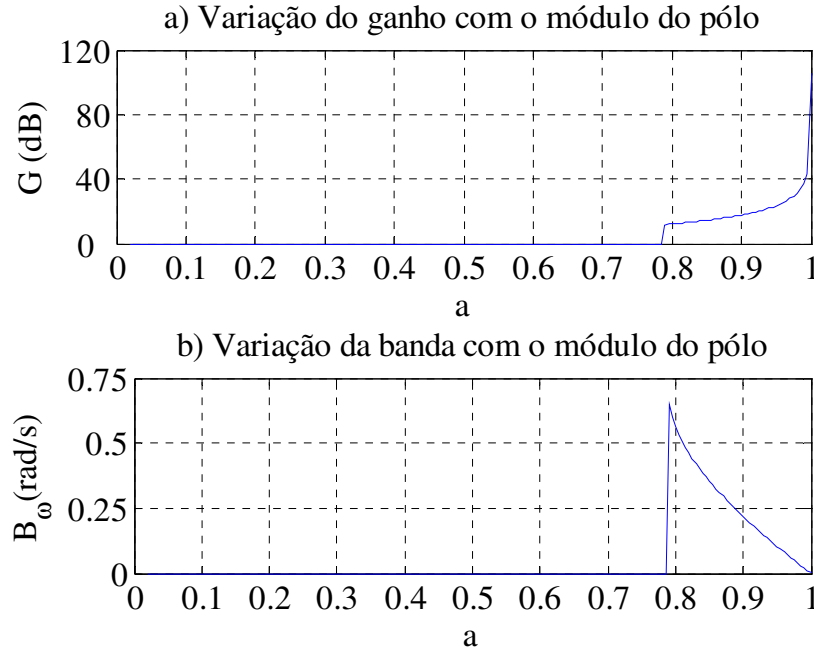


Gráfico 18 - a) Ganho máximo e b) banda variando o pólo a . Foi usado $b = 0.2$ e $\Omega = \pi/6$.

3.7 SOLUÇÃO PARA FILTROS INSTÁVEIS

Em filtros instáveis pode-se obter o gráfico da resposta em frequência, porém esta não existe, pois os pólos estão fora da circunferência unitária. Caso um filtro projetado tenha $|p| > 1$, uma boa solução é manter o Ω e mudar a magnitude para o inverso da magnitude original (STONICK; BRADLEY, 1996).

Por exemplo, suponha que o filtro $H(e^{j\Omega})$ do Diagrama 3, tenha seus pólos localizados em $p_{1,2} = \sqrt{2}/2 \angle \pm 45^\circ$ e zeros localizados em $z_{1,2} = 1.12 \angle \pm 26.6^\circ$. Para tornar o sistema $1/H(e^{j\Omega})$ estável, mas com o pico ressonante na mesma frequência, o novo módulo do pólo desse sistema pode ser $p_{1,2}^{novo} = 1.12^{-1} \angle \pm 26.6^\circ$. O sistema com o novo pólo tem uma resposta em frequência muito parecida com o que teria o filtro instável, conforme mostrado no Gráfico 19, em que $H_{ii}(e^{j\Omega})$ e $H_{ie}(e^{j\Omega})$ são as respostas em frequência dos filtros inverso instável e estável, respectivamente.

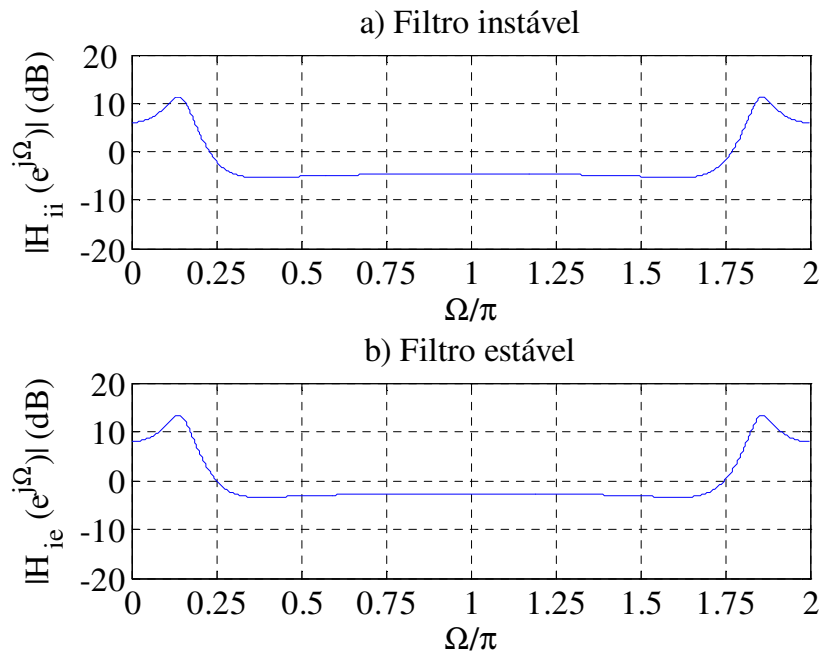


Gráfico 19 - Módulo da resposta em frequência do filtro inverso: a) instável; b) estável.

4 SIMULAÇÕES

Neste capítulo, são apresentadas e comentadas as simulações relacionadas aos instrumentos musicais em questão: o violão e o baixo. A principal referência utilizada é (STONICK; BRADLEY, 1996), tanto na parte bibliográfica, como na parte experimental. As funções baseadas em (STONICK; BRADLEY, 1996) são apresentadas no Apêndice A. Antes de apresentar as simulações, discute-se de forma abreviada sobre sintetizadores.

4.1 SINTETIZADORES

Sintetizadores, como os teclados, são capazes de simular vários instrumentos musicais. Estes são muito usados pelas bandas, pela sua versatilidade e qualidade razoável, pois simulam vários instrumentos musicais em um só equipamento e permitem adicionar instrumentos diferentes dos usados pela banda.

A maior parte dos sintetizadores são baseados em dois modelos: (i) da amostragem e (ii) da modelagem física (STONICK; BRADLEY, 1996).

No modelo baseado na amostragem são usadas amostras digitais das gravações do instrumento musical. Toda vez que é pressionada uma tecla, é reproduzida a nota correspondente. Porém, se não tiver essa nota guardada, é realizado uma mudança de tom da nota mais próxima da desejada, podendo gerar um problema, pois isso altera algumas características fundamentais do áudio, afetando a originalidade do instrumento. Esse processo é feito por dizimação e interpolação. A qualidade do sintetizador pode melhorar se for amostrada uma maior quantidade de notas de cada instrumento musical, mas para isso é necessário um espaço de armazenamento maior, o que o torna mais caro (STONICK; BRADLEY, 1996).

Na modelagem física, os instrumentos são caracterizados matematicamente e, inclusive, as não-linearidades são consideradas (STONICK; BRADLEY, 1996).

O modelo que foi utilizado neste trabalho chama-se Síntese Musical Recursiva Excitada por Código (CERMS – *Code-Excited Recursive Music Synthesis*). Este modelo apresenta boa expressividade e alta qualidade acústica, já que é uma modelagem física do instrumento baseada em um modelo computacional (STONICK; BRADLEY, 1996).

A idéia básica deste modelo é separar a modelagem da excitação da modelagem do corpo do instrumento, mudando o tom sem afetar a resposta característica do corpo (STONICK; BRADLEY, 1996).

A modelagem do corpo é feita neste trabalho por meio de filtros digitais, que podem ser facilmente projetados.

4.2 SIMULAÇÃO DE UM VIOLÃO

São mostradas aqui todas as etapas realizadas no Matlab para a criação e a simulação de um filtro digital correspondente ao corpo de um violão, em que as funções e os sinais foram fornecidos por (STONICK; BRADLEY, 1996). As funções foram modificadas para melhor apresentação neste trabalho.

4.2.1 Projeto do corpo do violão

A função `criafiltro`, que é baseada na função manual fornecida em (STONICK; BRADLEY, 1996) e é listada no item 1 do Apêndice A, cria filtros de 2ª ordem em cascata, podendo-se modificar as características do filtro selecionando os pontos de máximo ou mínimo na figura que apresenta a resposta em frequência do corpo do instrumento

em questão, o violão.

Também é fornecido em (STONICK; BRADLEY, 1996) um sinal que corresponde a resposta ao impulso do violão, chamado *impulso*. São usados os últimos 2049 pontos de um total de 4097 para o projeto do filtro digital, pois este sinal tem um pico no ponto 2049 e deste jeito o sinal se assemelha mais a um impulso.

A função `criafiltro` tem o seguinte formato:

```
[numo,deno]=criafiltro(impulso,num,den)
```

As variáveis *numo* e *deno* são os coeficientes do numerador e denominador, respectivamente, do filtro digital a ser criado. A variável *impulso* é a resposta ao impulso do corpo do instrumento e *num* e *den* são os coeficientes do numerador e denominador de um filtro possivelmente já criado, pois essa função pode ser usada novamente caso se deseje melhorar a resposta em frequência. Quando este não for o caso, pode-se colocar no lugar de cada uma dessas duas variáveis, colchetes [].

É importante ressaltar que essa função cria filtros digitais de 2º grau em cascata, correspondendo cada linha de *num*, *den*, *numo* e *deno* a um filtro de 2º grau.

Ao executar essa função as seguintes informações aparecem na tela:

```
>> [numo,deno]=criafiltro(impulse,[],[])
Digite a frequência de amostragem: 44100
Calculando a resposta em frequência do instrumento....
Calculando a resposta em frequência do filtro...
Intervalo de frequência atual:
      0      45000
```

A seguir, é apresentado o Gráfico 20, que mostra o módulo da resposta em frequência do corpo do violão e do filtro inicial. O módulo da resposta em frequência do corpo do violão foi obtido através da Transformada Rápida de Fourier (OPPENHEIM; SCHAFER; BUCK, 1999). Depois é pedida a faixa de frequência que se deseja visualizar. Usou-se o intervalo $0 \leq f \leq 2500$ Hz, pois as maiores magnitudes estão nesse intervalo.

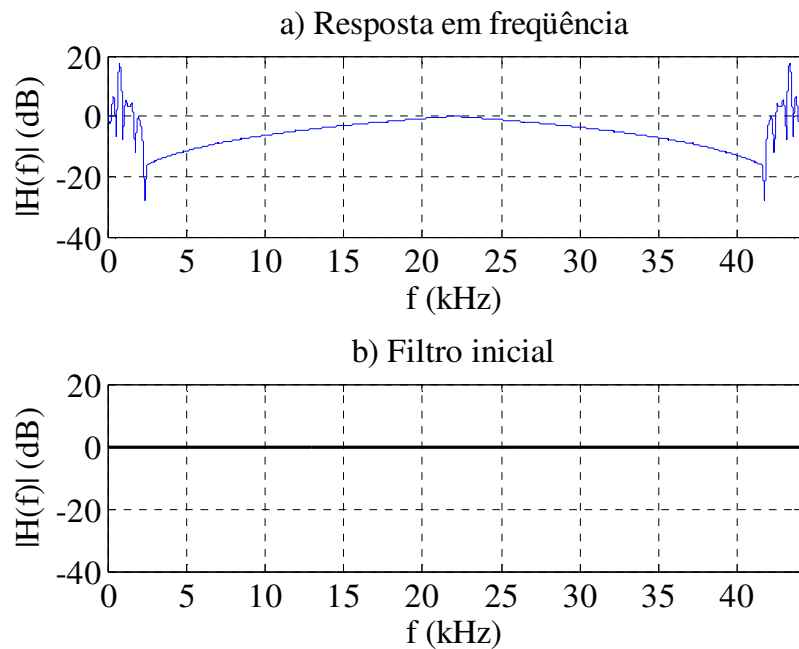


Gráfico 20 - Módulo da resposta em frequência: a) do corpo do violão; b) do filtro inicial.

Frequência mínima (Hz) visualizada no gráfico ([] para não modificar): 0
 Frequência máxima (Hz) visualizada no gráfico ([] para não modificar): 2500
 Será usado o intervalo de frequência [0.000000,2500.000000] Hz a partir de agora.

A seguir pergunta-se se o filtro atual $H(f)=1$ deve ser refeito, igualando o numerador a um e o denominador também. Neste momento não faz diferença, pois o filtro já é igual a 1.

Limpar o conteúdo do numerador e do denominador ($H[z]=1$) [n]? n
 Clique no gráfico para modificar o seu filtro atual.

Então é necessário escolher um pico para o filtro. Isso é feito com o uso do mouse como mostrado no Gráfico 21.

Após escolher um pico para o filtro, é informada a frequência f_0 e a magnitude do ponto que foi escolhido.

Frequência (Hz): 779.989919
 Magnitude (dB): 17.640000

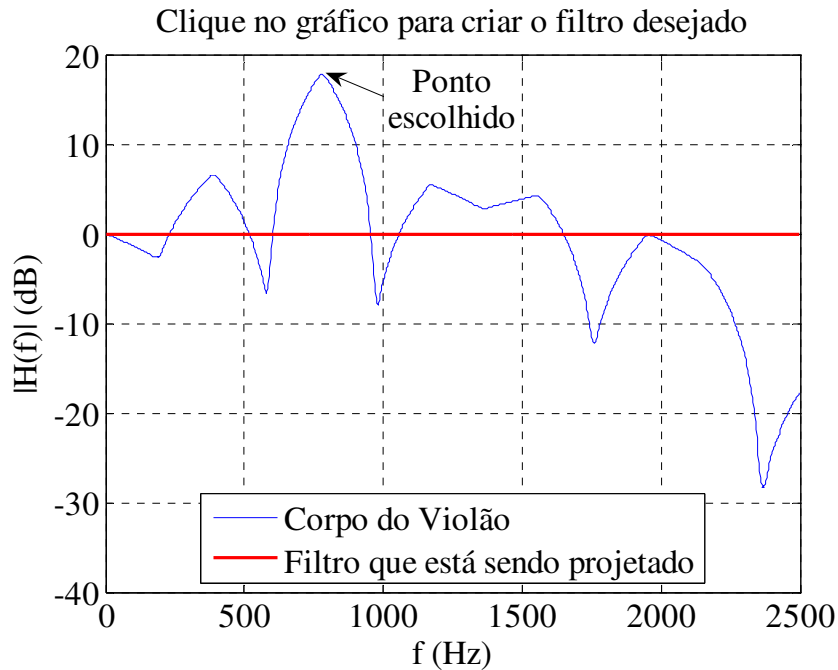


Gráfico 21 - Módulo da resposta em frequência do corpo do violão e do filtro que está sendo projetado

$$H(f) = 1.$$

Também é informada a banda B e a diferença de magnitude ΔG do ponto escolhido para o filtro atual. A banda mostrada é calculada pela diferença de 3 dB entre o pico ressonante escolhido e o ponto mais próximo à direita.

Diferença da magnitude (dB): 17.640000
Banda (Hz): 67.291260

Depois é pedido para confirmar se o ponto é aceitável ou não e caso seja, é perguntado se deseja mudar o ganho e a banda. Os valores parecem razoáveis.

Ponto aceitável [s]? s
Com ganho de 17.640000 dB [s]? s
Com banda de 67.291260 Hz [s]? s

A seguir, é mostrado o módulo da resposta em frequência do filtro que acabou de ser projetado, perguntando se os valores estão bons. No Gráfico 22 é apresentada essa resposta.

Os dados do filtro ΔG , f_0 e B se relacionam com as variáveis k , B_ω e Ω_0 da

função de transferência do filtro pelas seguintes equações:

$$k = 10^{\frac{\Delta G}{10}} \quad (33)$$

$$B_{\omega} = \frac{2\pi B}{f_s} \quad (34)$$

$$\Omega_0 = \frac{2\pi f_0}{f_s} \quad (35)$$

A variável k é o ganho em vezes, B_{ω} é a banda em rad/s e Ω_0 é a frequência de ressonância (pico) ou de anti-ressonância (vale) de tempo discreto.

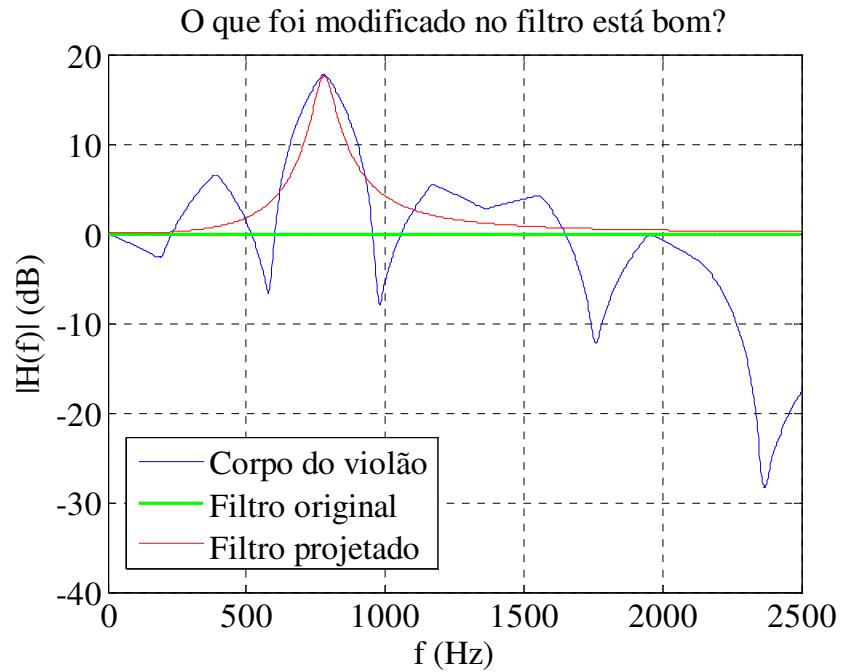
O filtro calculado pela função `criafiltro` tem a seguinte função de transferência (STONICK; BRADLEY, 1996):

$$H(z) = \frac{\frac{1 + k \tan\left(\frac{B_{\omega}}{2}\right)}{1 + \tan\left(\frac{B_{\omega}}{2}\right)} - \frac{2 \cos(\Omega_0)}{1 + k \tan\left(\frac{B_{\omega}}{2}\right)} z^{-1} + \frac{1 - k \tan\left(\frac{B_{\omega}}{2}\right)}{1 + \tan\left(\frac{B_{\omega}}{2}\right)} z^{-2}}{1 - \frac{2 \cos(\Omega_0)}{1 + \tan\left(\frac{B_{\omega}}{2}\right)} z^{-1} + \frac{1 - \tan\left(\frac{B_{\omega}}{2}\right)}{1 + \tan\left(\frac{B_{\omega}}{2}\right)} z^{-2}} \quad (36)$$

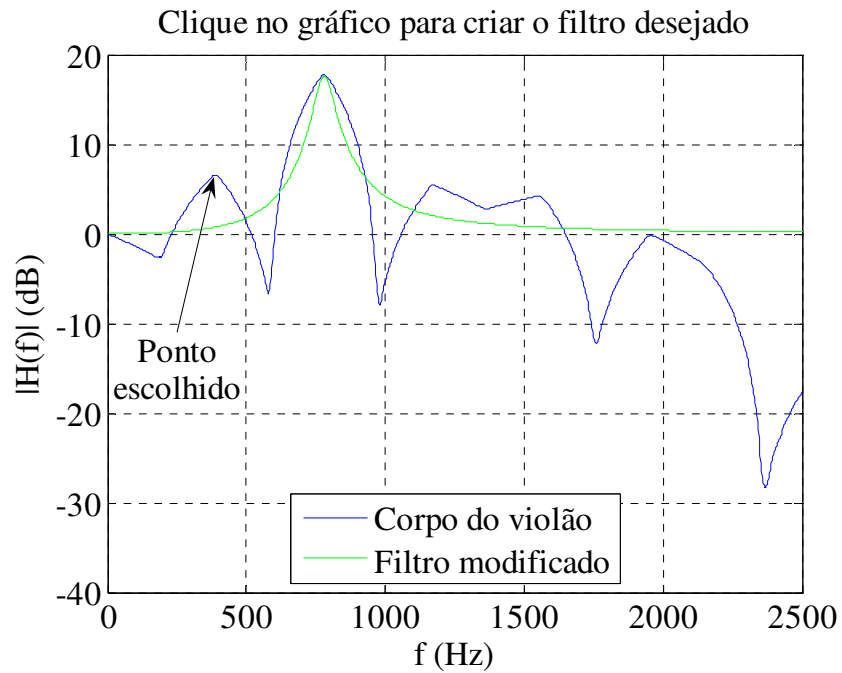
Esses pares de pólos/zeros são bons [s]? s
Integrando esses pares de pólos e zeros no filtro.

O filtro criado tem banda igual a B Hz, ou seja, a diferença entre as duas frequências que têm 3 dB a menos que o ganho máximo é igual a B Hz. É apresentado depois o módulo da resposta em frequência do filtro calculado e o programa pergunta se deseja adicionar mais algum filtro em cascata.

Vai ser criado mais um filtro de 2º grau por o filtro ainda não abranger a outra faixa de frequência. Agora é necessário escolher outro ponto. No Gráfico 23 é mostrado o ponto escolhido e no Gráfico 24 é apresentado o módulo da resposta em frequência atualizada após escolher esse ponto.



Adicionar outro ponto [s]? s
 Clique no gráfico para modificar o seu filtro atual.



Frequência (Hz): 389.364919

Magnitude (dB): 6.440000

Diferença da magnitude (dB): 5.688370

Banda (Hz): 78.057861

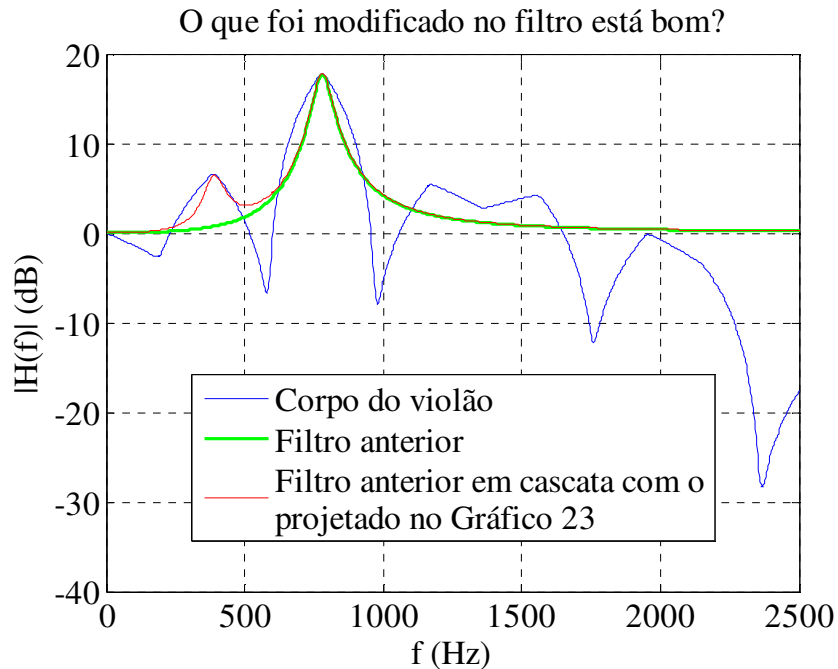


Gráfico 24 - Módulo da resposta em frequência do corpo do violão, do filtro anterior e do filtro anterior em cascata com o projetado no Gráfico 23.

Ponto aceitável [s]? s

Com ganho de 5.688370 dB [s]? s

Com banda de 78.057861 Hz [s]? s

Esses pares de pólos/zeros são bons [s]? s

Integrando esses pares de pólos e zeros no filtro.

Adicionar outro ponto [s]? n

Agora o filtro está projetado. Por fim, são mostradas mais duas janelas de gráficos: uma com o módulo da resposta em frequência do corpo do violão $H_{cv}(f)$ e do filtro criado $H_{f1}(f)$, mostradas no Gráfico 25, e a outra com o diagrama dos pólos e zeros no plano z , mostrada no Gráfico 26.

numo =

1.03158683947960	-1.97818003276375	0.95887144752201
1.00511493795769	-1.98588034273423	0.98382508831280

deno =

```
1.0000000000000000 -1.97818003276375 0.99045828700161
1.0000000000000000 -1.98588034273423 0.98894002627049
```

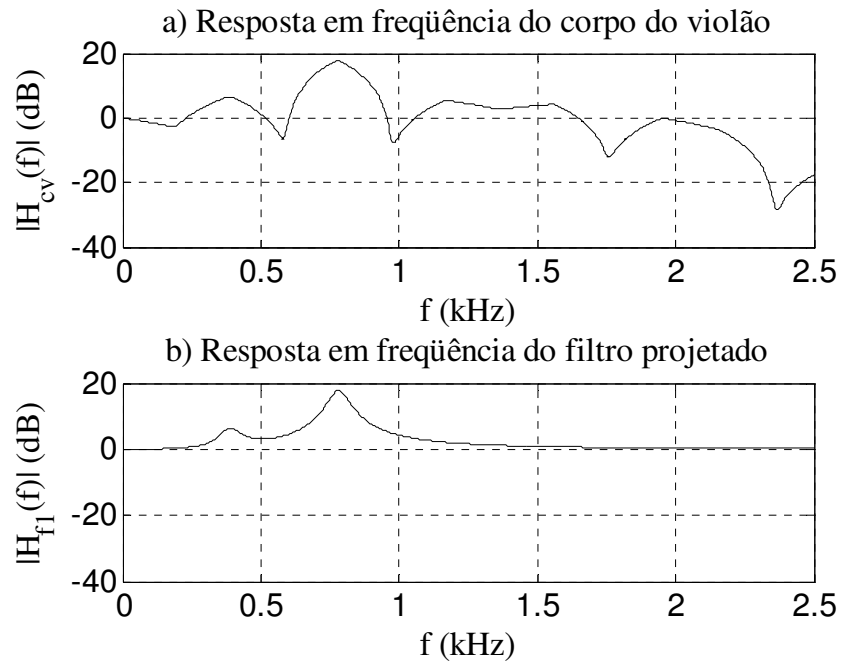


Gráfico 25 - Módulo da resposta em frequência: a) do corpo do violão; b) do filtro projetado.

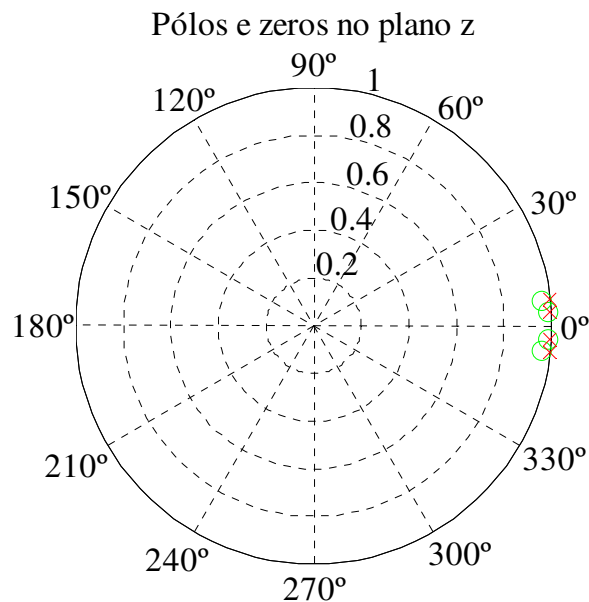


Gráfico 26 - Localização dos pólos e zeros do filtro no círculo unitário do plano z.

Os valores dos pólos e zeros do filtro são: $p_{1,2} = 0.9891 \pm j0.1103$,

$$p_{3,4} = 0.9929 \pm j0.0549, \quad z_{1,2} = 0.9588 \pm j0.1010 \quad \text{e} \quad z_{3,4} = 0.9879 \pm j0.0538.$$

Pelas variáveis *numo* e *deno*, verifica-se que o filtro tem a seguinte função de transferência:

$$H_{f1}(z) = \frac{(1.0316 - 1.9782z^{-1} + 0.9589z^{-2})(1.0051 - 1.9859z^{-1} + 0.9838z^{-2})}{(1 - 1.9782z^{-1} + 0.9905z^{-2})(1 - 1.9859z^{-1} + 0.9889z^{-2})} \quad (37)$$

4.2.2 Comparação entre o filtro e o corpo do violão

É interessante comparar a resposta ao impulso do filtro que simula o corpo do violão com a do próprio corpo do violão. Para fazer isso foi usada a função `makedimpulse`, fornecida em (STONICK; BRADLEY, 1996) e modificada depois para melhor apresentação neste trabalho. Esta função é listada no item 2 do Apêndice A. As respostas ao impulso $h_{f1}[n]$ e $h_{cv}[n]$ correspondentes, respectivamente, ao filtro e ao corpo do violão são mostradas no Gráfico 27.

Ambas as respostas ao impulso normalizadas são bem parecidas, pois a frequência da senóide amortecida é próxima uma da outra. A diferença entre as duas respostas é maior somente no começo, mas depois diminuiu bastante, como pode ser observado no Gráfico 27(c). O erro médio quadrático é 1.93×10^{-4} .

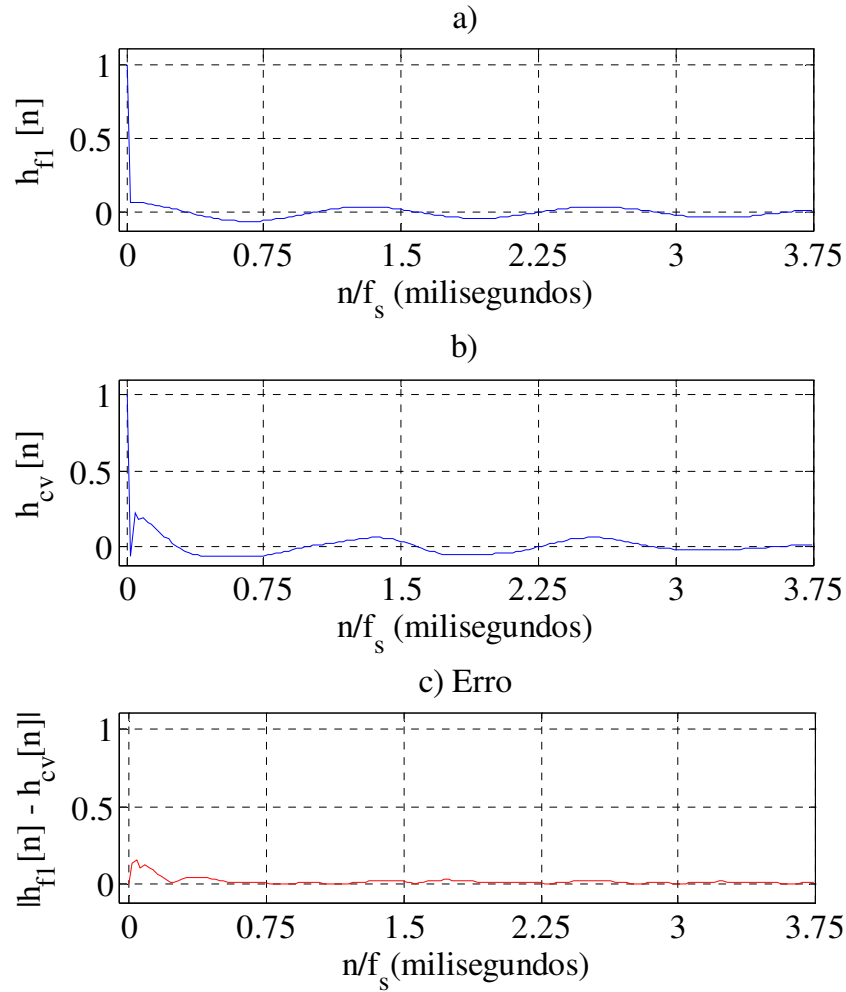


Gráfico 27 - Resposta ao impulso normalizada: a) do filtro criado; b) do corpo do violão. c) Erro.

4.2.3 Excitação do filtro: sinal produzido pelo violão

Para criar a excitação de um filtro, é necessário haver um sinal adequado para poder simular o instrumento. Pode-se criar a excitação através de um sinal que se deseja ter na saída do filtro. Nesta seção isso é feito seguindo a referência (STONICK; BRADLEY, 1996), em que o sinal que se deseja obter é a nota musical Sol (196Hz) gerada por um violão e notada por $y_{vsol}(t)$. Este sinal foi fornecido por (STONICK; BRADLEY, 1996).

Conforme mostrado no Diagrama 4, o sinal $y_{vsol}(t)$ é filtrado pelo inverso do filtro, gerando a excitação $s_{exc}(t)$, que é a base de dados para depois reproduzir o sinal da

nota Sol. A excitação $s_{exc}(t)$ é então filtrada, conforme mostrado no Diagrama 5, para reproduzir o sinal $y_{vsol}(t)$. Pode-se observar pelo Diagrama 4 e 5 que o sinal de entrada do Diagrama 4 é igual ao de saída do Diagrama 5, pois os dois filtros em cascata resultam em um sinal de saída igual ao da entrada.

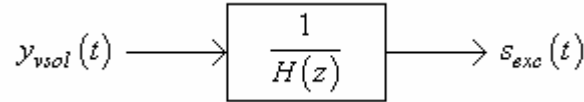


Diagrama 4 - Sistema para gerar a excitação $s_{exc}(t)$.

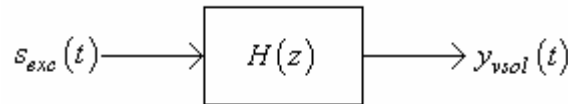


Diagrama 5 - Sistema para obter $y_{vsol}(t)$.

As funções `filtroinverso` e `cascata` modificadas de (STONICK; BRADLEY, 1996) e listadas no item 3 e 4 do Apêndice A, respectivamente, são usadas para criar essa excitação. As funções originais são `invert` e `cascade`, respectivamente.

A função `filtroinverso` é usada para inverter o filtro, verificando se o sistema tem pólos e zeros dentro do círculo unitário, ou seja, se é estável. Se for instável, ela calcula a nova função de transferência.

A função `cascata` é usada para gerar a saída dependendo do sinal de entrada e do filtro de 2ª ordem em cascata criado pela função `criafiltro`.

A função de transferência calculada por `filtroinverso` é a seguinte:

$$\frac{1}{H_{f1}(z)} = \frac{(1 - 1.9782z^{-1} + 0.9905z^{-2})(1 - 1.9859z^{-1} + 0.9889z^{-2})}{(1 - 1.9176z^{-1} + 0.9295z^{-2})(1 - 1.9758z^{-1} + 0.9788z^{-2})} \quad (38)$$

No Gráfico 28 são mostrados os sinais $y_{vsol}(t)$ e $s_{exc}(t)$.

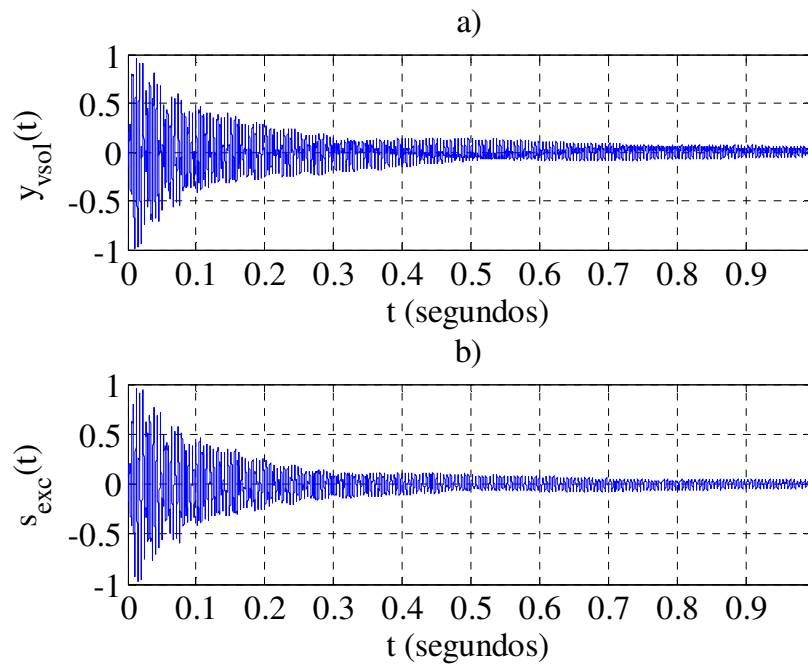


Gráfico 28 - a) Nota Sol gerada pelo violão; b) Entrada necessária para gerar na saída do filtro a nota Sol do violão.

Há poucas diferenças entre a nota Sol $y_{sol}(t)$ gerada pelo violão e a excitação $s_{exc}(t)$ do filtro. Ouvindo os dois sons também se percebe uma pequena diferença. Na excitação percebe-se que algumas frequências estão mais fracas. Esta solução para a excitação exige uma grande capacidade de armazenamento, pois é necessário gravar todas as notas possíveis geradas pelo violão. Pode-se, no entanto, também gerar outras notas utilizando a mesma excitação e um filtro diferente, porém isto não é feito aqui.

4.2.4 Excitação do filtro: dente de serra

Outro modo de simular um instrumento musical é utilizando outra excitação. Como o corpo do violão está sendo modelado pelo filtro, espera-se que com uma entrada adequada, a saída do filtro seja parecida com o som do violão. A corda ao ser acionada

produz uma frequência fundamental e outras harmônicas, que são frequências múltiplas da fundamental. Um sinal dente de serra com frequência igual a fundamental da nota pode ser usado como excitação do filtro e a amplitude da sua n -ésima harmônica no espectro é proporcional a $1/n$, como visto na Seção 2.2. Para criá-lo foi elaborada uma função que depende de parâmetros, como a frequência de amostragem, frequência do sinal e o número de pontos desejados ou o tempo. Essa função se chama `denteserra` e é listada no item 5 do Apêndice A. Vale ressaltar que a função dente de serra aqui usada é diferente da usada por (BRADLEY; CHENG; STONICK, 1995) e a sua forma temporal é apresentada no Gráfico 29(a).

A forma temporal da nota artificial, denominada $y_1(t)$ e apresentada no Gráfico 29(b), foi criada passando um dente de serra de 196 Hz $f_{serra1}(t)$ pelo filtro $H_{f1}[z]$.

No Gráfico 30 são mostrados os espectros normalizados $H_{y1}(f)$ e $H_{vsol}(f)$ correspondentes, respectivamente, aos sinais $y_1(t)$ e $y_{vsol}(t)$. Os espectros foram gerados pela função `especfreq` listada no item 6 do Apêndice A e a partir daqui são apresentados normalizados. Comparando os espectros de $y_1(t)$ e $f_{serra1}(t)$ do Gráfico 30, nota-se que a maior intensidade está na 4ª harmônica. Além disso, o sinal filtrado não tem um espectro em frequência parecido com o da nota Sol do violão e a nota gerada não é mais a mesma que a produzida pelo violão.

Pela resposta em frequência do corpo do violão mostrada no Gráfico 25, percebe-se que por a faixa de frequências de maior ganho estar em torno de 784 Hz, faz com que a harmônica de maior amplitude do dente de serra filtrado $y_1(t)$ seja a 4ª. Por causa disso foi projetado outro filtro que transforma um dente de serra em um sinal com espectro em frequência mais parecido com o da nota Sol gerada pelo violão.

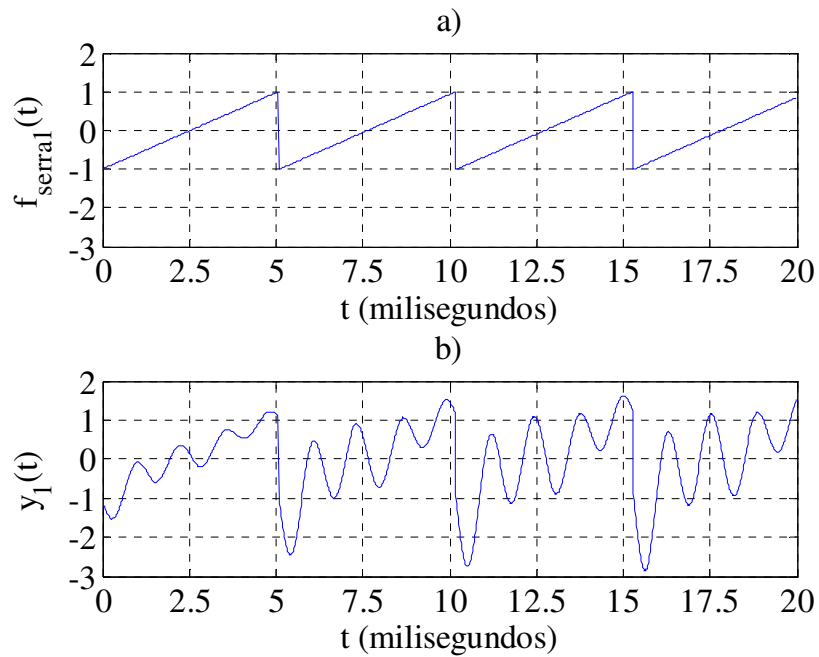


Gráfico 29 - Sinais em função do tempo: a) Dente de serra de 196 Hz $f_{serral}(t)$; b) Resultado $y_1(t)$ da filtragem do dente de serra.

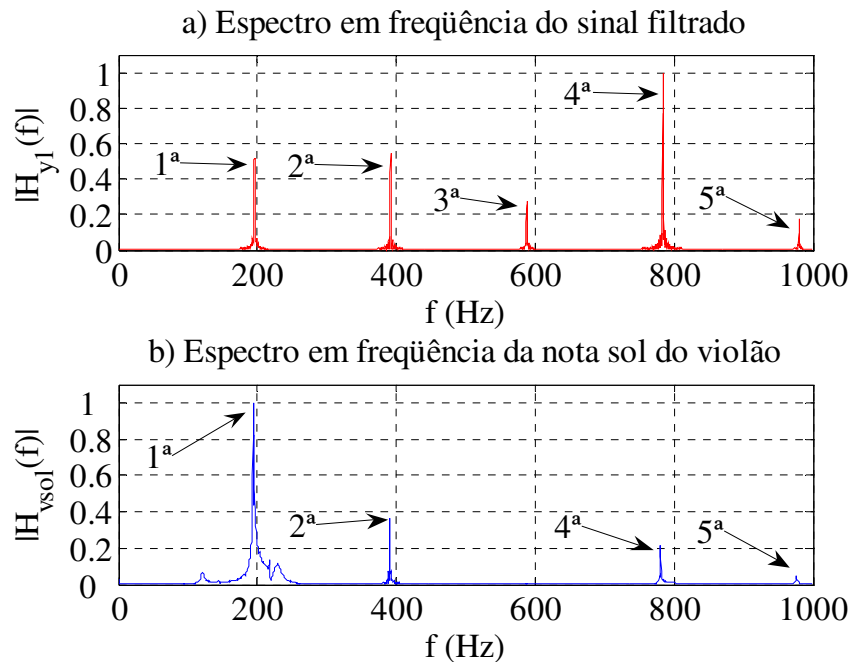


Gráfico 30 - Espectro em frequência normalizado: a) do sinal dente de serra filtrado $y_1(t)$; b) da nota Sol do violão. São indicadas as posições das harmônicas nos espectros.

4.2.5 Projeto modificado do corpo do violão

Para criar um sinal com espectro mais parecido com o som original foram testados vários filtros, porém o que apresentou melhores resultados foi um com amplitude menor nos dois picos ressonantes que existiam em $H_{f1}(f)$.

No Gráfico 31 é apresentado o módulo das respostas em frequência do novo filtro projetado $H_{f2}(f)$ e do corpo do violão.

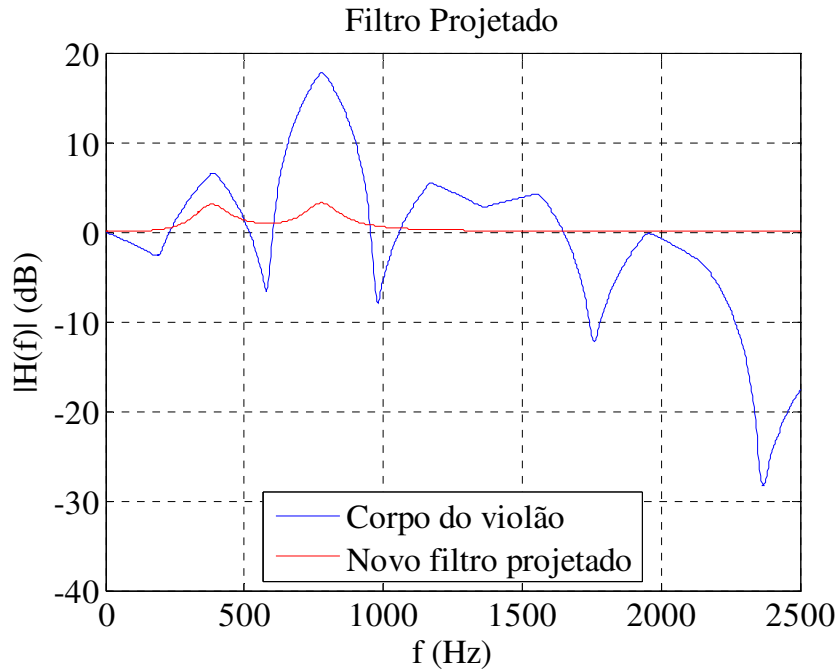


Gráfico 31 - Módulo da resposta em frequência do corpo do violão e do novo filtro projetado

$$H_{f2}(f).$$

Os dados deste filtro são: $f_0 = 383.06$ Hz, $G_0 = 3$ dB e $B_0 = 140$ Hz para o primeiro pico ressonante e $f_1 = 780$ Hz, $G_1 = 3$ dB e $B_1 = 140$ Hz para o segundo pico ressonante. A função de transferência desse filtro é:

$$H_{f2}(z) = \frac{(1.0041 - 1.9680z^{-1} + 0.9762z^{-2})(1.0041 - 1.9773z^{-1} + 0.9762z^{-2})}{(1 - 1.9680z^{-1} + 0.9802z^{-2})(1 - 1.9773z^{-1} + 0.9802z^{-2})} \quad (39)$$

As respostas ao impulso normalizadas mostradas no Gráfico 32 apresentam somente a parte oscilatória, pois a resposta ao impulso do filtro tem um pico inicial de amplitude muito maior que o restante da resposta. O erro médio quadrático é 2.52×10^{-4} , sendo um pouco maior que o da resposta impulsiva do primeiro filtro projetado.

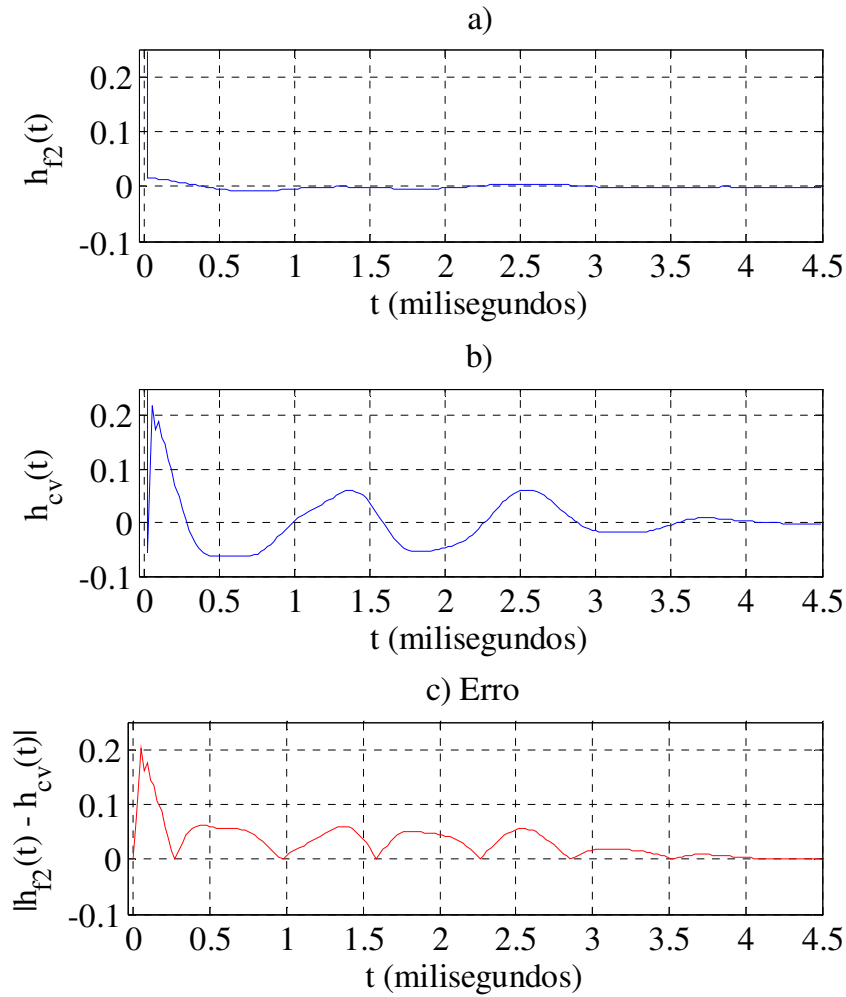


Gráfico 32 - Resposta ao impulso normalizada: a) do filtro $H_{f2}(z)$; b) do corpo do violão. c) Erro.

A resposta ao impulso de $H_{f2}(z)$ é parecida com a do corpo em termos de frequência, mas em relação a amplitude é bem diferente. No Gráfico 32(c) é mostrado que o erro em amplitude é praticamente igual a resposta impulsiva do corpo do violão.

O espectro em frequência $H_{y2}(f)$ do sinal dente de serra filtrado por

$H_{f2}(f)$ é mostrado no Gráfico 33.

Agora o espectro em frequência do dente de serra filtrado está um pouco mais parecido com o da nota gerada pelo violão, pois ambos têm a 1ª harmônica com maior amplitude. Já as outras harmônicas têm magnitudes diferentes.

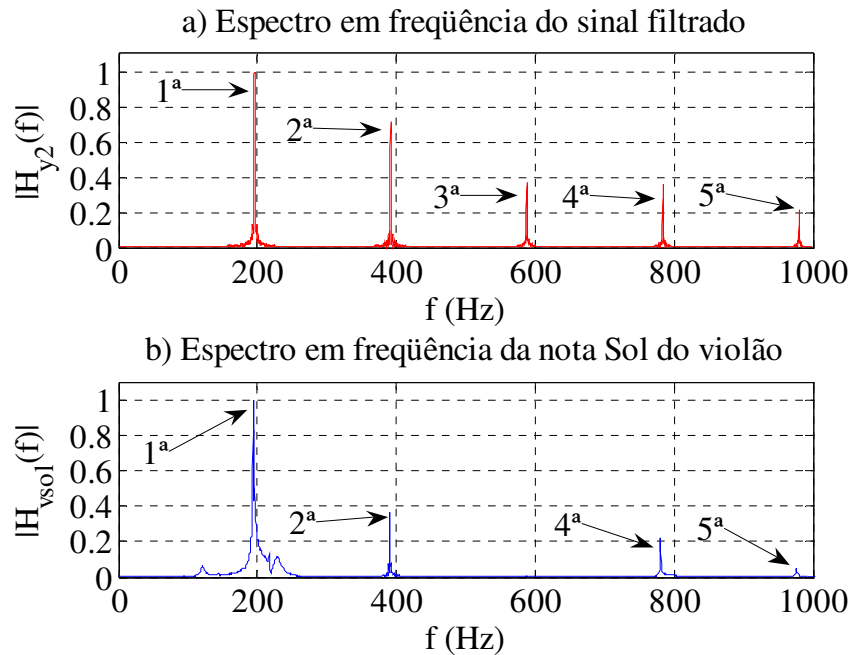


Gráfico 33 - Espectro em frequência normalizado: a) do dente de serra filtrado pelo novo filtro; b) da nota Sol gerada pelo violão. São indicadas as posições das harmônicas nos espectros.

4.2.6 Excitação da corda

A função `gerar`, modificada de (STONICK; BRADLEY, 1996), ajuda a modelar a excitação da corda do violão $s_{corda}(t)$ através de um filtro diferente do que é projetado pela função `criafiltro`. Ela usa determinados parâmetros da corda do violão para gerar a excitação através de um filtro, o que não é detalhado aqui, mas consta no item 7 do Apêndice A.

A excitação será usada para fazer decair com o tempo o sinal $y_2(t)$, já que a

excitação da corda cai com o tempo. Primeiro a excitação da corda é filtrada, gerando $s_{corda}(t)$, depois o dente de serra $f_{serra1}(t)$ é filtrado também. No final os dois são multiplicados, gerando $y_3(t)$. Utilizou-se esta solução porque apresentou melhores resultados experimentalmente. No Gráfico 34 é mostrada a forma temporal dos sinais $s_{corda}(t)$, $y_3(t)$ e $y_{vsol}(t)$.

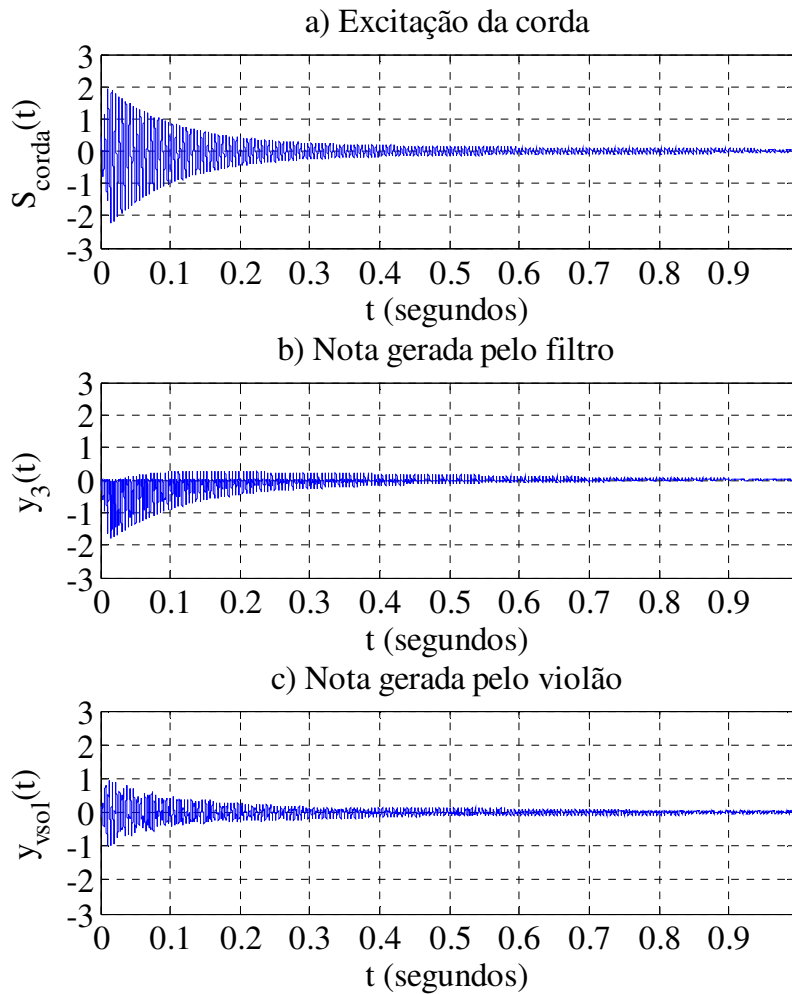


Gráfico 34 - Sinal: a) excitação da corda; b) nota Sol artificial; c) nota Sol gerada pelo violão.

A nota gerada pelo filtro parece com a nota Sol do violão, com a diferença que ela parece mais artificial. A nota artificial $y_3(t)$ e a nota Sol gerada pelo violão $y_{vsol}(t)$ têm o espectro em frequência mostrado no Gráfico 35 e são notados como $H_{y_3}(f)$ e $H_{vsol}(f)$,

respectivamente.

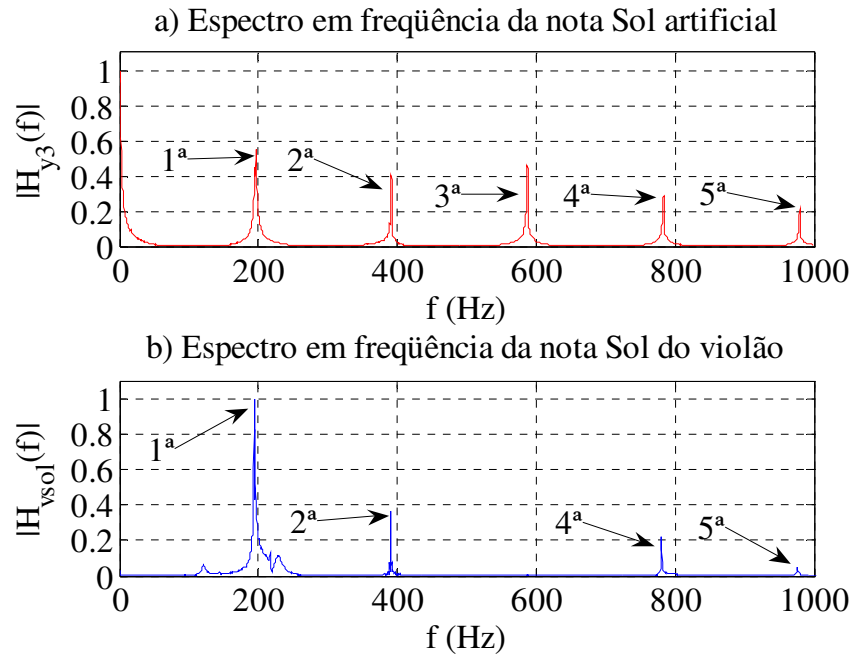


Gráfico 35 - Espectro em frequência normalizado: a) $H_{y3}(f)$; b) $H_{vsol}(f)$. São indicadas as posições das harmônicas nos espectros.

Utilizando a excitação da corda para criar o decaimento que existe ao acionar uma corda, verificou-se que apareceu no espectro em frequência uma componente de sinal contínuo com a maior magnitude, porém de todas as harmônicas, a 1ª harmônica é a maior em magnitude em ambos os sinais.

Se normalizarmos com relação a 1ª harmônica, as diferenças entre a 2ª e 5ª harmônica da nota gerada pelo violão e pelo filtro são devidas à resposta em frequência do corpo do violão, que amplifica mais em torno da 4ª harmônica e da 2ª harmônica. Pelo espectro da nota Sol do violão percebe-se que a maior ganho deveria ser em torno da 1ª harmônica, o que não acontece na resposta em frequência do corpo do violão.

4.2.7 Mudança de tom

As notas musicais são representadas pelas sete primeiras letras do alfabeto (A , B , C , D , E , F e G), sendo conhecidas como alfabeto musical. Os seus nomes são, respectivamente, Lá, Si, Dó, Ré, Mi, Fá e Sol. A oitava nota, conhecida como oitava, tem o mesmo nome que a primeira, mas está um tom acima.

Para representar um semitom acima ou abaixo de uma nota adiciona-se um símbolo após a nota. O sustenido (\sharp) indica um semitom acima da nota e o bemol (\flat) indica um semitom abaixo da nota. Por exemplo, A^\flat tem o mesmo som que o G^\sharp , mas está meio tom acima da nota Sol. Existem doze semitons: A , A^\sharp , B , C , C^\sharp , D , D^\sharp , E , F , F^\sharp , G , G^\sharp (WYATT; SCHROEDER, 1998). Também é colocado um número subscrito para indicar em que oitava está a nota. Por exemplo, a nota G_2 está uma oitava acima da nota G_1 , ou seja, G_2 é mais agudo que G_1 .

Os doze semitons de cada oitava são separados logaritmicamente na frequência. A frequência de uma nota é $2^{1/12}$ vezes a frequência da nota anterior. Subir uma oitava significa dobrar a frequência. O Quadro 1 relaciona as notas com a sua frequência (STONICK; BRADLEY, 1996).

Pode ser realizada uma mudança de tom usando o processo de dizimação e interpolação. Pelo Quadro 1 verifica-se que D_4 tem frequência igual a 293.66 Hz. Para mudar da nota Sol G_3 para D_4 é necessário multiplicar a frequência da nota Sol por um fator de 1.5 . Interpolando por 2 e realizando a dizimação num fator de 3 , será obtida a nota Ré $s_{vre}(t)$. É utilizada a nota Sol gerada pelo próprio violão para realizar a mudança de tom via dizimação e interpolação.

Quadro 1 - As notas musicais e suas respectivas frequências.

Nota	Frequência (Hz)
C_0	$13.75 * 2^{3/12}$
\vdots	\vdots
B_1	$27.5 * 2^{2/12}$
C_2	$27.5 * 2^{3/12}$
\vdots	\vdots
G_2	$55 * 2^{10/12}$
\vdots	\vdots
G_3	$110 * 2^{10/12}$
$G_3^\# / A_3^b$	$110 * 2^{11/12}$
A_3	220
$A_3^\# / B_3^b$	$220 * 2^{1/12}$
B_3	$220 * 2^{2/12}$
C_4	$220 * 2^{3/12}$
$C_4^\# / D_4^b$	$220 * 2^{4/12}$
D_4	$220 * 2^{5/12}$
$D_4^\# / E_4^b$	$220 * 2^{6/12}$
E_4	$220 * 2^{7/12}$
F_4	$220 * 2^{8/12}$
$F_4^\# / G_4^b$	$220 * 2^{9/12}$
G_4	$220 * 2^{10/12}$
\vdots	\vdots
$D_8^\# / E_8^b$	$3520 * 2^{6/12}$

No Gráfico 36 é mostrado o sinal $s_{re}(t)$ e o seu respectivo espectro em frequência normalizado $H_{re}(f)$.

A duração de $s_{re}(t)$ foi menor que $s_{vsol}(t)$ porque foi utilizado um fator maior para a dizimação do que para a interpolação. Ouvindo $s_{re}(t)$ percebe-se um som mais agudo

e que perde qualidade por causa desse processo de mudança de tom. Pelo espectro em frequência desse sinal nota-se que o sinal tem frequência fundamental aproximadamente igual a da nota Ré.

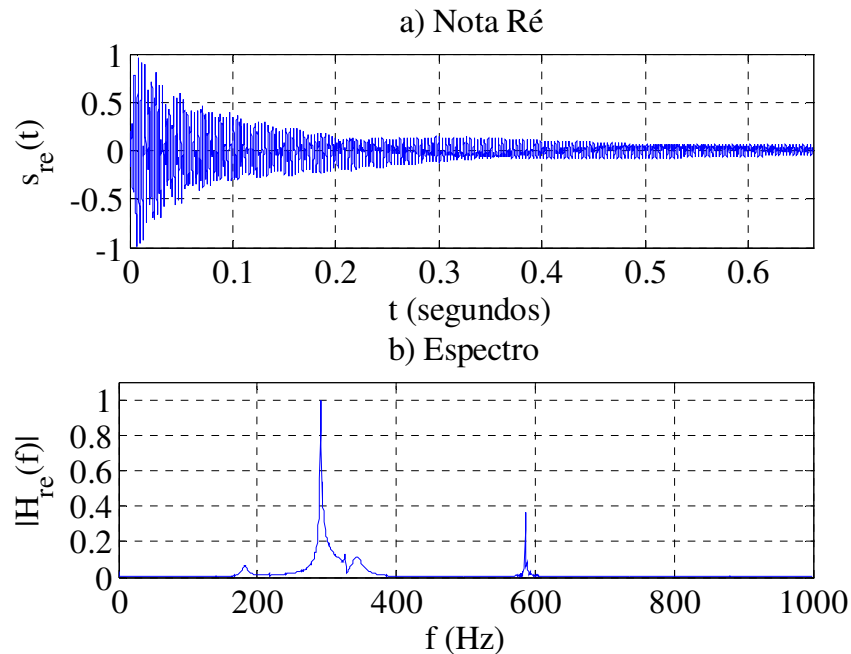


Gráfico 36 - a) Nota Ré gerada pelo processo de interpolação e dizimação da nota Sol do violão e b) seu respectivo espectro em frequência normalizado.

4.3 SIMULAÇÃO DE UM BAIXO

Nesta seção é realizada uma simulação de um baixo, em que as gravações foram obtidas pelo autor.

É feito aqui uma aproximação da resposta ao impulso obtida do baixo com a de um filtro. A seguir, foi realizada uma simulação para verificar o funcionamento do filtro como se fosse o corpo do baixo. Foi usado um baixo de 5 cordas da Washburn[®], modelo XB-125 e para amplificar o som foi utilizado o amplificador WarmMusic[®] 108rb. No processo de gravação, além de usar o baixo e o amplificador, usou-se o microfone SoundMax[®] Superbeam[®] ligado a placa de som Realtek[®] AC'97.

A resposta ao impulso é obtida registrando-se o som irradiado pelo amplificador ao aplicar-se um impulso mecânico nas cordas (KARJALAINEN; VÄLIMÄKI; JÁNOZY, 1993).

4.3.1 Espectro em frequência

Foram obtidas seis respostas ao impulso do baixo, pois cinco correspondem a cada corda separada e a sexta corresponde a todas acionadas ao mesmo tempo. A frequência de amostragem usada para capturar todos os sons originados do baixo foi de 44.1 kHz.

Os espectros em frequência das seis respostas ao impulso gravadas são mostradas no Gráfico 37. No Gráfico 38 é apresentado o módulo da resposta em frequência das cinco cordas acionadas ao mesmo tempo e também da somatória do módulo das respostas de cada uma das cordas acionadas separadamente. Além disso, é apresentado no Gráfico 38(b) a resposta impulsiva do baixo $h_{Baixo}(t)$ ao acionar as 5 cordas.

Cada uma das cordas possui uma resposta em frequência diferente. A resposta em frequência ao acionar as cinco cordas parece diferente da somatória das respostas de cada corda. Isso pode ter acontecido porque os sinais impulso de cada uma das cordas têm tempos de duração diferentes, obtendo espectros com número de pontos diferentes, como pode ser observado no Gráfico 37.

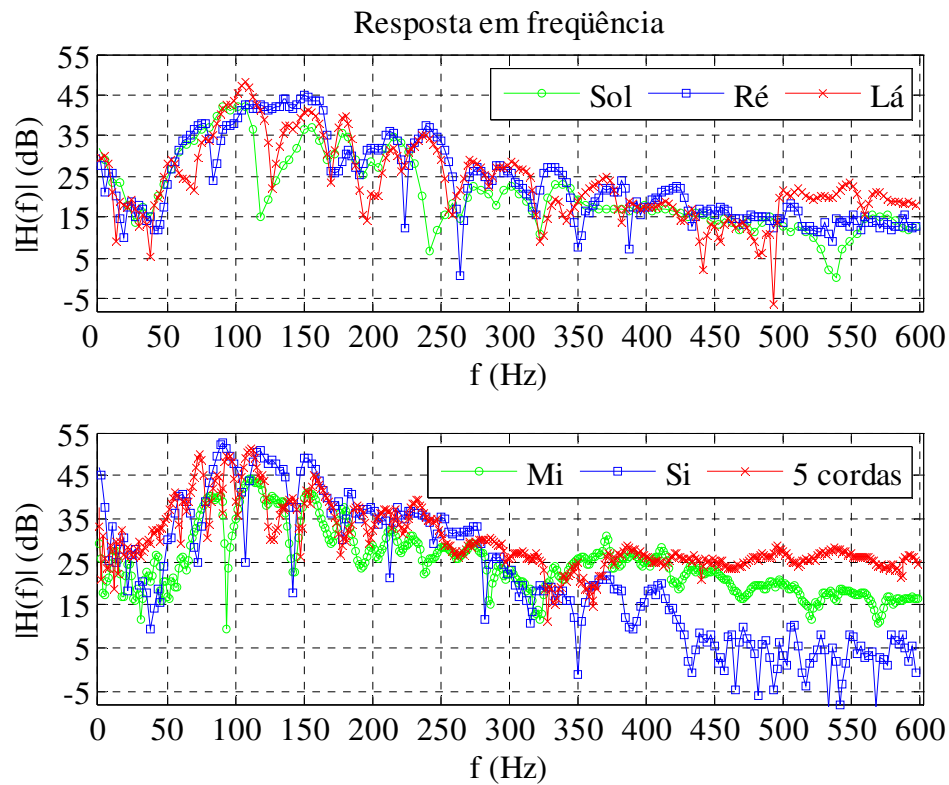


Gráfico 37 - Módulo da resposta em frequência das 5 cordas do baixo acionadas separadas (Sol, Ré, Lá, Mi e Si) e acionadas juntas (5 cordas).

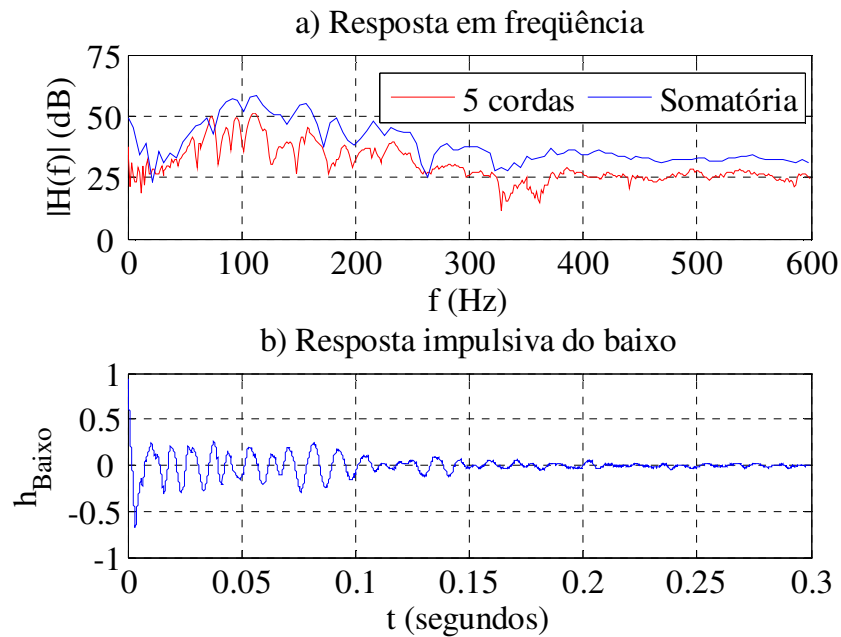


Gráfico 38 - a) Módulo da resposta em frequência do baixo e da somatória das respostas de cada corda; b) Resposta impulsiva do baixo ao acionar as 5 cordas.

4.3.2 Projeto do filtro

Foi projetado um filtro para gerar o som de todas as cordas e a sua resposta em frequência é apresentada no Gráfico 39, bem como a resposta em frequência do baixo ao acionar as 5 cordas.

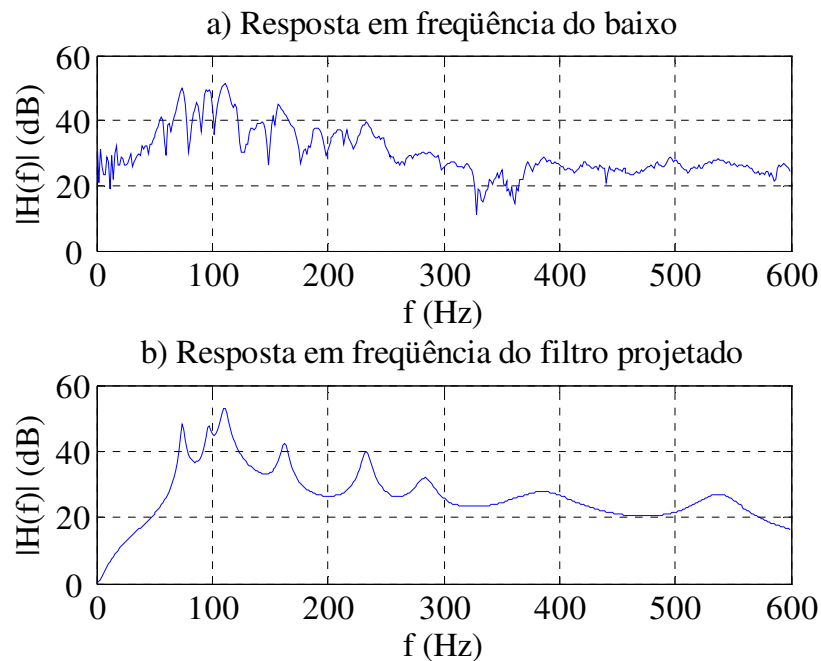


Gráfico 39 - Módulo da resposta em frequência: a) das 5 cordas do baixo; b) do filtro projetado.

Foram usados 8 filtros de 2º grau em cascata e como excitação usou-se o sinal dente de serra.

Inicialmente gerou-se um dente de serra com frequência igual a da nota Si (B_1). O espectro do sinal filtrado é apresentado no Gráfico 40(a). Além disso, é mostrado no Gráfico 40(b) o módulo do espectro em frequência da nota Si gerada pelo baixo. O sinal filtrado correspondente a nota Si tem espectro em frequência $H_1(f)$ e o da nota Si gerada pelo baixo é $H_{si}(f)$.

Pelo Gráfico 40 verifica-se que os dois sinais são parecidos no espectro,

apresentando amplitudes semelhantes na faixa entre a 2ª e 5ª harmônica.

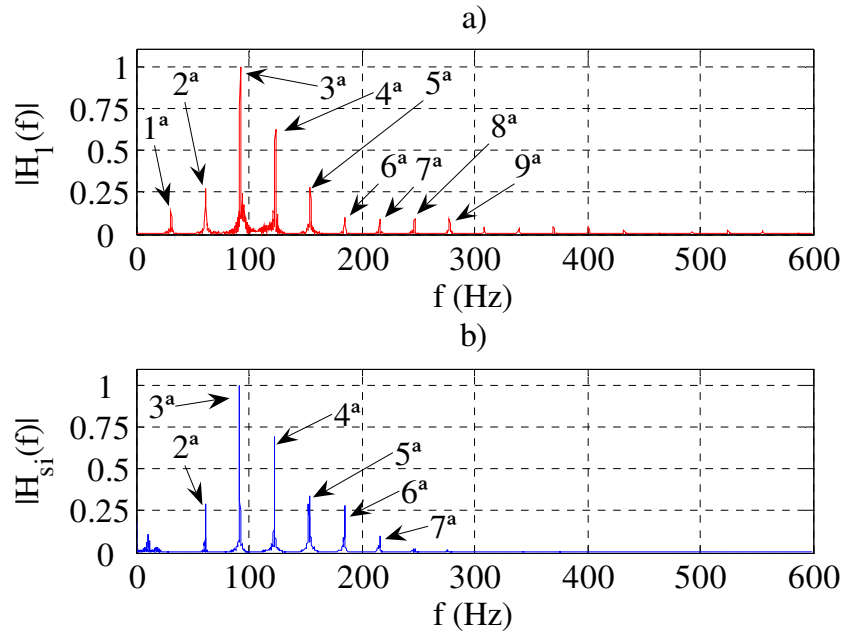


Gráfico 40 - Espectro em frequência da nota Si produzida: a) pelo filtro; b) pelo baixo. São indicadas as posições das harmônicas nos espectros.

Foi gerado também um dente de serra correspondente a nota Sol (G_2), para verificar se o filtro é bom para gerar todas as notas. O módulo dos espectros em frequência da nota Sol gerada pelo filtro e pelo baixo, notados por $|H_2(f)|$ e $|H_{sol}(f)|$, respectivamente, são mostrados no Gráfico 41. A nota Sol usada aqui corresponde a corda mais aguda do baixo usado.

Analisando o Gráfico 41 constata-se que o filtro não gerou um sinal com espectro em frequência parecido com o do sinal gerado pelo baixo. Esta diferença pode ter acontecido por não ter sido considerada a fase. No entanto, a nota produzida pelo filtro continua Sol (G_2).

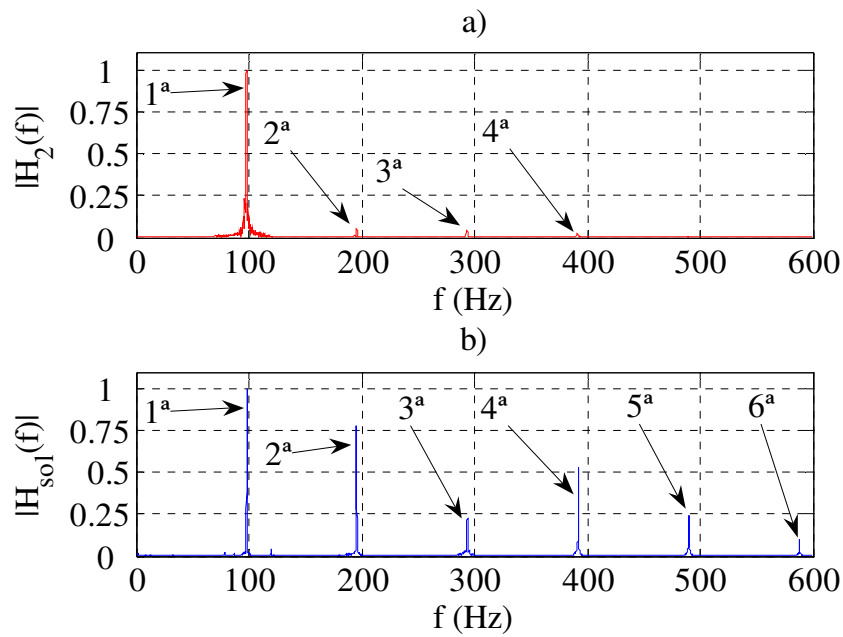


Gráfico 41 - Espectro em frequência da nota Sol produzida: a) pelo filtro; b) pelo baixo. São indicadas as posições das harmônicas nos espectros.

Com base nisso foi feito o projeto de um filtro que corresponda ao corpo do baixo ao acionar a corda Si.

4.3.3 Projeto do filtro para a corda Si

Após obter-se a resposta ao impulso $h_{si}(t)$ da corda Si, foi modelado um filtro $H_{Fsi}(z)$ que seleciona as faixas de frequência mais relevantes da nota Si que a corda Si produz. Para isso foram necessários sete filtros de 2ª ordem em cascata. Os dados do filtro são: $f_1=91.7627$ Hz, $G_1 = 45$ dB, $B_1 = 8.0750$ Hz; $f_2=61.6936$ Hz, $G_2 = 39.4737$ dB, $B_2 = 8.0750$ Hz; $f_3=37.8456$ Hz, $G_3 = 8.9398$ dB, $B_3 = 7$ Hz; $f_4=125.9793$ Hz, $G_4 = 47.6608$ dB, $B_4 = 13.4583$ Hz; $f_5=156.0484$ Hz, $G_5 = 46.4912$ dB, $B_5 = 10.7666$ Hz; $f_6=197.5230$ Hz, $G_6 = 36.5497$ dB, $B_6 = 16.1499$ Hz; $f_7=142.5691$ Hz, $G_7 = 17.8363$ dB,

$B_7 = 5.3833$ Hz. São mostrados o módulo das respostas em frequência $H_{si}(f)$ do corpo do baixo ao acionar a corda Si e $H_{Fsi}(f)$ do filtro projetado no Gráfico 42.

As respostas ao impulso de $H_{Fsi}(f)$ e $H_{si}(f)$ são mostradas no Gráfico 43.

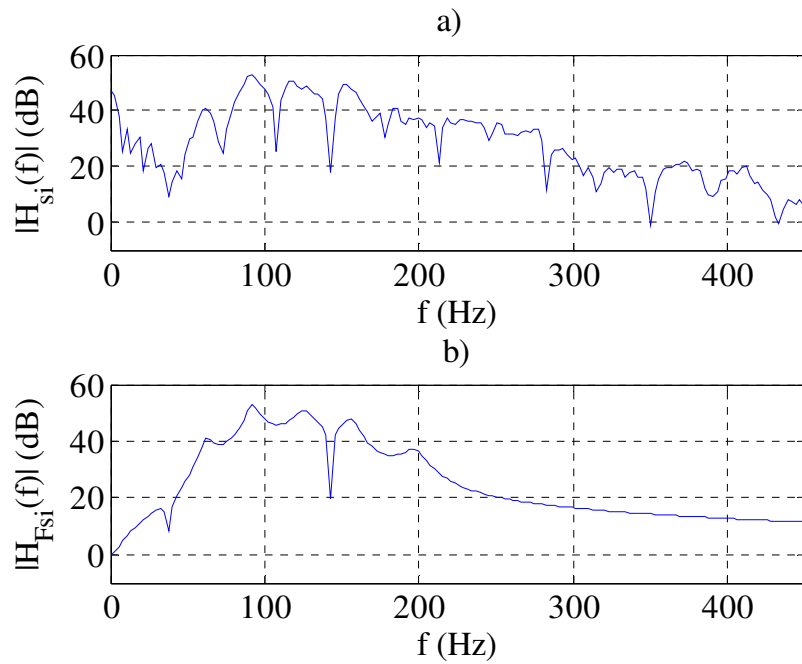


Gráfico 42 - Módulo da resposta em frequência: a) do corpo do baixo ao acionar a corda Si; b) do filtro projetado.

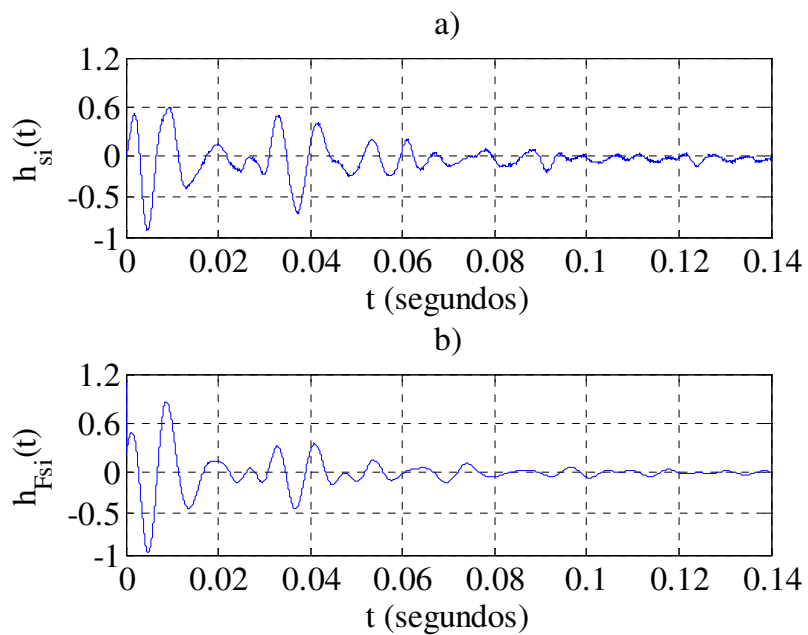


Gráfico 43 - Resposta ao impulso: a) $h_{si}(t)$ da corda Si do baixo; b) $h_{Fsi}(t)$ do filtro $H_{Fsi}(f)$.

As respostas ao impulso $h_{si}(t)$ e $h_{Fsi}(t)$ são bem semelhantes na frequência, contendo 5 picos nos primeiros 0.04 s. Já a partir desse tempo há uma diferença visível nas formas da onda.

4.3.4 Dente de serra como excitação do filtro

O comportamento do filtro foi também analisado com relação a uma excitação dente de serra, com frequência fundamental de 30.87 Hz, que corresponde a nota Si (B_1). Essa nota gerada pelo baixo e a resposta do filtro ao sinal dente de serra de 30.87 Hz são mostradas no Gráfico 44 e foram notadas como $y_{si}(t)$ e $y_{Fsi}(t)$, respectivamente.

O sinal gerado pelo filtro tem uma amplitude muito elevada se comparada com a nota Si do baixo. Isso é devido ao ganho do filtro e por isso é necessário normalizar as suas amplitudes antes de reproduzi-lo nas caixas de som. Além disso, faltam algumas frequências pelo fato de que entre 0s e 0.05s há oito picos em $y_{si}(t)$ e seis em $y_{Fsi}(t)$.

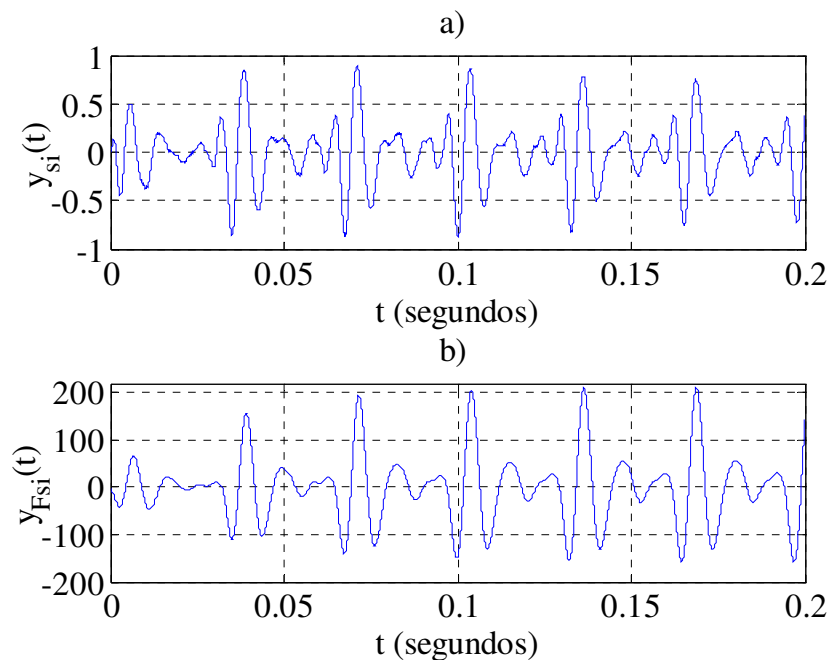


Gráfico 44 - a) Nota Si $y_{si}(t)$ gerada pelo baixo; b) Sinal filtrado $y_{Fsi}(t)$.

O módulo dos espectros em frequência normalizados $H_{si}(f)$ e $H_{Fsi}(f)$ de $y_{si}(t)$ e $y_{Fsi}(t)$, respectivamente, são mostrados no Gráfico 45.

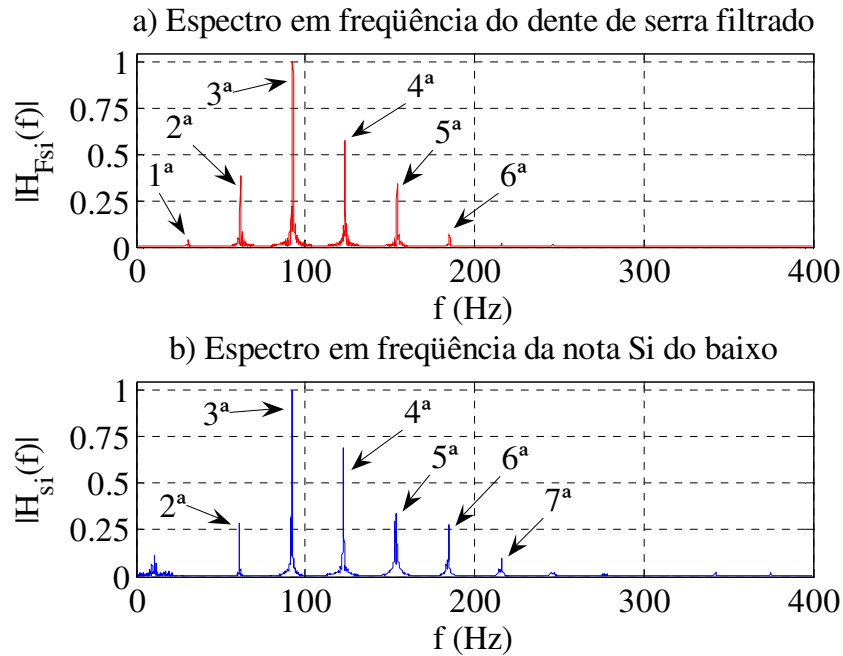


Gráfico 45 - Espectro em frequência: a) do dente de serra filtrado $y_{Fsi}(t)$; b) da nota Si do baixo

$y_{si}(t)$. São indicadas as posições das harmônicas nos espectros.

Pode-se ver que as duas maiores amplitudes de ambos os espectros normalizados estão em torno da mesma frequência. O corpo do baixo atenuou muito a frequência fundamental da nota Si, o que também aconteceu na filtragem do sinal dente de serra. A 2ª e 5ª harmônicas têm amplitudes semelhantes nos espectros de $y_{si}(t)$ e $y_{Fsi}(t)$, porém a 6ª harmônica tem uma diferença maior. A partir da 7ª harmônica as amplitudes são praticamente desprezíveis nos dois espectros.

Ouvindo os dois sinais, nota-se que eles são parecidos, porém existe uma diferença entre eles. Esta é causada pelo decaimento natural, ou seja, o sinal gerado pelo baixo diminui de intensidade com o tempo, enquanto que no sinal filtrado isso não ocorre.

5 CONCLUSÕES

Neste trabalho mostrou-se que os filtros digitais e o sinal de excitação dente de serra podem reproduzir convenientemente o som de um instrumento musical.

Nos Capítulos 2 e 3 foram abordados assuntos importantes para a realização das simulações, como os sistemas de tempo discreto, a Série de Fourier, a Transformada Z e os filtros digitais. No Capítulo 4 foram mostradas as simulações de dois instrumentos musicais: o violão e o baixo.

No caso do violão foram necessárias duas simulações. A primeira apresentou resultados aquém dos desejados, como discutido na Seção 4.2.4, página 50. O som gerado pelo dente de serra filtrado não apresentou um espectro em frequência parecido com o do som real.

Já na segunda simulação quando se aplicou um dente de serra e se considerou o decaimento natural da nota, o espectro do sinal de saída apresentou uma componente contínua com maior amplitude, seguida da frequência fundamental. No entanto, como mostrado no Gráfico 35, as outras harmônicas não se pareciam em ambos os sinais devido a resposta em frequência do corpo do violão, como discutido na Seção 4.2.6, página 55. A resposta do instrumento vista no Gráfico 31 mostra que a 4ª harmônica tinha o maior ganho, mas analisando pelo espectro da nota G_3 do violão verifica-se que essa frequência não tem a maior amplitude, como mostrado no Gráfico 35. Neste caso a nota gerada pelo filtro corresponde a desejada, porém ainda assim não foi possível gerar um sinal com espectro em frequência mais parecido. Somente a 1ª harmônica, por ter a maior amplitude se comparada às outras harmônicas, foi mais parecida com a 1ª harmônica da nota real.

O uso da dizimação e da interpolação para mudar da nota G_3 para D_4 sem a necessidade de armazenar as duas notas, mostrou-se funcional e acabou gerando a nota

correta. Além disso, o sinal modificado tem menos amostras e por causa disso há a necessidade de armazenar durante mais tempo as notas.

Os resultados da simulação do baixo, discutidos na Seção 4.3.2, página 63 e Seção 4.3.4, página 67, mostram que um filtro digital pode ser usado para reproduzir adequadamente os sons gerados por um baixo, porém seus coeficientes precisam ser ajustados para cada nota. Acusticamente os sinais obtidos na saída da cascata de filtros alimentados pelo dente de serra e o som real gerado pelo instrumento são parecidos. O dente de serra mostrou ser uma boa solução para a excitação do filtro. O decaimento natural no domínio do tempo que ocorre ao acionar uma corda não foi levado em conta, pois precisa de uma análise mais detalhada de como modelar esse efeito nas cordas do baixo. No entanto, é importante ressaltar que nesta simulação foram usados mais filtros em cascata do que foi usado nas duas simulações do violão.

Este trabalho buscou contribuir para a síntese de instrumentos musicais usando o Matlab como ferramenta para o projeto e simulação de filtros digitais.

Foi publicado um Trabalho de Iniciação Científica nos anais do XXV Simpósio Brasileiro de Telecomunicações (SBrT 2007), realizado entre 03-06 de Setembro de 2007. Este artigo e o pôster apresentado neste Simpósio estão colocados no Apêndice B.

Com este trabalho, foi possível entender melhor como funcionam os sistemas de tempo discreto e como a Transformada Z ajuda na análise desse sistema. Com isso pôde-se ver também como muda a resposta em frequência de um filtro ao modificar os pólos e zeros.

Como trabalhos futuros pode-se analisar o problema do decaimento natural no domínio do tempo do som gerado pelos instrumentos musicais. Além disso, a análise dos diferentes estilos de acionar as cordas do instrumento também é interessante, pois modifica a dinâmica da produção do som, alterando a resposta em frequência do instrumento. Alguns artigos interessantes abordam esses assuntos, como (BRADLEY; CHENG; STONICK, 1995;

KARJALAINEN; VÄLIMÄKI; JÁNOZY, 1993; SAYEGH; TENORIO, 1998; TRAUBE; SMITH III, 2001).

REFERÊNCIAS

- BELLANGER, M. G. *Digital processing of signals: theory and practice*. Tradução John C. C. Nelson. 3. ed. Chichester: John Wiley, 2000.
- BRADLEY, K.; CHENG, M.; STONICK, V. L. *Automated analysis and computationally efficient synthesis of acoustic guitar strings and body*. IEEE ASSP Workshop On Applications of Signal Processing to Audio and Acoustics, p. 238-241, 1995.
- DINIZ, P. S. R.; DA SILVA, E. A. B.; NETTO, S. L. *Digital signal processing: system analysis and design*. Tradução Luiz Wagner Pereira Biscainho. Porto Alegre: Bookman, 2004.
- GIROD, B.; RABENSTEIN, R.; STENGER, A. *Sinais e sistemas*. Tradução Bernardo Severo da Silva Filho. Porto Alegre: LTC, 2003.
- HAYKIN, S.; VEEN, B. V. *Sinais e sistemas*. Tradução José Carlos Barbosa dos Santos. Porto Alegre: Bookman, 1999.
- JACKSON, L. B. *Digital filters and signal processing*. 3. ed. Boston: Kluwer Academic, 1999.
- KARJALAINEN, M.; VÄLIMÄKI, V.; JÁNOZY, Z. *Toward high-quality sound synthesis of the guitar and string instruments*. Proceedings of the ICUC, p. 56-63, 1993.
- LATHI, B. P. *Signal Processing and Linear Systems*. New York: Oxford University Press, 1998.
- MITRA, S.K. *Digital signal processing: a computer based approach*. 2.ed. Boston: McGraw-Hill, 2001.
- OPPENHEIM, A. V.; SCHAFER, R. W.; BUCK, J. R. *Discrete-time signal processing*. 2. ed. New Jersey: Prentice Hall, 1999.
- PROAKIS, J. G.; MANOLAKIS, D. G. *Digital signal processing: Principles, algorithms and applications*. 3. ed. New Jersey: Prentice Hall, 1996.
- SAYEGH, S. I.; TENORIO, M. F. *Inverse viterbi algorithm as learning procedure and application to optimization in the string fingering problem*. IEEE International Conference of Neural Networks, v. 2, p. 491-498, 1998.
- STONICK, V. L.; BRADLEY, K. *Labs for signals and systems using Matlab*. Toronto: The PWS BookWare Companion Series, 1996.
- TRAUBE, C.; SMITH III, J. O. *Extracting the fingering and the plucking points on a guitar string from a recording*. IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics, p. 7-10, 2001.
- WYATT, K.; SCHROEDER, C. *Harmony & Theory: A comprehensive source for all musicians*. Milwaukee: Musicians Institute Press – Hal Leonard Corporation, 1998.

APÊNDICE A – FUNÇÕES DO MATLAB

1. Projeto do filtro – modificado da função manual de (STONICK; BRADLEY, 1996)

```
function [numo,deno] = criafiltro(impulso,num,den);
% [numo,deno] = criafiltro(impulso,num,den);
%
% Cria um filtro baseado na resposta ao impulso dada.
% num,den: filtro de 2ª ordem em cascata.
% Para criar um filtro novo substitua num e den por []
% Cada linha corresponde a um filtro de 2ª ordem, sendo que
% os coeficientes do filtro são armazenados da seguinte maneira:
%
% [ numa(0)   numa(1)   numa(2) ]
% [ numb(0)   numb(1)   numb(2) ]
% [ .....   .....   ..... ]
% [ numN(0)   numN(1)   numN(2) ]
%
% Essa configuração é usada pelas funções cascata e makedimpulse.
% ATENÇÃO:
% Esta função é baseada e modificada de
% 'Labs for signals and systems using Matlab' - V. STONICK & K.
BRADLEY

fs = input('Digite a frequência de amostragem: ');
clf;
hold off;
format long;
disp('Calculando a resposta em frequência do instrumento....');
namostras = length(impulso);

%% Torna o comprimento do impulso uma potência de 2
for k = 1:100,
    if 2^k > namostras,
        elevadoa = k;
        break;
    end
end
namostras = 2^elevadoa;
N = 2*namostras;
impulso = [impulso(:)' zeros(1,namostras - length(impulso))];

%% resposta em frequência
espectro = abs(fft(impulso,N));
espectrodb = 20*log10(espectro);
w = (0:N-1)*2*pi/N;
f = linspace(0,fs,length(w));

%% calcula filtro atual
disp('Calculando a resposta em frequência do filtro...')
[n,m] = size(num);
if length(num)==0 & length(den)==0
    h = ones(size(w));
```

```

    n = 0;
    % Ajusta para vetorial os coeficientes do filtro (h), se não
    estão dessa forma.
    % Coeficientes do filtro de 2º grau em cada linha.
else if n==3 & m ~= 3,
    num = num';
    den = den';
    n = m;
    h = ones(size(w));
end
end
% Se tiver filtros de 2ºgrau em cascata, é calculada a resposta em
% frequência do filtro total
if n>=1
    h=freqz(num(1,:),den(1,:),w,'whole');
end
for k = 2:n,
    h = h.*freqz(num(k,:),den(k,:),w,'whole');
end
% Gráficos do espectro da resposta impulsiva do instrumento e do
filtro
% atual
figure(1);
clf;
subplot(2,1,1)
plot(f,espectrodb);
xlabel('f (Hz)','fontname','Times New Roman','fontsize',12);
ylabel('|H(f)| (dB)','fontname','Times New Roman','fontsize',12);
title('Resposta em frequência','fontname','Times New
Roman','fontsize',12)
grid on
Ax = axis;
axis([0 max(f) Ax(3) Ax(4)])
subplot(2,1,2)
plot(f,20*log10(abs(h)),'k');
xlabel('f (Hz)','fontname','Times New Roman','fontsize',12);
ylabel('|H(f)| (dB)','fontname','Times New Roman','fontsize',12);
title('Filtro atual','fontname','Times New Roman','fontsize',12)
grid on
axis([0 max(f) Ax(3) Ax(4)])
pause
disp('Intervalo de frequência atual:');
disp(Ax(1:2));
Ax(2) = max(f); % é feito isto para não dar problema caso não
modificar a freq. max.
fmin = input('Frequência mínima (Hz) visualizada no gráfico ([] para
não modificar): ');
if ~isempty(fmin),
    Ax(1) = fmin;
else
    fmin = Ax(1);
end
fmax = input('Frequência máxima (Hz) visualizada no gráfico ([] para
não modificar): ');
if ~isempty(fmax),
    Ax(2) = fmax;

```

```

else
    fmax = Ax(2);
end
string = sprintf('Será usado o intervalo de frequência [%f,%f] Hz a
partir de agora.',Ax(1),Ax(2));
disp(string);
temp = find(f >= fmin);
ptfmin = temp(1); % (indice) frequência mínima mais próxima do que
se quer visualizar
temp = find(f >= fmax);
ptfmax = temp(1); % (indice) frequência máxima mais próxima do que
se quer visualizar
subplot(2,1,1)
plot(f(ptfmin:ptfmax),espectrodb(ptfmin:ptfmax));
xlabel('f (Hz)','fontname','Times New Roman','fontsize',12);
ylabel('|H(f)| (dB)','fontname','Times New Roman','fontsize',12);
title('Resposta em frequência','fontname','Times New
Roman','fontsize',12)
axis(Ax);
grid on
subplot(2,1,2)
plot(f(ptfmin:ptfmax),20*log10(abs(h(ptfmin:ptfmax))), 'k'); % filtro
atual
xlabel('f (Hz)','fontname','Times New Roman','fontsize',12);
ylabel('|H(f)| (dB)','fontname','Times New Roman','fontsize',12);
title('Filtro criado','fontname','Times New Roman','fontsize',12)
grid on
axis(Ax);
reseta = input('Limpar o conteúdo do numerador e do denominador
(H[z]=1) [n]? ','s');
if ~isempty(reseta) & reseta ~= 'n',
    disp('Numerador e denominador iguais a 1. H[z]=1')
    num = [];
    den = [];
    h = ones(size(f));
    subplot(2,1,2)
    plot(f(ptfmin:ptfmax),20*log10(abs(h(ptfmin:ptfmax))), 'k');
%filtro atual
    axis(Ax);
    xlabel('f (Hz)','fontname','Times New Roman','fontsize',12);
    ylabel('|H(f)| (dB)','fontname','Times New
Roman','fontsize',12);
    title('Filtro criado','fontname','Times New
Roman','fontsize',12)
    grid on
    pause
end
raizesnum = [];
raizesden = [];
for k = 1:n,
    raizesnum = [raizesnum roots(num(k,:))'];
    raizesden = [raizesden roots(den(k,:))'];
end
for i=1:1000
    if i>=2
        figure(1)
    end
end

```

```

        hold off
        subplot(2,1,2)
        plot(f(ptfmin:ptfmax),20*log10(abs(h(ptfmin:ptfmax))), 'g');
%filtro atual
        axis(Ax);
        xlabel('f (Hz)', 'fontname', 'Times New Roman', 'fontsize', 12);
        ylabel('|H(f)| (dB)', 'fontname', 'Times New
Roman', 'fontsize', 12);
        title('Filtro criado', 'fontname', 'Times New
Roman', 'fontsize', 12)
        grid on
        pause
        close(2)
    end
    figure(2)
    plot(f(ptfmin:ptfmax),20*log10(abs(h(ptfmin:ptfmax))), 'g');
%filtro atual
    hold on;
    plot(f(ptfmin:ptfmax),espectrodb(ptfmin:ptfmax)); % espectro do
impulso
    axis(Ax);
    xlabel('f (Hz)', 'fontname', 'Times New Roman', 'fontsize', 12);
    ylabel('|H(f)| (dB)', 'fontname', 'Times New
Roman', 'fontsize', 12);
    title('Clique no gráfico para criar o filtro
desejado', 'fontname', 'Times New Roman', 'fontsize', 12)
    grid on
    %pause()
    disp('Clique no gráfico para modificar o seu filtro atual.');
```

[x1,y] = ginput(1);

```

    stringfreq = sprintf('\nFrequência (Hz): %f',x1);
    disp(stringfreq);
    if(y > 0)
        gdb=y;
    else
        gdb = -100;
    end
    fprintf('Magnitude (dB): %f\n\n',gdb);
    pt = find(f >= x1);
    pt = pt(1); % ponto mais próximo do w clicado
    gthere = gdb - 20*log10(abs(h(pt))); % diferença entre esse
ponto e o filtro atual
    fprintf('Diferença da magnitude (dB): %f\n',gthere);
    if(gthere > 0)
        temp = find(espectrodb(pt:ptfmax) < gdb - 3); % 3dB de
diferença do máximo ou mínimo ganho para achar a banda
    else
        temp = find(espectrodb(pt:ptfmax) > gdb + 3);
    end
    bpt = temp(1) + pt; % índice de 'f' que corresponde a diferença
3dB
    bw = w(bpt) - w(pt); % banda entre o pico/entalhe e a diferença
de 3dB
    bwf = bw*fs/(2*pi);
    fprintf('Banda (Hz): %f\n\n',bwf); % Banda em Hz
    aceita = input('Ponto aceitável [s]? ', 's');
```

```

if isempty(aceita) | aceita == 's',
    string = sprintf('Com ganho de %f dB [s]? ',gthere);
    inp = input(string,'s');
    gdb = gthere;
    if ~isempty(inp)
        if inp(1) ~= 's'
            gdb = input('Ganho desejado (dB): ');
        end
    end
    essabanda = sprintf('Com banda de %f Hz [s]? ',bwf);
    inp2 = input(essabanda,'s');
    if ~isempty(inp2)
        if inp2(1) ~= 's'
            bwf = input('Banda (Hz): ');
            bw = 2*pi*bwf/fs;
        end
    end
    end
    % Cálculo do novo filtro
    x = 2*pi*x1/fs;
    novonum = zeros(1,3);
    novoden = zeros(1,3);
    keq = 10^(gdb/20);
    abw = (1-tan(bw/2))/(1+tan(bw/2));
    ggain = 0.5*(1+keq+abw-keq*abw);
    novoden(1) = 1;
    novoden(2) = -2*cos(x) * (1/(1+tan(bw/2)));
    novoden(3) = abw;
    novonum(1) = 1;
    novonum(2) = (-2*cos(x)*(1+abw))/(1+keq+abw-keq*abw);
    novonum(3) = (abw+keq*abw+1-keq)/(abw-keq*abw+1+keq);
    novonum=novonum*ggain;
    % Salva o antigo filtro, caso precise depois
    antigasraizesnum = raizesnum;
    antigasraizesden = raizesden;
    anticonum = num;
    antigoden = den;
    antigoh = h;
    % Junta o novo filtro
    raizesnum = [raizesnum roots(novonum)'];
    raizesden = [raizesden roots(novoden)'];
    num = [num ; novonum];
    den = [den ; novoden];
    h = h .* freqz(novonum,novoden,w);
    % Verifica se o filtro é o desejado
    figure(2)
    hold on;
    plot(f(ptfmin:ptfmax),20*log10(abs(h(ptfmin:ptfmax))), 'r');
    title('O que foi modificado no filtro está
bom?','fontname','Times New Roman','fontsize',12)
    axis(Ax);
    grid on
    bons = input('Esses pares de pólos/zeros são bons [s]?
','s');
    acao = ['Integrando esses pares de pólos e zeros no
filtro.'];
    % Descarta as últimas modificações no filtro, se desejado

```

```

        if ~isempty(bons)
            if bons(1) ~= 's',
                acao = ['Desfazendo os pares de pólos/zeros.'];
                raizesnum=antigasraizesnum;
                raizesden=antigasraizesden;
                num = antigonum;
                den = antigoden;
                h = antigoh;
            end
        end
        disp(acao)
    end
    adicionar = input('Adicionar outro ponto [s]? ','s');
    if ~isempty(adicionar)
        if adicionar(1) == 'n',
            break;
        end
    end
end

end
% Plota a resposta em freqüência do instrumento no 1° gráfico e
% a resposta em freqüência do filtro atual no 2° gráfico
pause
close(2)
subplot(2,1,2)
hold off
plot(f(ptfmin:ptfmax),20*log10(abs(h(ptfmin:ptfmax))), 'k');
title('Filtro projetado','fontname','Times New
Roman','fontsize',12);
xlabel('f (Hz)','fontname','Times New Roman','fontsize',12)
ylabel('|H(f)| (dB)','fontname','Times New Roman','fontsize',12)
axis(Ax);
grid on
pause
% Plano z
figure(2);
polar(angle(raizesnum),abs(raizesnum),'go');
hold on;
polar(angle(raizesden),abs(raizesden),'rx');
title('Pólos e zeros no plano z','fontname','Times New
Roman','fontsize',12);
grid on
% Coeficientes do(s) filtro(s) de 2ª ordem em cascata
numo = num;
deno = den;

```

2. Resposta ao impulso do filtro – modificado da função makedimpulse de (STONICK; BRADLEY, 1996)

```
function respimp = makedimpulse(num,den,amostras);
% saida = makedimpulse(num,den,amostras);
%
% Calcula a resposta ao impulso de sistemas de
% tempo discreto de segunda ordem em cascata.
% Usa dimpulse para achar a primeira resposta.
% Amostras é o número de amostras que se deseja
% ter na resposta calculada.
% ATENÇÃO:
% Esta função é baseada e modificada de
% 'Labs for signals and systems using Matlab' - V. STONICK & K.
BRADLEY

% n é o número de linhas
[n,m]=size(num);

temp = dimpulse(num(1,:),den(1,:),amostras);

if n ~= 1,
    respimp = cascata(num(2:n,:),den(2:n,:),temp);
    respimp = respimp';
else
    respimp = temp';
end
```

3. Inversão do filtro – modificado da função invert de (STONICK; BRADLEY, 1996)

```
function [novonum, novoden] = filtroinverso(num,den);

% [novonum,novoden] = filtroinverso(num,den);
%
% Gera os coeficientes do filtro inverso,
% modificando os pólos e zeros caso estes
% estejam fora do círculo unitário.
% ATENÇÃO:
% Esta função é baseada e modificada de
% 'Labs for signals and systems using Matlab' - V. STONICK & K.
BRADLEY

[n,m]=size(num);
novoden = zeros(size(den));
novonum = zeros(size(den));
% Verifica cada um dos filtros de 2ª ordem
for l=1:n,
    g = num(l,1);
    raizesnum = roots(num(l,:));
    raizesden = roots(den(l,:));
    modden = abs(raizesden);
    modnum = abs(raizesnum);
    % se o módulo do pólo ou do zero for maior que 1, é calculado um
    novo
    % módulo do pólo ou do zero, conforme visto neste Trabalho
```



```

    for k = 1:length(modden),
        if modden(k) > 1
            raizesden(k) =
1/abs(raizesden(k))*cos(angle(raizesden(k)))+j*1/abs(raizesden(k))*s
in(angle(raizesden(k)));
            disp('Invertendo o pólo.');
```

end

```

        if modnum(k) > 1
            raizesnum(k) =
1/abs(raizesnum(k))*cos(angle(raizesnum(k)))+j*1/abs(raizesnum(k))*s
in(angle(raizesnum(k)));
            disp('Invertendo o zero.');
```

end

```

    end
    % gera os polinômios com os novos valores do zero e do pólo
    novonum(1,:) = real(poly(raizesden));
    novoden(1,:) = real(poly(raizesnum));
end
```

4. Cálculo da saída em função da excitação dos filtros em cascata – modificado da função cascade de (STONICK; BRADLEY, 1996)

```

function saida = cascata(num,den,entrada);

% saida = cascata(num,den,entrada);
%
% Implementa filtros de segunda ordem em cascata
% chamando a função filter sucessivamente no sinal
% de entrada.
%
% num,den: coeficientes do filtro (gerado em criafiltro).
% entrada: sinal a ser filtrado
% ATENÇÃO:
% Esta função é baseada e modificada de
% 'Labs for signals and systems using Matlab' – V. STONICK & K.
BRADLEY

[n,m]=size(num);

% Numerador e denominador nulos
if n==0 & m==0,
    saida = entrada;
    return;
end

if num(1,1) == 0 & num(1,2) == 0,
    % Filtro só com ganho
    saida = filter(num(1,3),den(1,3),entrada);
elseif num(1,1) == 0,
    % Filtro com um pólo/zero
    saida = filter(num(1,2:3),den(1,2:3),entrada);
else
    saida = filter(num(1,:),den(1,:),entrada);
end
```

```

if n ~= 1,
    for k = 2:n,
        if num(k,1) == 0 & num(k,2) == 0,
            % Filtro só com ganho
            saida = filter(num(k,3),den(k,3),saida);
        elseif num(k,1) == 0,
            % Filtro com um pólo/zero
            saida = filter(num(k,2:3),den(k,2:3),saida);
        else
            saida = filter(num(k,:),den(k,:),saida);
        end
    end
end
end

```

5. Cria um dente de serra

```

function [tf,yf]=denteserra(fs,f,pts,tempo)
% Cria um dente de serra em função das frequências do sinal e de
amostragem
% [t,dente]=denteserra(fs,f,pontos,tempo);
% t = tempo na forma de vetor
% dente = dente de serra 50% crescente, 50% decrescente
% fs = frequência de amostragem
% f = frequência desejada do dente de serra
% pontos = n° de pontos
% tempo = tempo total do dente de serra
% se definir pontos>0 e tempo>0, a função usa o número de pontos
% pontos = 0 indica que é para considerar o tempo e vice-versa

% número de amostras por período do dente de serra
if tempo == 0 || (tempo>0 && pts>0)
    per = ceil(fs/f);
    nper = ceil(pts/per);
elseif pts == 0
    per = ceil(fs/f);
    nper = ceil(tempo*f);
else tempo == 0 && pts == 0
    return
end
y = linspace(-1,1,per);
yf = [];
for i = 2:nper
    yf = [yf y];
end
if nper == 1
    yf = y;
end
yf = [yf -1];
tf = linspace(0,length(yf)/fs,length(yf));

```

6. Espectro em frequência

```
function [H,f]=especfreq(sinal,fs)
% [H,f]=especfreq(sinal,fs)
% Esta função retorna o espectro em frequência de um sinal.
% H é a amplitude do espectro
% f são as frequências do espectro
% sinal é o que se deseja analisar
% fs é a frequência de amostragem
namostras = length(sinal);
%% Torna o comprimento do sinal uma potência de 2
for k = 1:100,
    if 2^k > namostras,
        elevadoa = k;
        break;
    end
end
namostras = 2^elevadoa;
N = 2*namostras;
impulso = [sinal(:)' zeros(1,namostras - length(sinal))];
% Calcula a transformada de Fourier
H = abs(fft(impulso,N));
f = (0:N-1)*fs/N;
```

7. Modelagem da excitação da corda do violão – modificado da função generate de (STONICK; BRADLEY, 1996)

```
function [output] = gerar(bodynum, bodyden,len);
% output = gerar(bodynum, bodyden, len);
%
% Sintetiza a resposta do modelo do violão calculando a resposta
% da corda e então filtrando pelo corpo do violão.
% Use valores nulos [] de bodynum e bodyden para criar a excitação
% da corda.
%
% bodynum, bodyden: coeficientes do filtro do corpo do violão
% len: comprimento em amostras do sinal que se deseja gerar.
% ATENÇÃO:
% Esta função é baseada e modificada de
% 'Labs for signals and systems using Matlab' - V. STONICK & K.
BRADLEY

% parâmetros adequados para fazer o som da corda

fprintf('Gerando os dados...\n');

% Dados da modelagem da corda...

Th = 222;
Bh = [ 0.01181465712323 -0.09931185121095 0.62296411573858
0.54818909354767 -0.09517274292938 0.01151672773084 ];
Tv = Th;
Bv = Bh;
Dh = -.99275;
```

```

Dv = -.945;
Mh = .24;
Mv = .75;
Zh = 7;
Zv = 65;
freq = 195.5931;

% Denominadores para filtros da corda
horden = [1 zeros(1,Th) Bh*Dh/(sum(Bh))];
verden = [1 zeros(1,Tv) Bv*Dv/sum(Bv)];

% Fourier Synthesis of string excitation
t = 0:Th+5;
excl = 0;
for q = 1:50,
    excl =
excl+(sin(2*q*pi/Zh)/((2*q*pi/Zh)*(2*q*pi/Zh)))*sin(2*q*freq*pi*t/44
100);
end;

% Actual string excitation: 1 period and zeros
hexc = [excl zeros(1,len-length(excl))];
hexc = hexc / max(hexc)*Mh;

% Fourier Synthesis of vertical excitation
t = 0:3*(Th+6)-3;
excl = 0;
for q = 1:50,
    excl =
excl+(sin(2*q*pi/Zh)/((2*q*pi/Zh)*(2*q*pi/Zh)))*sin(2*q*freq*pi*t/44
100);
end;

% Creation of vertical excitation
exc3 = (excl+randn(size(excl))/4000).*(1-exp(-
.006*(1:length(excl))))*0.5;
exc3 = exc3 / max(abs(exc3)) * Mv;
vexc = [exc3 zeros(1,len-length(exc3))];

disp('Filtering data...')

% Filtering operations: string models
horiz = filter(1,horden,hexc);
vert = filter(1,verden,vexc);

% Run the body filter
output = cascata(bodynum,bodyden,horiz+vert);

```

**APÊNDICE B – ARTIGO PUBLICADO NOS ANAIS DO XXV SIMPÓSIO
BRASILEIRO DE TELECOMUNICAÇÕES (SBrT 2007)**

Artigo publicado nos anais do XXV Simpósio Brasileiro de Telecomunicações, Recife, 03 a
06 de Setembro de 2007

Modelagem e simulação de instrumentos musicais utilizando filtros digitais

Gustavo de Melo Valeira e Marcio Eisencraft

Resumo—Este trabalho de iniciação científica apresenta os resultados da simulação por meio de filtros digitais de um instrumento musical de corda. A resposta em frequência do filtro é modelada a partir da Transformada de Fourier da resposta ao impulso do instrumento obtida batendo-se em uma das cordas. A comparação entre as saídas do filtro e do corpo do instrumento indica que este pode ser efetivamente modelado por filtros digitais.

Palavras-chave—Processamento de sinais musicais, modelagem de instrumentos musicais, filtros digitais.

Abstract—This short paper presents results of the simulation by digital filters of a musical string instrument. The frequency response of the filter is modeled using the Fourier Transform of the instrument's impulse response obtained beating in one of the strings. The comparison between filter and real instrument body outputs indicates that it can be effectively modeled by digital filters.

Keywords—Musical signal processing, musical instruments modeling, digital filters.

I. INTRODUÇÃO

A modelagem de instrumentos musicais utilizando filtros digitais tem gerado muitos artigos interessantes [1-3]. O objetivo deste trabalho é modelar digitalmente um baixo, criando uma entrada e um filtro que juntos gerem um sinal semelhante ao produzido pela corda Si deste instrumento. Os filtros digitais seletivos em frequência têm a mesma função que o corpo do baixo ao receber um som: deixar passar determinadas faixas de frequência e remover outras [4].

Sintetizadores simulam vários instrumentos musicais, como baixo, guitarra e violão. Para isso, usam filtros que criam uma determinada nota dependendo do instrumento selecionado pelo usuário [5]. Notas são os sons produzidos por instrumentos musicais e recebem nomes de acordo com sua frequência fundamental. As sete notas principais são: Dó, Ré, Mi, Fá, Sol, Lá e Si.

O uso de filtros digitais para a síntese de notas musicais tem uma série de vantagens práticas. Entre elas, destaca-se a facilidade em se mudar as características naturais de um instrumento, permitindo assim maiores variações do som gerado [5]. Além disso, podem-se simular diversos instrumentos musicais usando o mesmo equipamento, bastando para isso ajustar os coeficientes dos filtros para cada caso.

Na Seção II é examinada a obtenção da resposta em frequência do instrumento em questão, o baixo. Na Seção III é discutido o projeto do filtro que simula o corpo do instrumento. Na Seção IV são descritas simulações que permitem comparar o

desempenho do filtro frente ao instrumento musical. Por fim, na Seção V, são traçadas conclusões acerca dos resultados deste trabalho.

II. RESPOSTA EM FREQUÊNCIA DO INSTRUMENTO

Uma estimativa da resposta impulsiva do corpo do baixo ao se acionar a corda Si foi obtida batendo-se rapidamente na corda Si de um baixo Washburn® XB-125, de forma a simular um impulso. Para capturar este som foi usado um amplificador WarmMusic® 108rb e um microfone SoundMax® Superbeam® ligado a entrada de microfone de uma placa de som Realtek® AC'97 com frequência de amostragem de 44.1 kHz. O sinal normalizado obtido, $h_{si}(t)$, é mostrado na Figura 1.

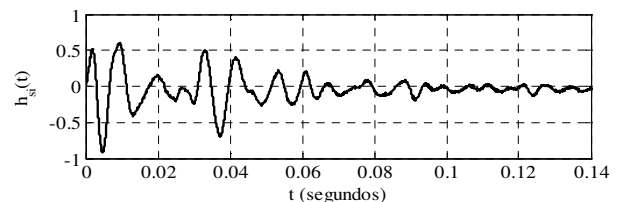


Fig. 1. Resposta ao impulso da corda Si do baixo.

Com a resposta impulsiva $h_{si}(t)$ é possível obter a resposta em frequência deste sistema, para que posteriormente projete-se um filtro digital que corresponda de forma satisfatória a essa resposta. Para isso foi usada a Transformada Rápida de Fourier [4] e obteve-se a resposta em frequência $H_{si}(f)$, cujo módulo é mostrado na Figura 2.

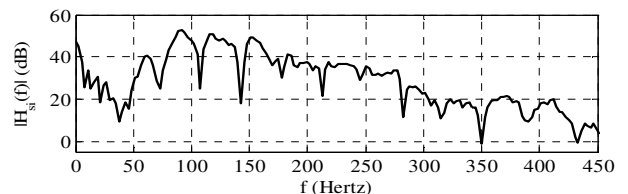


Fig. 2. Módulo da resposta em frequência da corda Si.

III. PROJETO DO FILTRO

O espectro em frequência da Figura 2 é usado como base para projetar um filtro digital correspondente ao corpo do baixo ao ser acionada a corda Si. Foram usados filtros de 2ª ordem em cascata para simular o corpo do instrumento, como feito em [5].

Pela Figura 2 verifica-se que este espectro possui muitos vales anti-ressonantes e picos ressonantes. Dessa forma, é neces-

sário projetar um filtro que selecione as faixas de frequência mais relevantes das notas que a corda Si produz. Foram usados sete filtros em cascata usando o método de projeto descrito em [5], em que foi modelado um violão.

O módulo da resposta em frequência $H_{Fsi}(f)$ do filtro projetado é mostrado na Figura 3. Esta resposta é semelhante à da Figura 2 nas faixas de frequência mais relevantes, ou seja, que contribuem mais para a resposta acústica do instrumento.

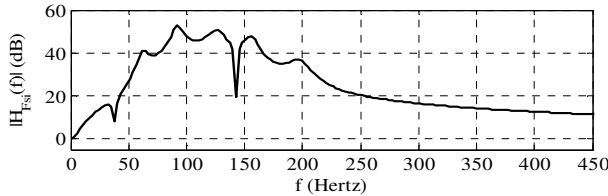


Fig. 3. Espectro em frequência do filtro projetado.

IV. SIMULAÇÕES

Como o corpo do baixo está sendo modelado pelo filtro, espera-se que com uma entrada adequada, a saída do filtro seja um sinal semelhante ao som original do baixo.

A corda ao ser tocada produz uma frequência fundamental e outras harmônicas, que são frequências múltiplas da fundamental. A excitação gerada pela vibração da corda no corpo do instrumento é simulada aqui por um sinal dente de serra com frequência igual à fundamental da nota. A amplitude da n -ésima harmônica no espectro deste sinal é proporcional a $1/n$ [4]. O sinal dente de serra foi escolhido porque mostrou ser uma boa solução experimentalmente.

O resultado da filtragem $y_{Filtro}(t)$ de um dente de serra de 30.87 Hz, que é a frequência fundamental da nota Si mais grave do baixo, é apresentado na Figura 4(b). Na Figura 4(a) é mostrada a forma temporal do sinal $y_{Si}(t)$ relativo à nota Si produzida pelo baixo.

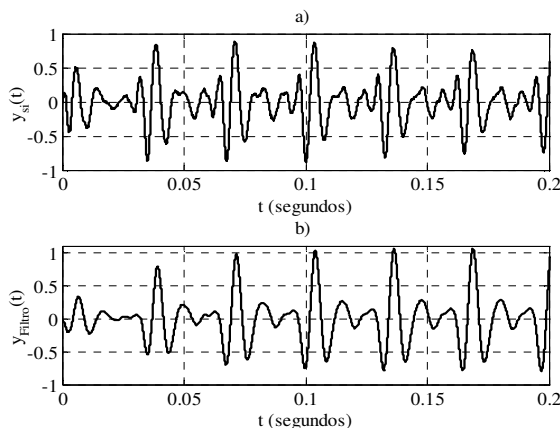


Fig. 4. a) Nota Si produzida pelo baixo; b) sinal gerado pelo filtro.

Na Figura 5 é apresentado o espectro em frequência dos dois sinais. As duas maiores amplitudes de ambos os espectros normalizados estão em torno da mesma frequência. O corpo do

baixo atenuou muito a frequência fundamental da nota Si, o que também aconteceu na filtragem do sinal dente de serra. A 2ª e 5ª harmônicas têm amplitudes semelhantes em ambos os espectros $Y_{Si}(f)$ e $Y_{Filtro}(f)$, porém a 6ª harmônica apresenta uma diferença maior. A partir da 7ª harmônica as amplitudes são praticamente desprezíveis nos dois espectros.

Ouvindo os dois sinais, nota-se que eles são parecidos, porém existe uma diferença entre eles. Esta é causada pelo decaimento natural, ou seja, o sinal gerado pelo baixo diminui de intensidade com o tempo, enquanto que no sinal filtrado isso não ocorre.

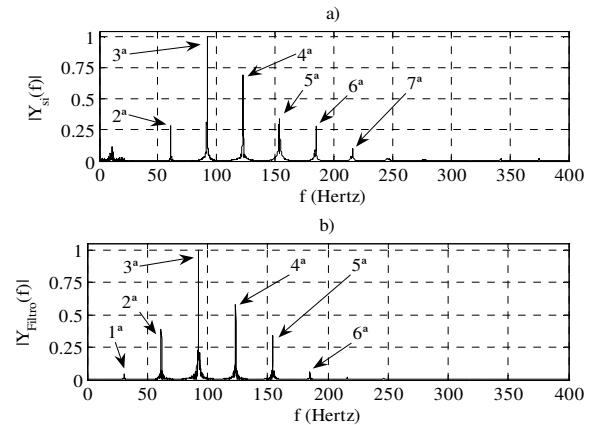


Fig. 5. Espectro em frequência normalizado: a) nota Si do baixo $y_{Si}(t)$; b) dente de serra filtrado $y_{Filtro}(t)$. São indicadas as posições das harmônicas em cada espectro.

V. CONCLUSÕES

Os resultados apresentados nas Figuras 4 e 5 mostram que um filtro digital pode modelar adequadamente o corpo de um baixo. Acusticamente os sinais obtidos na saída da cascata de filtros alimentados pelo dente de serra e o som real gerado pelo instrumento são parecidos. O dente de serra mostrou ser uma boa solução para a excitação do filtro.

O decaimento natural no domínio do tempo que ocorre ao tocar uma corda não foi levado em conta, pois precisa de uma análise mais detalhada de como modelar esse efeito nas cordas do instrumento. Este problema está em estudo atualmente.

REFERÊNCIAS

- [1] K. Bradley, M. Cheng and V. L. Stonick, "Automated analysis and computationally efficient synthesis of acoustic guitar strings and body", *IEEE ASSP Workshop on Applications of Signal Processing to Audio and Acoustics*, p. 238-241, Outubro 1995.
- [2] M. Karjalainen, V. Välimäki and Z. Jánosy, "Toward high-quality sound synthesis of the guitar and string instruments", *Proceedings of the ICUC*, p. 56-63, Setembro 1993.
- [3] C. Traube and J. O. Smith III, "Extracting the fingering and the plucking points on a guitar string from a recording", *IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics*, p. 7-10, 2001.
- [4] A. V. Oppenheim, R. W. Schaffer and J. R. Buck, *Discrete-time signal processing*. Prentice Hall, 1999.
- [5] V. Stonick and K. Bradley, *Labs for signals and systems using Matlab*. PWS BookWare Companion Series, 1996.

MODELAGEM E SIMULAÇÃO DE INSTRUMENTOS MUSICAIS UTILIZANDO FILTROS DIGITAIS

Gustavo de Melo Valeira e Marcio Eisencraft

{gusval; marcioft}@mackenzie.br

Escola de Engenharia

Universidade Presbiteriana Mackenzie

São Paulo - Brasil

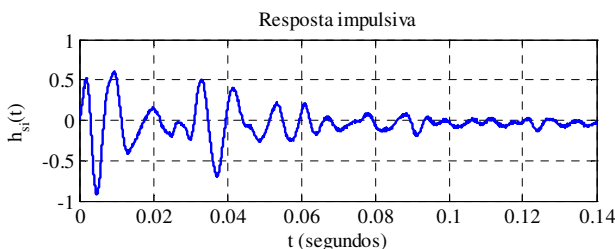
XXV SIMPÓSIO BRASILEIRO DE TELECOMUNICAÇÕES - SBTr'07

1 Objetivos

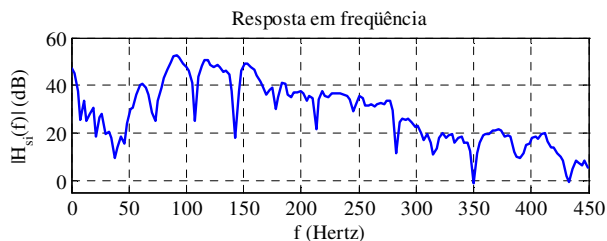
- Modelar digitalmente a corda Si de um baixo
- Obter um filtro digital e uma entrada que juntos gerem um sinal semelhante ao produzido pelo baixo

2 Resposta em frequência do instrumento

- Bateu-se rapidamente na corda Si do baixo para obter a resposta impulsiva

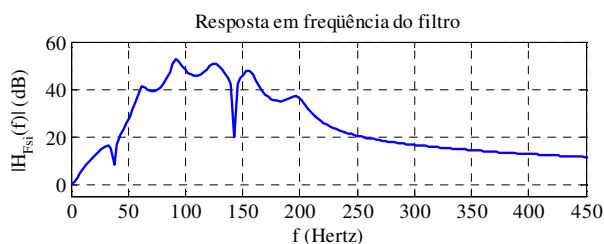


- Resposta em frequência é a TF da resposta impulsiva



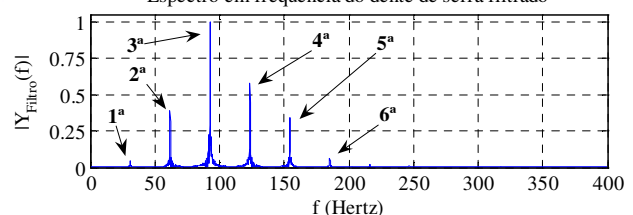
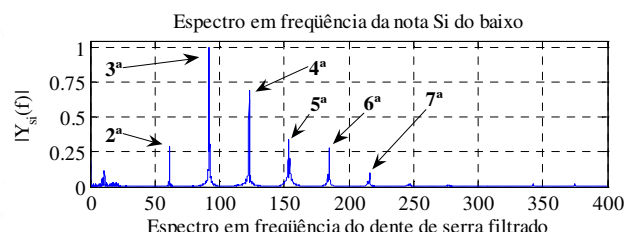
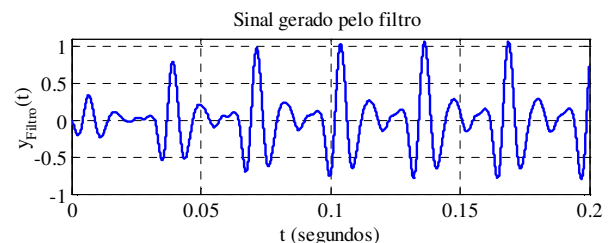
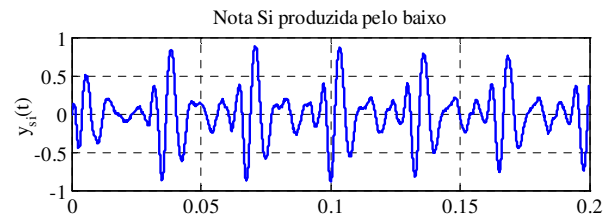
3 Projeto do filtro

- Filtro reproduz características do baixo nas frequências acusticamente mais relevantes
- Usados 7 filtros de 2ª ordem em cascata projetados como descrito por Stonick e Bradley (1996)



4 Simulações

- Foi usado como excitação do filtro um sinal dente de serra com frequência igual a da nota



5 Conclusões

- Acusticamente o sinal obtido na saída do filtro e o som real gerado pelo instrumento são parecidos
- Sinal dente de serra mostrou ser boa solução para a excitação do filtro
- É necessário uma análise mais detalhada de como modelar o decaimento natural no domínio do tempo