

**UNIVERSIDADE DO ESTADO DE SANTA CATARINA
CENTRO DE CIÊNCIAS TECNOLÓGICAS - CCT
BACHARELADO EM ENGENHARIA ELÉTRICA**

LUAN VINÍCIUS FIORIO

**EMULAÇÃO DE UM PRÉ-AMPLIFICADOR VALVULADO
UTILIZANDO REDES NEURAIS ARTIFICIAIS**

JOINVILLE

2019

LUAN VINÍCIUS FIORIO

**EMULAÇÃO DE UM PRÉ-AMPLIFICADOR VALVULADO
UTILIZANDO REDES NEURAIS ARTIFICIAIS**

Trabalho de Conclusão de Curso apresentado ao curso de Bacharelado em Engenharia Elétrica do Centro de Ciências Tecnológicas da Universidade do Estado de Santa Catarina, como requisito parcial para a conclusão do curso.

Orientador: Yales Rômulo de Novaes, Dr

Coorientador: Chrystian Lenon Remes, Ms

JOINVILLE

2019

Fiorio, Luan Vinícius

EMULAÇÃO DE UM PRÉ-AMPLIFICADOR VALVULADO UTILIZANDO
REDES NEURAIS ARTIFICIAIS / Luan Vinícius Fiorio. -
JOINVILLE, 2019.

119 p. : il. ; 30 cm.

Orientador: Yales Rômulo de Novaes, Dr

Coorientador: Chrystian Lenon Remes, Ms

Trabalho de Conclusão e Curso - Universidade
do Estado de Santa Catarina, Centro de Ciências
Tecnológicas, Programa de Pós-Graduação em Engenharia
Elétrica, JOINVILLE, 2019.

1. Identificação de sistemas. 2. Redes neurais
artificiais. 3. Pré-amplificador valvulado. 4.
Algoritmos de treinamento. 5. Topologia. I. de Novaes,
Dr, Yales Rômulo . II. Remes, Ms, Chrystian Lenon .
III. Universidade do Estado de Santa Catarina, Centro
de Ciências Tecnológicas, Programa de Pós-Graduação em
Engenharia Elétrica. IV. Título

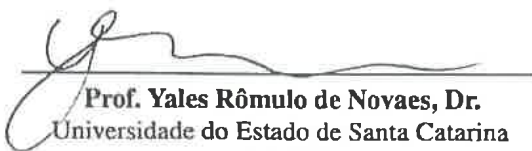
LUAN VINÍCIUS FIORIO

**EMULAÇÃO DE UM PRÉ-AMPLIFICADOR VALVULADO
UTILIZANDO REDES NEURAIS ARTIFICIAIS**

Trabalho de Conclusão de Curso apresentado ao curso de Bacharelado em Engenharia Elétrica do Centro de Ciências Tecnológicas da Universidade do Estado de Santa Catarina, como requisito parcial para a conclusão do curso.

Banca Examinadora:

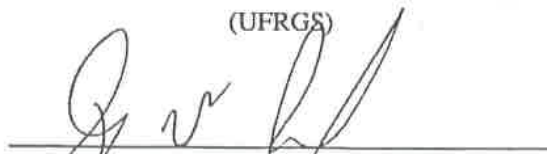
Orientador:


Prof. Yales Rômulo de Novaes, Dr.
Universidade do Estado de Santa Catarina
(UDESC)

Coorientador:


Prof. Chrystian Lenon Remes, Ms.
Universidade Federal do Rio Grande do Sul
(UFRGS)

Membros:


Prof. Douglas Wildgrube Bertol, Dr.
Universidade do Estado de Santa Catarina
(UDESC)


Prof. Renan Sebem, Ms.
Universidade do Estado de Santa Catarina
(UDESC)

JOINVILLE, 28 de junho de 2019

AGRADECIMENTOS

À minha família, pelo suporte, apoio e amor incondicional.

Aos meus atuais e antigos colegas do nPEE, pela ajuda sempre que necessária e pelas discussões cheias de *insights*, principalmente ao Ederson Antônio de Costa e ao Bruno Bertoldi, que acompanharam e fizeram parte do meu trabalho com áudio.

À Universidade do Estado de Santa Catarina, pela estrutura, laboratórios e professores que tornaram este trabalho, assim como minha jornada acadêmica, possível.

Aos meus amigos próximos da UDESC, à Ordem dos Tauras, aos membros remanescentes da antiga Raptor e aos meus amigos de infância, por todos os momentos que mantiveram minha sanidade durante o estudo da engenharia, assim como pelas amizades mais valiosas que eu poderia ter.

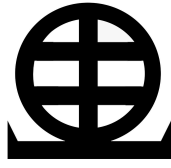
À cerveja e ao café, por serem os combustíveis do meu funcionamento, em quaisquer situações que se possa imaginar.

Aos professores de matemática/cálculo que tive durante a vida, principalmente Alexandre Veiga, Daniel Montibeller e José Rafael Furlanetto, por me ensinarem a mais importante das ciências.

Aos meus orientadores, pelo conhecimento que me passaram, por acompanharem meu trabalho de perto e por servirem de inspiração acadêmica e profissional.

Ao meu professor de guitarra, Manoel Carlos Busatto, pelos ensinamentos musicais que inspiraram minhas decisões tomadas até o presente momento.

Ao Tenacious D, por mudar o rumo da minha vida.



*"Like a day without the dawn
Like a ray void of the sun
Like a storm that brings no calm
I'm most complete yet so undone"
(Adam "Nergal" Darski)*

RESUMO

Este trabalho propõe utilizar redes neurais artificiais na emulação de um pré-amplificador valvulado destinado à guitarra elétrica. Estudos são realizados sobre os sinais de excitação, onde estes são definidos com base na teoria de identificação de sistemas. Para sobrepor o problema do *vanishing gradient* das redes neurais recorrentes treinadas com *back-propagation through time*, utiliza-se duas possíveis soluções: a mudança do algoritmo de treinamento para o filtro de Kalman estendido e a utilização de outra topologia de rede neural, a rede *long short-term memory*. Um teste é elaborado sobre a definição da estrutura de redes neurais e é aplicado à uma das redes utilizadas. Um pré-amplificador valvulado é projetado e simulado, a válvula triodo é modelada com base na literatura. Um protótipo também é confeccionado. Com base nos dados reais e de simulação, três redes neurais são utilizadas para identificar o pré-amplificador e os resultados são comparados de forma quantitativa e qualitativa, mostrando que a rede *long short-term memory* obtém resultados muito superiores aos das redes neurais recorrentes mais simples e que a utilização do filtro de Kalman estendido como algoritmo de treinamento não garante uma diminuição substancial do erro, somente acelera o processo de treinamento.

Palavras-chave: Identificação de sistemas. Redes neurais recorrentes. Amplificador valvulado.

ABSTRACT

This work proposes the use of artificial neural networks in the task of emulating a tube preamplifier for electric guitar. Studies are made about the excitation signals based on the system identification theory. To overcome the vanishing gradient problem of the most simple recurrent neural networks trained with the back-propagation through time algorithm, two solutions are proposed: the changing of the training technique to the extended Kalman filter; and the use of another network topology, the long short-term memory network. A test is developed about the structure definition of the networks, which is applied to one of the recurrent neural networks. A tube preamplifier is designed, simulated through a circuit simulation software using the triode vacuum-tube models that are available in the literature. A prototype is built. Based on the data from the prototype and the simulations, the preamplifier is identified by the three networks and the results are compared in a quantitative and qualitative form, showing that the long short-term memory network obtain better results than the simpler recurrent neural networks, and the use of the extended Kalman filter as the training algorithm does not reduce the error, only increase the training's speed.

Keywords: System identification. Recurrent neural networks. Tube amplifier.

LISTA DE ILUSTRAÇÕES

Figura 1 – Estrutura das válvulas triodo e pentodo.	26
Figura 2 – Exemplo de válvulas triodo e pentodo.	26
Figura 3 – Amplificador <i>K&F</i> da década de 1940.	27
Figura 4 – Amplificador <i>Marshall JCM800 2203</i>	27
Figura 5 – Amplificador <i>Mesa/Boogie Mark V</i>	28
Figura 6 – Amplificador <i>Roland JC-40 Jazz Chorus</i>	29
Figura 7 – Amplificador <i>BOSS Katana-50</i>	29
Figura 8 – Amplificador <i>Kemper Profiler</i>	30
Figura 9 – Comparação das componentes harmônicas entre diferentes topologias de pré-amplificadores.	31
Figura 10 – Diagrama de blocos do sistema de treino e teste da rede neural.	38
Figura 11 – Esquemática de uma rede de <i>reservoir computing</i>	39
Figura 12 – Modelo de um <i>perceptron</i>	48
Figura 13 – Modelo de um neurônio.	48
Figura 14 – Exemplo de uma rede <i>multilayer perceptron</i>	50
Figura 15 – Redes neurais recorrentes de Jordan e Elman.	51
Figura 16 – Rede recorrente de Jordan com uma camada oculta.	55
Figura 17 – Rede recorrente de Jordan desdobrada no tempo.	56
Figura 18 – Grafo de fluxo de sinal para um sistema dinâmico e linear.	61
Figura 19 – Célula LSTM.	66
Figura 20 – Modelo de Dempwolf de uma válvula 12AX7 no software <i>PowerSIM</i> ®. . .	69
Figura 21 – Alimentação dos anodos.	70
Figura 22 – Alimentação dos filamentos.	71
Figura 23 – Entrada do pré-amplificador.	72
Figura 24 – Primeiro estágio de ganho.	73
Figura 25 – Reta de carga para o resistor de 220 kΩ.	73
Figura 26 – Segundo estágio de ganho.	75
Figura 27 – Terceiro estágio de ganho.	76
Figura 28 – Reta de carga para o resistor de 100 kΩ.	76
Figura 29 – FMV <i>tonestack</i> , chave 3PDT e saída.	77
Figura 30 – Circuito entre o estágio 1 e 2.	79
Figura 31 – Circuito entre o estágio 2 e 3.	79
Figura 32 – Circuito entre o estágio 1 e 3.	80
Figura 33 – Protótipo do pré-amplificador valvulado.	81
Figura 34 – Sinal de saída do primeiro estágio.	81
Figura 35 – Sinal de saída do segundo estágio.	82
Figura 36 – Sinal de saída do terceiro estágio.	82
Figura 37 – Sinal de saída do pré-amplificador valvulado.	83

Figura 38 – FFT do sinal de saída do pré-amplificador valvulado.	83
Figura 39 – Esquemática completa do pré-amplificador valvulado.	84
Figura 40 – <i>Layout</i> da PCB do pré-amplificador valvulado.	85
Figura 41 – Sinal de entrada do pré-amplificador simulado.	86
Figura 42 – Saída do primeiro estágio de ganho para o canal 1.	87
Figura 43 – Saída do segundo e terceiro estágios de ganho para o canal 1.	88
Figura 44 – Saída do pré-amplificador para o canal 1.	88
Figura 45 – Saída do primeiro estágio de ganho para o canal 2.	89
Figura 46 – Saída do segundo estágio de ganho para o canal 2.	90
Figura 47 – Saída do terceiro e quarto estágios de ganho para o canal 2.	90
Figura 48 – Saída do pré-amplificador para o canal 2.	91
Figura 49 – Sinais utilizados no primeiro teste de definição de parâmetros da rede neural.	94
Figura 50 – Árvore de testes de parâmetros.	95
Figura 51 – Sinal desejado e emulado pela rede neural de menor MSE no teste de parâmetros.	97
Figura 52 – MSEs das 150 execuções da rede neural com diferentes pesos iniciais.	98
Figura 53 – MSE de validação das redes RNN-EKF e RNN-BPTT para um <i>sine-sweep</i> de amplitude 0,3 V e baixo ganho.	105
Figura 54 – Trecho do sinal emulado utilizando os <i>sine-sweeps</i> no treinamento.	107
Figura 55 – Trecho do sinal emulado utilizando os sinais de ruído no treinamento.	108
Figura 56 – Trecho do sinal emulado utilizando os sinais de guitarra no treinamento.	108

LISTA DE ABREVIATURAS E SIGLAS

EDO	Equação diferencial ordinária
RI	Resposta ao impulso
RNA	Rede neural artificial
RNR	Rede neural recorrente
LSTM	<i>Long short-term memory</i>
AR	Modelo auto-regressivo
ARX	Modelo auto-regressivo com entrada exógena
ARMA	Modelo auto-regressivo com média móvel
ARMAX	Modelo auto-regressivo com média móvel e entradas exógenas
NARX	Modelo auto-regressivo não linear com entradas exógenas
NARMAX	Modelo auto-regressivo não linear com média móvel e entradas exógenas
PE	Persistentemente excitante
MSE	Erro médio quadrático
MLP	<i>Multilayer perceptron</i>
BPTT	<i>Back-propagation through time</i>
eBPTT	<i>Epochwise back-propagation through time</i>
tBPTT	<i>Truncated back-propagation through time</i>
EKF	<i>Extended Kalman filter</i>
CEC	<i>Constant error carousel</i>
RNN-BPTT	Rede neural recorrente treinada com <i>back-propagation through time</i>
RNN-EKF	Rede neural recorrente treinada com <i>extended Kalman filter</i>
B-LSTM	Rede <i>long short-term memory</i> treinada com <i>back-propagation through time</i>

LISTA DE SÍMBOLOS

$\mathbf{y}$	Vetor de saída
$\hat{\mathbf{y}}$	Vetor de saída estimado
$\boldsymbol{\theta}$	Vetor de parâmetros do mínimos quadrados
$\hat{\boldsymbol{\theta}}$	Vetor de parâmetros estimados do mínimos quadrados
F	Função não linear
v	Valor instantâneo do campo local induzido
w	Peso sináptico
b	<i>Bias</i>
x	Sinal de entrada
y	Sinal de saída
φ, f	Função de ativação não linear
U	Saída de um neurônio realimentado
e	Erro
$\mathcal{E}$	Energia do erro
$\mathcal{E}_{med.}$	Energia média do erro
η	Taxa de aprendizagem
j	Índice de um neurônio
$\mathcal{C}$	Conjunto de neurônios de saída
i, n, r	Índice de amostras
i_o	Início de uma batelada de dados
i_1	Fim de uma batelada de dados
$\mathcal{A}$	Conjunto de realimentações
δ	Gradiente local induzido
h	Profundidade temporal do <i>back-propagation through time</i>
$\mathbf{w}$	Vetor de estados do sistema

d	Vetor de observação
$\hat{d}$	Vetor de estimação
C	Matriz de medição
v	Ruído de medição
R	Matriz de covariância do ruído do sistema
α	Inovação
K	Matriz de covariância do erro
Γ	Fator de conversão que relaciona a estimativa de erro às inovações
G	Ganho de Kalman
c	Função não linear que representa a não linearidade na saída de uma rede neural recorrente
V_{ak}	Corrente anodo-catodo de uma válvula triodo
V_{gk}	Corrente grade-catodo de uma válvula triodo
I_a	Corrente no anodo de uma válvula triodo
I_k	Corrente no catodo de uma válvula triodo
I_g	Corrente na grade de uma válvula triodo
G, μ, γ, C	Parâmetros do modelo de Dempwolf para a corrente do catodo de uma válvula triodo
G_g, ξ, C_g, I_{g0}	Parâmetros do modelo de Dempwolf para a corrente da grade de uma válvula triodo

SUMÁRIO

1	INTRODUÇÃO	23
2	REVISÃO BIBLIOGRÁFICA: AMPLIFICADORES DE GUITARRA E EMULAÇÃO DE SISTEMAS	25
2.1	TIPOS DE AMPLIFICADORES	25
2.1.1	Valvulados	26
2.1.2	Transistorizados, digitais e híbridos	28
2.2	DISTORÇÃO EM AMPLIFICADORES DE GUITARRA ELÉTRICA . . .	30
2.3	TÉCNICAS PARA EMULAÇÃO E IDENTIFICAÇÃO DE AMPLIFICA- DORES VALVULADOS	31
2.3.1	Mínimos quadrados	32
2.3.2	<i>Static waveshaping</i>	32
2.3.3	<i>Wave digital filters</i>	33
2.3.4	Métodos analíticos	34
2.3.4.1	<i>Método K</i>	34
2.3.4.2	<i>Modified blockwise</i>	34
2.3.5	Técnicas baseadas em convolução	35
2.3.5.1	<i>Resposta ao impulso variável</i>	35
2.3.5.2	<i>Núcleos de Volterra</i>	36
2.3.5.3	<i>Núcleos de Hammerstein excitado por senoide de fase variável</i>	36
2.3.6	Redes neurais artificiais	37
2.3.6.1	<i>Redes neurais recorrentes</i>	37
2.3.6.2	<i>Computação reservatória</i>	38
2.3.6.3	<i>Long short-term memory</i>	40
2.4	CONSIDERAÇÕES FINAIS DO CAPÍTULO	40
3	IDENTIFICAÇÃO DE SISTEMAS	43
3.1	ESTRUTURA DOS MODELOS	43
3.1.1	Modelo auto-regressivo	43
3.1.2	Modelo auto-regressivo com entrada exógena	44
3.1.3	Modelo auto-regressivo com média móvel e entradas exógenas	44
3.1.4	Modelo NARX e NARMAX	45
3.2	SINAIS DE EXCITAÇÃO	45
4	REDES NEURAIS ARTIFICIAIS	47
4.1	O NEURÔNIO ARTIFICIAL	47
4.2	<i>MULTILAYER PERCEPTRON</i>	49
4.3	REDES NEURAIS RECORRENTES	50

4.3.1	Redes de Jordan e Elman	50
4.4	TÉCNICAS DE APRENDIZADO DE REDES NEURAIS RECORRENTES	52
4.4.1	<i>Back-propagation</i>	52
4.4.2	<i>Back-propagation through time</i>	54
4.4.2.1	<i>Unfolding in time</i>	54
4.4.2.2	<i>Epochwise back-propagation through time</i>	57
4.4.2.3	<i>Truncated backpropagation through time</i>	58
4.4.2.4	<i>Vanishing gradient</i>	59
4.4.3	<i>Filtro de Kalman estendido</i>	60
4.4.3.1	<i>Filtro de Kalman</i>	60
4.4.3.2	<i>Filtro de Kalman estendido</i>	63
4.5	<i>LONG SHORT-TERM MEMORY</i>	64
4.5.1	<i>Constant error carousel</i>	64
4.5.2	A célula de memória	65
5	PROJETO E SIMULAÇÃO DO PRÉ-AMPLIFICADOR VALVULADO	67
5.1	MODELO DE SIMULAÇÃO	67
5.1.1	Modelo de Dempwolf e sua simulação	67
5.2	PROJETO DO PRÉ-AMPLIFICADOR	69
5.2.1	Alimentação	70
5.2.2	Entrada do pré-amplificador	71
5.2.3	Primeiro estágio de ganho	72
5.2.4	Segundo estágio de ganho	74
5.2.5	Terceiro estágio de ganho	75
5.2.6	<i>Tonestack</i> e saída	76
5.2.6.1	<i>Seleção de canais</i>	78
5.2.6.2	<i>Saída do pré-amplificador</i>	78
5.2.7	Circuito entre estágios 1 e 2	78
5.2.8	Circuito entre estágios 2 e 3	79
5.2.9	Circuito entre estágios 1 e 3	80
5.3	RESULTADOS DO PROTÓTIPO	80
5.4	SIMULAÇÃO DO PRÉ-AMPLIFICADOR	86
5.4.1	O canal de baixo ganho	87
5.4.2	O canal de alto ganho	89
6	DEFINIÇÃO DA ESTRUTURA DA REDE NEURAL	93
6.1	TESTES PARA DEFINIÇÃO DA ESTRUTURA DAS REDES NEURAIS	93
6.1.1	Definição de faixa de parâmetros	93
6.1.2	Refinamento dos parâmetros para definição final da estrutura	97
6.1.3	Análise de convergência da estrutura final	98

7	RESULTADOS EXPERIMENTAIS	101
7.1	DADOS DE IDENTIFICAÇÃO	101
7.1.1	Dados de simulação	101
7.1.2	Dados do protótipo	102
7.2	RESULTADOS EXPERIMENTAIS DE IDENTIFICAÇÃO	103
8	CONCLUSÃO	109
	REFERÊNCIAS	111
	APÊNDICES	115
	APÊNDICE A – CIRCUITO DO CANAL DE BAIXO GANHO DO PRÉ-AMPLIFICADOR VALVULADO IMPLEMEN- TADO NO SOFTWARE <i>POWERSIM</i>®	117
	APÊNDICE B – CIRCUITO DO CANAL DE ALTO GANHO DO PRÉ- AMPLIFICADOR VALVULADO IMPLEMENTADO NO SOFTWARE <i>POWERSIM</i>®	119

1 INTRODUÇÃO

Amplificadores valvulados são utilizados na indústria musical desde a década de 1940. Tipicamente, no caso de amplificação para a guitarra elétrica, utilizados de forma que o sinal de saída esteja saturado, ocasionando em um efeito sonoro denominado distorção. As válvulas distorcem o sinal de uma forma diferente que os componentes de estado sólido e os ouvintes costumam preferir o som "valvulado". Esta preferência é, provavelmente, devida a popularização dos amplificadores valvulados nas décadas de 1950 e 1960 por bandas de *rock and roll*, fazendo com que os ouvintes se acostumassem com a sonoridade produzida pelas válvulas (PAKARINEN; YEH, 2009). Hamm (1972) mostra que as válvulas distorcem o sinal de forma assimétrica, resultando em um sinal com mais harmônicas de ordem par do que as geradas por um transistor, que são normalmente de ordem ímpar. Musicalmente, a maior parte das harmônicas pares equivalem à oitavas musicais.

Apesar da qualidade da distorção das válvulas, amplificadores valvulados possuem alguns problemas: são volumosos e pesados, tem baixa durabilidade, consomem muita potência, são pouco eficientes, possuem um custo muito elevado e, comumente, os componentes não são facilmente encontrados (PAKARINEN; YEH, 2009). A principal parte de um amplificador responsável por gerar sua distorção é o pré-amplificador, composto por válvulas triodo. O comportamento das válvulas triodo, assim como suas características são observados nos Capítulos 2 e 5.

A utilização de amplificadores transistorizados para a guitarra popularizou-se em meados da década de 1970. Os benefícios de um amplificador transistorizado estão, principalmente, relacionado à suas características físicas, como peso e durabilidade, e ao fato de serem mais baratos e eficientes. No entanto, a distorção gerada por componentes de estado sólido é, em sua maioria, originada de componentes harmônicas de ordem ímpar. Tais harmônicas costumam ter uma sonoridade mais dissonante que harmônicas pares, sendo menos interessante ao ouvido humano (HAMM, 1972). Assim, o próximo passo da evolução dos amplificadores de guitarra é aliar as características positivas dos valvulados a dos transistorizados. Esta junção foi feita de duas formas, utilizando um pré-amplificador valvulado e um amplificador de potência transistorizado simultaneamente ou emulando a sonoridade dos amplificadores valvulados, seja utilizando técnicas de identificação de sistemas ou simplesmente tentando recriar as características "sonoras" do amplificador através de processamento digital de sinais. Uma vasta revisão sobre amplificadores valvulados, transistorizados e técnicas utilizadas na emulação destes é apresentada no Capítulo 2.

De fato, com a evolução da tecnologia, tornou-se viável emular um sistema de áudio não linear dinâmico no tempo. A tarefa de emular um sistema com base em um modelo é denominada identificação de sistemas e é abordada no Capítulo 3. As técnicas de identificação de sistemas não lineares, no entanto, são normalmente de alta complexidade e necessitam de um processamento computacional elevado para serem executadas (AGUIRRE, 2015). Uma das alternativas para identificar o sistema não linear reside na utilização de redes neurais artificiais

como modelo para emulação do sistema. As redes neurais possuem uma característica não linear devido às suas funções de ativação - funções presentes dentro de um neurônio ou célula - e assim, são promissoras na identificação de sistemas não lineares (HAYKIN, 1999). Tais redes, após treinadas, podem ser executadas de forma *offline*, como se fossem o modelo do sistema.

As redes neurais artificiais, abordadas no Capítulo 4, são aproximadoras universais de funções (HORNIK; STINCHCOMBE; WHITE, 1989) e a utilização de recorrências aumenta sua capacidade de identificar sistemas variantes no tempo, adicionando dependência temporal à rede neural (HAYKIN, 1999). Teoricamente, redes neurais recorrentes são uma ótima opção para serem utilizadas na emulação de amplificadores valvulados, visto que possuem um comportamento dinâmico não linear e são dinâmicas no tempo. No entanto, realizando o treinamento delas por *back-propagation through time*, algoritmo mais comum de treinamento de redes neurais recorrentes, surge o problema do *vanishing gradient*. Este fenômeno resulta em um impedimento da continuação do treinamento, fazendo com que os parâmetros não sejam mais atualizados (HOCHREITER; SCHMIDHUBER, 1997).

É apresentada por Hochreiter e Schmidhuber (1997) uma topologia de rede neural recorrente - a rede *long short-term memory* - que é capaz de eliminar, por completo, o problema do *vanishing gradient*. Porém, esta rede neural demanda de um grande poder computacional devido a sua complexidade. Uma alternativa para amenizar os efeitos do *vanishing gradient* seria modificar a técnica de treinamento para um algoritmo que aproveite melhor a informação disponível: o filtro de Kalman estendido (HAYKIN, 1999). Neste trabalho, são abordadas e comparadas, na emulação de um pré-amplificador valvulado, a rede neural recorrente treinada por *back-propagation through time*, a rede neural recorrente treinada por filtro de Kalman estendido, e a rede *long short-term memory*. Esta última é abordada de acordo com a implementação realizada em (SCHMITZ; EMBRECHTS, 2018), por ser a uma das utilizações deste tipo de rede com maior sucesso na emulação de amplificadores valvulados atualmente.

O Capítulo 6 apresenta uma rotina de testes desenvolvida para determinar a estrutura de uma rede neural recorrente - número de camadas, número de neurônios, função de ativação, função de ativação do neurônio de saída e taxa de aprendizagem. A estrutura determinada neste capítulo será aplicada às três redes neurais escolhidas. Com base em diferentes sinais de excitação, o pré-amplificador simulado e real - apresentado no Capítulo 5 - é identificado pelas redes e os resultados são apresentados no Capítulo 7, onde a rede neural recorrente treinada pelo filtro de Kalman estendido obtém resultados semelhantes à mesma treinada pelo *back-propagation through time*. A rede *long short-term memory* obtém os melhores resultados entre as três redes, visto que esta foi desenvolvida para corrigir o problema estrutural de suas antecessoras.

2 REVISÃO BIBLIOGRÁFICA: AMPLIFICADORES DE GUITARRA E EMULAÇÃO DE SISTEMAS

A era da eletrônica foi iniciada com a invenção da válvula em 1902 por John Fleming, com aplicação imediata em rádio-comunicação (MACKSEY; WOODHOUSE, 1991). Na década de 1930, foram criados os primeiros amplificadores valvulados para uso com uma guitarra elétrica (FRYETTE, 2016). A popularização de tais amplificadores possibilitou que mais empresas surgissem no ramo, como a *Fender Musical Instruments Corporation* em 1945, a *Vox Amplification* ao final dos anos 50 e a *Marshall Amplification* em 1962.

Em uma próxima linha de tempo, um dispositivo formado por três camadas de materiais semicondutores com tipos alternados foi criado em 1948, o transistor, por um grupo de pesquisa liderado por William Shockley na empresa Bell Telephone (BRAGA, c.a. 2010), ocasionando a primeira revolução da eletrônica após o surgimento da válvula. Os transistores são comumente usados para amplificação de sinais e comutação.

A evolução tecnológica proporcionou diversas mudanças na área de eletrônica, inclusive no ramo de áudio. Nesta revisão são apresentados ambos os casos, mostrando diferentes modelos de amplificadores e uma breve descrição de cada um. Uma análise quantitativa e qualitativa dos tipos de distorção gerados por amplificadores valvulados e transistorizados também é abordada.

Na falta de meios viáveis de emular, com sucesso, a sonoridade das válvulas de forma analógica, pesquisadores partiram para formas digitais de emulação. Assim, métodos relevantes para a emulação de áudio não linear (i.e. amplificadores valvulados) são expostos ao leitor. As técnicas apresentadas são selecionadas com base em critérios como a eficácia destas em emular tais amplificadores, observando esforço computacional e qualidade dos resultados obtidos, assim como sua popularidade no que tange a área de emulação de áudio não linear. Também são mostrados alguns resultados obtidos por pesquisadores que aplicaram tais métodos, de forma quantitativa em sua maioria.

2.1 TIPOS DE AMPLIFICADORES

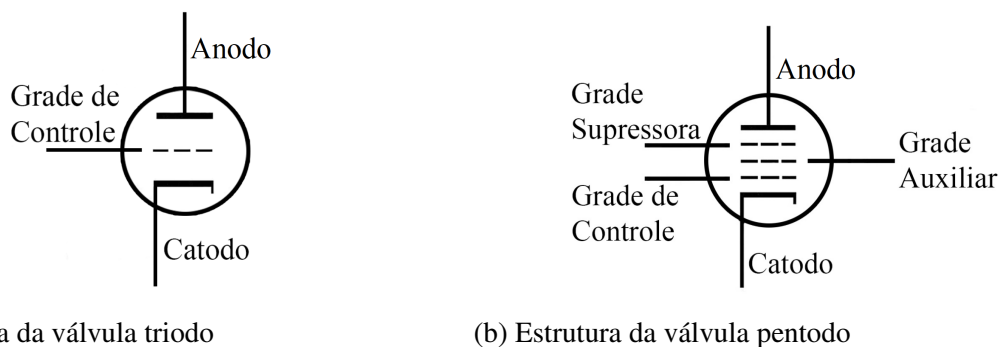
Os transistores possuem características positivas sobre às válvulas no quesito dos amplificadores de áudio, destacando-se tamanho, custo, peso e acessibilidade. Porém, dentre as restritas desvantagens relevantes ao tópico de áudio, a principal se dá pela característica da distorção deste dispositivo - esta que é comumente utilizada em amplificadores de instrumentos musicais - que leva a uma sonoridade diferente do tipo de distorção gerado pelas válvulas.

Nesta seção, são apresentados seis modelos de amplificadores, sendo três destes valvulados e os três restantes transistorizados ou digitais. É realizada uma breve análise sobre suas características, visando a percepção da evolução decorrida nas últimas décadas no escopo de amplificadores para instrumentos musicais, mais especificamente, para a guitarra elétrica.

2.1.1 Valvulados

A válvula consiste em um tubo de vidro com vácuo ou gases em seu interior, contendo um anodo e um catodo, além de grades metálicas. Para que elétrons fluam de um polo ao outro, é necessário que a válvula seja aquecida (emissão termiônica), operação realizada por uma resistência interna denominada filamento aquecedor (PAKARINEN; YEH, 2009). A válvula pode ser diodo, triodo, tetrodo ou pentodo, sendo respectivamente formada por: anodo e catodo; anodo, catodo e grade de controle; anodo, catodo, grade de controle, grade auxiliar e grade de supressão (BLENCOWE, 2009). A grade de controle serve para controlar o fluxo de elétrons entre os polos, enquanto a grade auxiliar acelera os mesmos. A grade de supressão impede que elétrons repelidos pelo polo receptor sejam emitidos. A estrutura das válvulas triodo e pentodo pode ser vista conforme a Figura 1, enquanto um exemplo da forma física de cada uma está presente na Figura 2.

Figura 1 – Estrutura das válvulas triodo e pentodo.



(a) Estrutura da válvula triodo

(b) Estrutura da válvula pentodo

Fonte: (BERTOLDI, 2016).

Figura 2 – Exemplo de válvulas triodo e pentodo.



(a) Válvula triodo (modelo 12AX7)



(b) Válvulas pentodo (modelo 6L6)

Fonte: (MESA/BOOGIE, 2018b).

A estrutura dos amplificadores valvulados é comumente dividida em pré-amplificador e amplificador de potência. No primeiro caso, normalmente utilizam-se válvulas do tipo triodo, amplificando e distorcendo, no caso de amplificadores destinados à guitarras, o sinal do instrumento musical por meio da saturação das válvulas e dos diversos filtros de equalização, atingindo um nível de amplitude adequada para a próxima etapa. No segundo caso, tem-se o amplificador de potência, que usualmente é composto por válvulas pentodo e também adiciona distorção ao

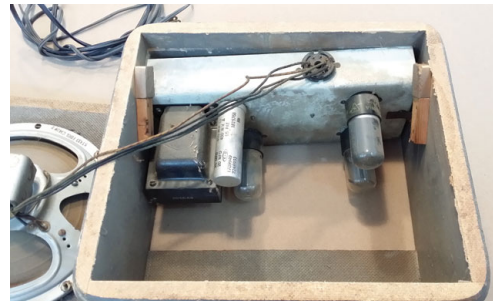
sinal, porém de forma mais sutil. Sua principal função é amplificar o sinal anterior de forma que a potência do mesmo seja capaz de excitar a membrana de um auto falante.

O primeiro amplificador relevante ao tópico foi comercializado pela *Kauffman & Fender Amplifiers*. Este amplificador era destinado ao uso com *lap steel guitars*, tipo de guitarra que precedeu o modelo que se consagrou no mercado na década de 1950. O amplificador denominado *K&F Amplifier* possuía três válvulas, sendo estas para retificação, pré-amplificação e amplificação de potência, totalizando 3 W no total (FJESTAD, 2016). Não havia nenhum tipo de equalização no próprio amplificador, característica que surgiu anos depois. O mesmo pode ser observado conforme a Figura 3.

Figura 3 – Amplificador *K&F* da década de 1940.



(a) Parte frontal do amplificador *K&F*



(b) Parte traseira do amplificador *K&F*

Fonte: (FJESTAD, 2016).

Avançando na linha do tempo, tem-se o renomado *Marshall JCM800 2203* (Figura 4), modelo inicialmente lançado em 1981 pela *Marshall Amplifiers*, sucessor do antigo *Marshall 2203* de 1975 (MARSHALL, 2018). O amplificador conta com três válvulas triodo em seu pré-amplificador e quatro válvulas pentodo, entregando 100 W de potência no seu estágio final. Neste caso, já percebe-se a presença de um *tone stack*, filtro equalizador de três bandas (grave, médio e agudo), e de um controle de ganho.

Figura 4 – Amplificador *Marshall JCM800 2203*.



Fonte: (MARSHALL, 2018).

O *JCM800* é mundialmente reconhecido como um dos amplificadores mais icônicos já criados (MARSHALL, 2017). Nomeado conforme as iniciais de seu criador, *Jim Charles Marshall*, este amplificador destaca-se pela simplicidade e facilidade de utilização, assim como pela alta qualidade de construção e boa qualidade sonora.

Em 2009, a empresa californiana *Mesa Engineering* (também conhecida como *Mesa/Boogie*) lançou um sucessor à série *Mark* de amplificadores de guitarra, denominado de *Mark V* (Figura 5). Em seu pré-amplificador, estão presentes sete válvulas triodo, enquanto no estágio de potência, situam-se quatro válvulas pentodo, com potência que pode variar de acordo com a escolha do usuário, de 10 W à 90 W. Percebe-se neste caso a presença de diversos filtros de equalização e tipos de sonoridade, mostrando que foi possível incluir diversas características extras em um equipamento sem alterar o seu tamanho significativamente.

Figura 5 – Amplificador *Mesa/Boogie Mark V*.



Fonte: (MESA/BOOGIE, 2018b).

A evolução dos amplificadores valvulados deu-se principalmente por necessidades dos instrumentistas, seja esta por diferentes sonoridades ou por flexibilidade, como percebe-se conforme a breve linha do tempo apresentada acima. Há de se enfatizar que, a partir do surgimento dos primeiros amplificadores valvulados destinados ao tipo de guitarra elétrica que domina o mercado nos dias atuais, a tecnologia envolvida nestes foi pouco ou em nada modificada.

2.1.2 Transistorizados, digitais e híbridos

Os amplificadores de estado sólido (ou transistorizados) destinados a instrumentos musicais começaram a ganhar força em meados de 1970 (SOLLO, 2016). Em 1975, a *Roland* lançava seu mais famoso amplificador de estado sólido, o *JC-40 Jazz Chorus*. Lançado como *combo* - amplificador e caixa com auto falante juntos em um mesmo produto, diferentemente dos cabeçotes que só incluem o amplificador em si - este amplificador é reconhecido por ser a base para sonoridades limpas (sem distorção) dos últimos quarenta anos (MUSICCRITIC, 2018).

O *JC-40 Jazz Chorus* possui equalização de grave, médio e agudo, reverberação por mola, diversos efeitos analógicos e oferece 40 W de potência em sua versão mais comum. Este

amplificador pode ser observado na Figura 6.

Figura 6 – Amplificador *Roland JC-40 Jazz Chorus*.



Fonte: (MUSICCRITIC, 2018).

No início de 2017 a *BOSS*, renomada pelos seus pedais de efeitos para guitarra elétrica, lançou uma nova linha de amplificadores transistorizados/digitais chamada *Katana*. Os amplificadores desta linha variam de modelos com 50 a 100 W no seu estágio de potência classe AB, possuindo diversos tipos de efeitos e distorções analógicas e digitais, customização de efeitos via *software* pelo próprio usuário, equalização e controle de potência para diferentes volumes (BOSS, 2017). De acordo com White (2017), o *BOSS Katana* soa de forma "natural e real"¹, contando com um custo benefício muito elevado. O amplificador pode ser observado conforme a Figura 7.

Figura 7 – Amplificador *BOSS Katana-50*.



Fonte: (BOSS, 2017).

¹ A percepção de tais características é altamente subjetiva e, neste contexto, fazem referência aos amplificadores valvulados, considerados mais "naturais e reais" em questão de sonoridade.

A necessidade dos guitarristas de obter uma sonoridade distorcida de qualidade, seja em seu quarto, no estúdio ou no palco é um motivo de frustração desde o surgimento dos amplificadores transistorizados (KEMPERAMPS, c.a. 2016). O *Kemper Profiler* é apresentado como solução para este problema, onde o usuário pode "copiar", de forma digital, qualquer amplificador disponível simplesmente com base na entrada e saída (por meio de um microfone) do mesmo. Este conceito será apresentado de forma mais precisa na próxima seção.

O *Kemper Profiler* (Figura 8) possui um amplificador de potência classe D de 600 W de potência, enquanto que a pré-amplificação é realizada digitalmente. O amplificador conta com várias opções de equalização e diversos efeitos digitais.

Figura 8 – Amplificador *Kemper Profiler*.



Fonte: (KEMPERAMPS, c.a. 2016).

A evolução da eletrônica afetou diretamente os amplificadores de guitarra não valvulados, seja por meio de circuitos analógicos ou de efeitos e emulações digitais. Percebe-se, portanto, que a implementação de emulações digitais nos pré-amplificadores dos dias atuais é uma característica "em alta", por ser possível atingir uma sonoridade de maior qualidade, enquanto que o uso de amplificadores de potência classe D é tido como a opção mais viável em termos de peso, tamanho e eficiência.

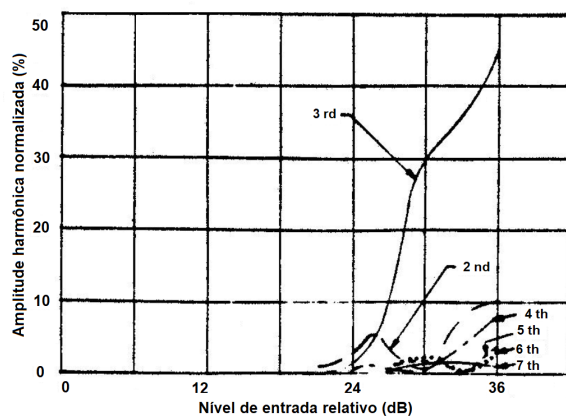
2.2 DISTORÇÃO EM AMPLIFICADORES DE GUITARRA ELÉTRICA

A distorção harmônica compreende a deformação ou modificação de um sinal de saída causada pela presença de frequências que não estão presentes no sinal de entrada (MILLET, 2004). Estas harmônicas podem ser divididas em dois tipos principais, harmônicas de ordem par (segunda, quarta, etc.) e harmônicas de ordem ímpar (terceira, quinta, etc.).

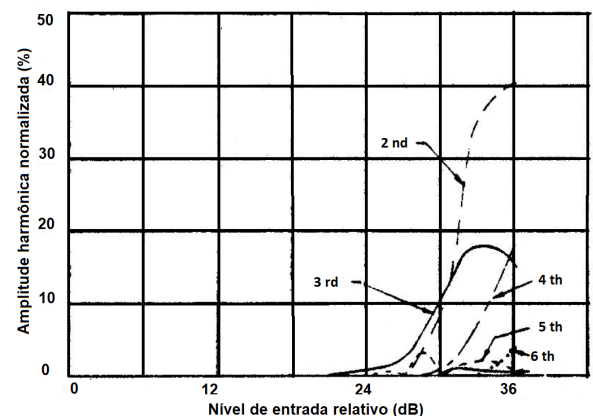
As harmônicas de ordem par são resultado de não linearidades assimétricas em uma função de transferência, enquanto as ímpares resultam da simetria presente nas não linearidades da função. Em circuitos reais, ambas as não linearidades estão presentes. Ao mesmo tempo, sabe-se que harmônicas de segunda ordem correspondem às oitavas (musicais) referentes à frequência ou nota fundamental (SUNDBERG; LINDQVIST, 1973).

Conforme a Figura 9a, percebe-se que a componente harmônica dominante gerada pela distorção em um pré-amplificador de estado sólido é a terceira. Enquanto isso, na Figura 9b percebe-se o domínio da segunda harmônica. Tal fato não é uma regra e dependerá do projeto dos circuitos em questão, porém deve-se saber que as componentes harmônicas pares, no caso de um pré-amplificador, são geradas pelas características inerentes às válvulas triodo, por representarem sistemas não lineares, enquanto as mesmas não estão tão presentes em dispositivos de estado sólido.

Figura 9 – Comparação das componentes harmônicas entre diferentes topologias de pré-amplificadores.



(a) Componentes harmônicas de um pré-amplificador de estado sólido



(b) Componentes harmônicas de um pré-amplificador valvulado

Fonte: adaptado de (HAMM, 1972).

Portanto, infere-se que os amplificadores valvulados tem uma sonoridade, geralmente, mais consonante² se comparados aos amplificadores de estado sólido. A preferência humana por acordes (ou sonoridades) consonantes, conforme Lots e Stone (2008), é justificada pelo costume e exposição da sociedade à práticas musicais ocidentais. Assim, estendendo esta ideia para a linha de amplificadores para guitarra elétrica, assume-se que uma parte da preferência dos guitarristas por estes tipos de amplificadores é dado pelo costume e influências de artistas, não somente pelas características físicas dos componentes utilizados nos mesmos.

2.3 TÉCNICAS PARA EMULAÇÃO E IDENTIFICAÇÃO DE AMPLIFICADORES VALVULADOS

Esta seção contém uma breve revisão sobre as técnicas que já foram utilizadas na emulação de amplificadores valvulados. As técnicas aqui citadas não são, necessariamente, utilizadas neste trabalho. Visa-se uma boa contextualização do leitor, para que este compreenda a necessidade da utilização de técnicas como as redes neurais em uma situação tão específica.

² Fenômeno originado da combinação de harmônicas de um sinal sonoro em que estas possuem frequências de multiplicidade par, considerado na música um acorde estável. Seu sinônimo, a dissonância, representa um acorde instável. A explicação física presente nestes fenômenos foi definida em (HELMHOLTZ, 1895).

De acordo com Pakarinen e Yeh (2009), o próximo passo na evolução dos amplificadores seria simular a sonoridade destes utilizando computadores e processadores de sinais digitais (do inglês, *digital signal processors* - DSP's). O comportamento não linear dos amplificadores valvulados os caracterizam. Portanto, para emulá-los, há de se utilizar algoritmos capazes de representar tal característica.

O pré-amplificador valvulado pode ser dividido em duas partes principais: a parte linear e a parte não linear (OLIVEIRA; BARRETO; PASQUAL, 2013). Não há necessidade de identificar sistemas lineares utilizando técnicas de elevada complexidade, como redes neurais e séries de Volterra, assim como não é possível emular sistemas não lineares com base em técnicas muito simples ou limitadas.

Nesta seção são apresentadas técnicas de emulação e identificação de sistemas lineares e não lineares. Para o caso linear, é apresentada a técnica dos mínimos quadrados, visto que esta é muito adotada na literatura. Já para o caso não linear, são apresentadas diversas técnicas de emulação, além de sua aplicação para o caso de pré-amplificadores e amplificadores (completos) valvulados.

2.3.1 Mínimos quadrados

O *tone stack* do pré-amplificador consiste em um filtro composto por componentes passivos, com resposta linear. A linearidade desta parcela possibilita que técnicas simples sejam utilizadas para realizar sua identificação e emulação. Conforme Aguirre (2015), a aplicação da técnica dos mínimos quadrados (MQ) consiste em estimar o vetor de saída $\mathbf{y}$ no formato

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\theta}} + \mathbf{e}, \quad (2.1)$$

onde $\hat{\mathbf{y}}$ é a estimativa de $\mathbf{y}$, $\mathbf{X}$ é o vetor de entrada (ou de regressores), $\hat{\boldsymbol{\theta}}$ é um vetor de parâmetros estimado conforme

$$\hat{\boldsymbol{\theta}} = [\mathbf{X}^T \mathbf{X}]^{-1} \mathbf{X}^T \mathbf{y}, \quad (2.2)$$

e $\mathbf{e}$ é o erro ao tentar explicar o vetor de saída em função do vetor de regressores e do vetor de parâmetros.

O estimador MQ possui duas características principais: a boa robustez ao ruído, pois o MQ estima um conjunto de parâmetros a partir de um sistema sobre-determinado de equações, e a ortogonalidade entre os vetores $\hat{\mathbf{y}}$ e $\mathbf{e}$, resultando em uma diferença entre $\hat{\mathbf{y}}$ e $\mathbf{y}$ mínima (AGUIRRE, 2015).

2.3.2 Static waveshaping

A forma mais simples de gerar distorção não linear em um sinal de áudio se dá pela aplicação do sinal à uma função não linear estática³ (OLIVEIRA; BARRETO; PASQUAL, 2013).

³ Tal termo refere-se ao fato desta função ser um sistema sem memória i.e. a saída atual não depende de valores passados da mesma, nem da própria entrada.

A patente de Araya e Suyama (1996) apresenta três não linearidades, em que duas são simétricas e uma é assimétrica. Visto que não foram mostrados resultados no documento referentes à sua aplicação, Oliveira, Barreto e Pasqual (2013) analisaram as funções não lineares, concluindo que a não linearidade assimétrica era a única que continha componentes harmônicas de ordem par. Conforme esperado, as não linearidades simétricas somente apresentaram harmônicas ímpares.

Para que esta técnica apresente um resultado fidedigno quando utilizada para emular um pré-amplificador valvulado já existente, há a necessidade de se obter uma função não linear estática que esteja de acordo com a não linearidade das válvulas presentes nos circuitos. A dificuldade em se obter tal função é elevada, visto que são muitos os fatores físicos que alteram o funcionamento da válvula, além de que cada uma não é perfeitamente igual a outra, fazendo-se necessário a obtenção de várias funções não lineares.

2.3.3 Wave digital filters

Wave digital filters (WDF's) são tipos de filtros digitais capazes de simular o comportamento de um sistema real, em que os parâmetros dos filtros dependem das características físicas dos componentes simulados (KARJALAINEN; PAKARINEN, 2006).

Karjalainen e Pakarinen (2006) utiliza três métodos de implementação para um modelo de um pré-amplificador valvulado em WDF, sendo estes: iteração de equações, iterações com *look-up table*, e observação direta de uma tabela a-b⁴. A tabela a-b foi o método que utilizou o menor esforço computacional. As três formas resultaram em um sinal onde ambas harmônicas pares e ímpares estavam presentes.

Posteriormente, Pakarinen e Karjalainen (2010) obtiveram uma forma melhorada do modelo WDF de um estágio de ganho valvulado obtido em (KARJALAINEN; PAKARINEN, 2006). O novo modelo difere do anterior em abordar o estágio de ganho de uma forma mais detalhada, incluindo todos os componentes nele presentes com exceção das capacitâncias Miller, assim como o acoplamento entre a grade e o catodo, que antes fora omitido. De acordo com o autor, o modelo em WDF e a simulação no *LTSpice* resultaram em resultados próximos. Diferenças majoritárias começaram a surgir somente quando a distorção foi severa o bastante para que o efeito do *aliasing* começasse a aparecer na simulação em WDF, podendo ser atenuado utilizando frequências de amostragem mais elevadas.

O modelo apresentado em (PAKARINEN; KARJALAINEN, 2010) foi capaz de atingir uma simulação razoavelmente realista, incluindo os efeitos causados pela corrente grade-catodo. Também, foi percebido pelo autor que devido à natureza bidirecional dos WDF's, uma carga reativa afeta o comportamento do modelo da válvula, de forma similar aos circuitos valvulados reais.

⁴ *a* e *b* são parâmetros que definem os WDF's. Sua explicação detalhada está presente em (KARJALAINEN; PAKARINEN, 2006).

Dunkel et al. (2016) realizou a simulação de um pré-amplificador valvulado 5F6-A/JTM 45 presente no amplificador *Fender Bassman*. O autor aplicou o método de Newton com *back-tracking* para solucionar as equações não lineares do WDF de forma direta, não necessitando utilizar tabelas como nos métodos apresentados por Karjalainen e Pakarinen (2006). O modelo do triodo utilizado foi o de Dempwolf, semelhante ao modelo de Koren apresentado em (KOREN, 2003), porém com duas diferenças principais: o modelo de Dempwolf não possui descontinuidades e necessita de menor esforço computacional. Dunkel et al. (2016) conseguiu simular o pré-amplificador obtendo uma simulação muito próxima da simulação SPICE, porém, não sendo possível executá-la em tempo real por conta do poder de processamento requerida.

Conclui-se, portanto, que os WDF's necessitam de um esforço computacional considerável para que o algoritmo do modelo seja executado em tempo real. Tal capacidade de processamento não está presente nos DSP's mais simples, fato que torna a técnica economicamente inviável para a ideia proposta neste trabalho. A necessidade do conhecimento das características físicas das válvulas e de outros componentes também compõe uma dificuldade para a utilização de modelos em WDF.

2.3.4 Métodos analíticos

Métodos analíticos consistem em técnicas de obter o modelo de um sistema por meio de análises matemáticas e físicas, como por exemplo, equacionar um sistema eletrônico com base na análise do seu circuito. Como a emulação utilizando análises teóricas consiste em solucionar equações de forma numérica, as técnicas apresentadas também abordam formas de diminuir o esforço computacional necessário para se emular ou simular o sistema em tempo real.

2.3.4.1 Método K

Em (YEH; ABEL; SMITH, 2010) é apresentado o método K, este que abrange um algoritmo para resolução das equações diferenciais ordinárias (EDO's) não lineares em forma matricial. Este método foi aplicado pelo autor obtendo uma resposta satisfatória, se comparada à resposta do *software* LTSpice, principalmente se considerar que este gera o filtro para a não linearidade automaticamente, por meio do algoritmo proposto por Yeh, Abel e Smith (2010). A aplicação da técnica de forma tabelada necessita de menor esforço computacional, possibilitando implementações em tempo real mais estáveis e funcionais, porém, ainda há a necessidade de conhecimento das características físicas dos componentes presentes no circuito.

2.3.4.2 Modified blockwise

A simulação em *modified blockwise* consiste na separação do circuito por blocos. Este fato implica na diminuição da complexidade computacional e portanto, um *hardware* com menor capacidade de processamento é necessário para a utilização do modelo (MACAK; SCHIMMEL, 2010).

Para esta técnica, Macak e Schimmel (2010) utilizaram três métodos diferentes de aproximação, sendo que cada um deles apresenta uma diferente precisão do algoritmo, em relação aos resultados sem aproximação, e diferentes requerimentos de armazenamento. O primeiro método utilizado foi a interpolação linear, sendo o mais rápido para implementar, porém, com o custo de que vários valores sejam pré-computados para que se comporte de forma suave⁵. A segunda forma de aproximação utilizada em (MACAK; SCHIMMEL, 2010) foi a interpolação *spline*. Este método possui um comportamento suave desde que haja o armazenamento de valores pré-processados. Finalmente, a terceira técnica de aproximação consiste em uma variação da segunda, a aproximação *spline* cúbica. O comportamento suave desta se dá pelo armazenamento dos coeficientes dos polinômios, tendo este requerimento um custo de armazenamento de memória menor.

Em (MACAK; SCHIMMEL, 2011) é utilizado o método de *modified blockwise* para simular um pré-amplificador valvulado. É mostrado que as formas de interpolação linear e *spline* cúbica apresentaram o menor número de iterações na simulação. Ao comparar as componentes harmônicas do resultado obtido pela simulação com *spline* cúbica às harmônicas obtidas conforme o sinal de saída do pré-amplificador real, percebeu-se uma grande diferença na amplitude das harmônicas pares, implicando em uma simulação não satisfatória.

Além de os resultados obtidos na literatura para esta técnica não serem satisfatórios, a representação de componentes reais utilizando modelos matemáticos pode ocasionar em imprecisão, devido à dificuldade de se obter o modelo ou desconhecimento físico sobre seu funcionamento.

2.3.5 Técnicas baseadas em convolução

A convolução linear de um sinal com a resposta ao impulso de um sistema pode ser uma boa representação do comportamento do mesmo desde que este seja linear e invariante no tempo (FARINA; ARMELLONI, 2005). Caso essas necessidades não sejam satisfeitas, a resposta obtida pode não ser satisfatória. Naturalmente, algumas variações das técnicas básicas de convolução para sistemas lineares surgiram com a intenção de emular sistemas não lineares. Três delas são apresentadas na sequência.

2.3.5.1 Resposta ao impulso variável

De acordo com Farina e Armelloni (2005), este método é simples e consiste em modificar a resposta ao impulso (RI) conforme a amplitude instantânea do sinal de entrada. Esta mudança é feita por meio de um algoritmo que muda o valor da RI com base em uma função logarítmica do valor da amostra.

⁵ O termo suave, neste contexto, refere-se à convergência do método de aproximação utilizado. Caso este não se comporte de forma suave, há maior probabilidade de ocorrer divergências numéricas.

Esta técnica foi aplicada por Farina e Armelloni (2005), utilizando um sinal sonoro com boa dinâmica⁶ (uma música do artista Elton John), e aplicando-a a um efeito de distorção para guitarra elétrica altamente não linear.

De acordo com o autor, o sinal obtido pela simulação do efeito de distorção foi analisado qualitativamente e teve uma sonoridade semelhante ao sinal obtido conforme o efeito de distorção real. Análises quantitativas não foram realizadas, assim como a simulação não foi executada em tempo real.

2.3.5.2 Núcleos de Volterra

De acordo com Aguirre (2015), a saída de um sistema não linear pode ser representada pela série de Volterra. O número de parâmetros a serem determinados pela série é muito grande, visto que esta tenta explicar a saída somente em função da entrada.

Farina e Armelloni (2005) aplicaram o método da mesma forma que foi apresentado na sub-seção 2.3.5.1. O resultado obtido foi analisado qualitativamente e teve uma sonoridade diferente do sinal aplicado ao efeito de distorção real. Nenhuma análise quantitativa foi realizada.

2.3.5.3 Núcleos de Hammerstein excitado por senoide de fase variável

O modelo de Hammerstein *kernels* (Núcleos de hammerstein) consiste em uma forma simplificada das séries de Volterra (AGUIRRE, 2015). Schmitz e Embrechts (2017) sugerem a utilização da mesma com um sinal de excitação senoidal, para uma fase que varia de forma exponencial.

Schmitz e Embrechts (2017) aplicam o método em dois amplificadores valvulados, o TSA15h da *Ibanez* e o Retro Tube 50 E762 da *Engl*. Para o caso do TSA15h, o autor compara a amplitude das componentes harmônicas do sinal obtido via emulação com o sinal real, percebendo que a resposta é muito próxima para as harmônicas de menor ordem, indiferentemente de estas serem pares ou ímpares. O erro observado entre a amplitude das harmônicas foi diretamente proporcional à ordem das mesmas. Para o amplificador da *Engl*, que possui uma não linearidade mais severa que o anterior devido ao sinal de saída ser altamente assimétrico, o número de núcleos de Hammerstein foi aumentado. O erro médio encontrado na emulação deste amplificador foi de 2,3 %.

O método apresentado em (SCHMITZ; EMBRECHTS, 2017) mostrou-se válido para a emulação de amplificadores valvulados, porém, percebe-se que para não linearidades severas é necessário aumentar o número de núcleos de Hammerstein, fato que ocasiona em uma maior dificuldade de execução da emulação em tempo real, visto que o esforço computacional necessário é maior.

⁶ Boa dinâmica refere-se à presença de várias frequências, em que todas podem ser relativamente bem distinguidas ao se ouvir. A compressão de frequências sonoras, técnica muito utilizada na edição de áudio atualmente, é um fator inversamente proporcional à dinâmica sonora.

2.3.6 Redes neurais artificiais

Redes neurais artificiais (RNAs) consistem em algoritmos inspirados pelo sistema neuronal presente no cérebro de mamíferos, pois este é altamente complexo, não linear e capaz de processar informação de forma paralela. As unidades de processamento das RNAs são os neurônios, que podem ser dispostos em uma ou mais camadas.

Em (HORNIK; STINCHCOMBE; WHITE, 1989), é mostrado que redes neurais multi-camadas *feedforward*⁷ são aproximadores universais de funções, ou seja, são teoricamente capazes de emular/identificar qualquer tipo de função. Diante disso, pesquisadores começaram a utilizar redes neurais como uma técnica de identificação de sistemas. As RNAs possuem uma característica de processamento não linear, o que é conveniente no âmbito dos amplificadores valvulados, pois a não linearidade é uma característica relevante de tais sistemas. Existem diversos tipos de redes neurais. Os que mais foram aplicados na emulação de amplificadores valvulados são abordados nesta seção, enquanto que uma abordagem mais profunda sobre o tema é realizada no Capítulo 4.

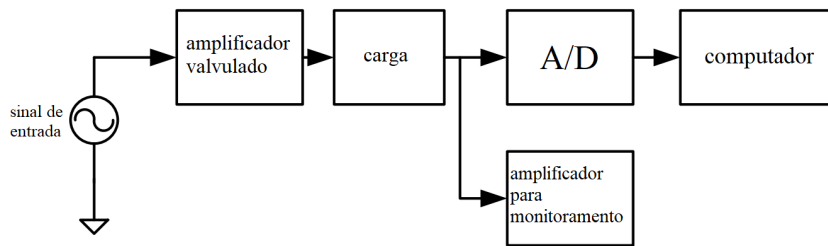
2.3.6.1 Redes neurais recorrentes

Uma rede neural recorrente (RNR) consiste em uma rede neural artificial com memória de curto prazo. Esta memória é implementada por meio de realimentação e atraso, o que é necessário para que a rede seja capaz de emular comportamento dinâmico. Um modelo NARX (do inglês, *nonlinear autoregressive network with exogenous inputs*) de uma rede neural recorrente consiste em uma RNR com conectividade limitada, proporcionando um funcionamento semelhante à sua originária, porém com menor necessidade de esforço computacional. Este modelo é melhor abordado na Subseção 3.1.4.

Covert e Livingston (2013) utilizaram um modelo NARX da RNR de duas camadas, *feedforward* para identificar um pré-amplificador valvulado, com número variável de neurônios não lineares na camada oculta e um neurônio linear na camada de saída. O formato do sistema para treino e teste da rede neural é apresentado na Figura 10.

⁷ *Feedforward*, do inglês, alimentada à frente, é um termo utilizado para definir redes neurais artificiais em que a informação é sempre processada da entrada para a saída, nunca na direção contrária.

Figura 10 – Diagrama de blocos do sistema de treino e teste da rede neural.



Fonte: adaptado de (COVERT; LIVINGSTON, 2013).

A aquisição de dados foi realizada utilizando um sinal senoidal com frequência entre 100 e 500 Hz na entrada de um amplificador valvulado de 4 W da *Vox Amplification* modelo AC4TV, associado à uma carga RLC - circuito para modelo de um auto-falante de 16 Ω . Para digitalizar o sinal, foi utilizada uma interface de áudio em uma frequência de amostragem de 96 kHz, com quantização em 24 bits.

Testou-se os sinais em algumas redes neurais com configurações diferentes, em que o número de neurônios e atrasos foi determinado de forma empírica, para o sinal com pouca distorção e muita distorção. Os valores obtidos dos erros ficaram muito altos, segundo o autor, principalmente para sinais com bastante distorção. O autor justifica o erro com base em possíveis problemas de treinamento, sendo insuficiente para emular o sistema desejado, além de muitos atrasos, parâmetros e neurônios.

É possível notar que a rede neural proposta em (COVERT; LIVINGSTON, 2013) foi elaborada, em grande parte, empiricamente. A deficiência do artigo em um estudo de sinal de excitação, número de neurônios, de camadas, e técnicas de aprendizagem pode ter sido um dos fatores que refletiu em valores de RMSE elevados, assim como uma deficiência na técnica de aprendizagem ou na própria topologia da rede neural.

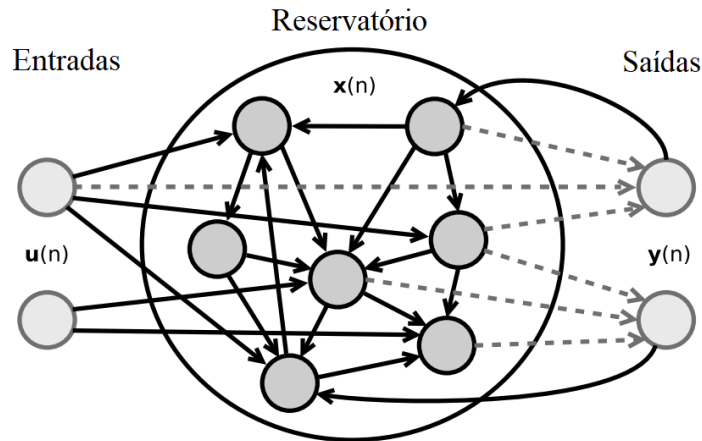
2.3.6.2 Computação reservatória

Uma forma diferenciada de lidar com redes neurais recorrentes é a de simplificá-las fixando um sistema dinâmico não linear, chamado de sistema reservatório ou líquido, em uma única camada oculta formada por neurônios recorrentes conectados de forma aleatória. A grande vantagem da computação reservatória (ou *reservoir computing*) é que os neurônios da rede neural presente na camada oculta podem ser treinados por regressão linear, garantindo que o treinamento não fique preso em mínimos locais⁸. Reservatórios computacionais são inerentemente temporais, fato que os torna candidatos para o processamento de áudio.

⁸ Mínimos são pontos de menor energia de funções de custo. O treinamento de uma rede neural é feito com base na minimização da função de custo que a descreve, objetivando atingir um mínimo global. Mínimos locais, como o nome já diz, são mínimos somente para uma pequena parcela da função custo, enquanto um mínimo global é o mínimo de energia absoluto.

A estrutura da rede de computação reservatória proposta por Holzmann (2009) é dada conforme a Figura 11, onde $\mathbf{u}(n)$ são os sinais de entrada e $\mathbf{y}(n)$, os sinais de saída. Percebe-se a presença de dois neurônios de entrada, dois neurônios de saída e diversos neurônios no reservatório computacional.

Figura 11 – Esquemática de uma rede de *reservoir computing*.



Fonte: adaptado de (HOLZMANN, 2009).

Os sinais de excitação utilizados para sistemas lineares são impulsos, senoides com frequência variável, ruído e sequências binárias pseudo-randômicas, que costumam resultar em um desempenho satisfatório. Para sistemas não lineares, todas as frequências e todas as amplitudes possíveis devem estar presentes no sinal de excitação. Holzmann (2009) observa empiricamente que o sinal para a excitação de áudio não linear que obteve menor erro fora o ruído branco com variação linear de amplitude e um filtro passa-baixas adicional.

Em (HOLZMANN, 2009), o autor aplica a técnica de computação reservatória na emulação da distorção produzida por válvulas. Primeiro, foi aplicada uma gravação do som de uma flauta a um modelo de distorção chamado *Valve Saturation*. Posteriormente, aplicou-se um sinal de entrada com vários instrumentos e um cantor em um modelo de distorção similar. O autor aplicou os mesmos sinais aos mesmos sistemas por meio de séries de Volterra. Os resultados obtidos mostram que a *reservoir computing*, em comparação às séries de Volterra, apresenta resultados quantitativos superiores em todos os testes realizados.

Posteriormente, Keuninckx, Danckaert e Sande (2017) propõem uma rede com vários reservatórios computacionais em cascata, cada um contendo um único neurônio, evitando as dificuldades de se construir uma rede neural recorrente com vários nós não lineares. Esta técnica é atrativa por requisitar de um esforço computacional muito pequeno e ser de fácil implementação em *hardware*. O método foi aplicado somente para amplificadores de áudio com característica linear, resultando em um bom resultado quantitativo.

2.3.6.3 Long short-term memory

A rede neural recorrente é incapaz de aprender as conexões entre dois neurônios que estão longe um do outro, pois o gradiente do erro - grandeza utilizada para atualizar o valor dos pesos sinápticos - é comumente reduzido a um valor muito pequeno. Este problema é conhecido como *vanishing gradient* e é comum em redes neurais profundas. Para evitar este problema, são utilizadas células de memória que permitam a propagação de termos a longo prazo sem que estes desapareçam, formando a rede conhecida como *long short term memory* (LSTM). Tal topologia de rede neural é exposta em detalhes na Seção 4.5.

Schmitz e Embrechts (2018) apresentam uma proposta de RNR com LSTM dedicada à proposta de emular amplificadores valvulados. O processo de aprendizagem utilizado em (SCHMITZ; EMBRECHTS, 2018) pode ser dividido nas seguintes etapas: escolha de formatação de dados, elaboração da rede neural, execução da fase de aprendizagem e uso do modelo obtido para emulação. Para a obtenção do conjunto de dados, o autor utiliza um sinal composto de dois sinais provindos de uma guitarra elétrica: algumas notas simples e alguns acordes⁹. Empiricamente, foi proposto por Schmitz e Embrechts (2018) que um conjunto de dados compondo um áudio com duração de vinte segundos é longo o bastante para se obter bons resultados. Os conjuntos de dados obtidos foram divididos em três partes, um conjunto de treinamento, um conjunto de teste e um conjunto de validação.

O autor emulou o amplificador valvulado *Engl Retro 50* com ganho máximo (altamente não linear) utilizando sua RNR com LSTM, obtendo um erro médio quadrático muito pequeno. Schmitz e Embrechts (2018) aplicaram também os mesmos sinais, por questões comparativas, em uma rede neural profunda de seis camadas com 512 neurônios cada, obtendo erros aproximadamente vinte vezes superiores aos obtidos na LSTM. Os valores obtidos pelo autor comprovam a eficiência do uso da LSTM em comparação à outras topologias de redes neurais apresentadas, porém, esta rede demanda de maior processamento computacional devido a sua complexidade.

2.4 CONSIDERAÇÕES FINAIS DO CAPÍTULO

Foi observada a dificuldade presente na emulação de um pré-amplificador (ou amplificador) valvulado devido ao sistema ser, na maioria dos casos, altamente não linear, resultando em um sinal de saída assimétrico e, portanto, com uma quantidade considerável de harmônicas de ordem par. As técnicas utilizadas para a identificação de sistemas não lineares são complexas e normalmente necessitam de um poder de processamento considerável e, ainda assim, não necessariamente obtendo uma emulação satisfatória. As redes neurais, por outro lado, demonstram-se promissoras no quesito de emulação em tempo real, pois, após treinadas, podem ser executadas de forma *offline* possibilitando uma execução com pouco atraso temporal. Da mesma forma, a capacidade de generalização de uma rede neural mostra-se superior a de técnicas comuns de identificação de sistemas não lineares.

⁹ Um acorde consiste em várias notas sendo tocadas simultaneamente.

No entanto, a topologia de rede neural que apresentou melhores resultados foi a *long short-term memory*. Tal rede neural demanda um grande processamento computacional, mesmo se executado somente o modelo obtido após o treinamento. Percebe-se que na utilização de outras topologias menos complexas de redes neurais - como a rede neural recorrente simples - não houve um estudo aprofundado de sua estrutura e de possíveis sinais para serem utilizados na identificação. Tais estudos são realizados neste trabalho, quanto aos sinais de excitação (Capítulo 3) e quanto à estrutura da rede neural recorrente (Capítulo 6).

3 IDENTIFICAÇÃO DE SISTEMAS

A obtenção do modelo de um sistema baseado na sua observação e no estudo de suas propriedades, seja esse modelo uma hipótese, uma lei da física, um paradigma, entre outros, é uma tentativa de ligar observações a um certo padrão, sendo esta ideia a base motivacional da ciência (LJUNG, 1986).

A motivação para o estudo de técnicas de identificação de sistemas é originada pelo fato de não se conhecer, em muitos casos, as equações que envolvem um modelo, ou que estas sejam conhecidas porém impraticáveis por limitações de tempo e recursos (AGUIRRE, 2015). As técnicas de identificação lidam, portanto, com o problema de elaborar modelos matemáticos de sistemas dinâmicos com base em informações e dados observados do sistema.

Para obter os dados que serão utilizados na identificação de um sistema, deve-se excitá-lo com um sinal de entrada, denominado sinal de excitação. O sinal aplicado ao sistema e a resposta do mesmo a este sinal (saída) devem ser coletados de alguma forma. A identificação pode ocorrer de forma *online* - realizando a obtenção de dados e elaboração do modelo em tempo real - ou *offline* - o sistema é excitado e os dados são obtidos anteriormente à aplicação da técnica de identificação para obtenção do modelo matemático.

Neste capítulo, são abordados alguns dos principais conceitos para a compreensão da teoria de identificação de sistemas que baseia este trabalho: tipos de modelo matemático para sistemas lineares, para sistemas não lineares e sinais de excitação. Visto que o objetivo deste trabalho é identificar um sistema, este capítulo objetiva, principalmente, situar o leitor no conceito da identificação de sistemas, assim como na parte mais básica desta teoria. Essencialmente, a Seção 3.2 será utilizada de forma direta neste trabalho, enquanto que as seções anteriores expõem a estrutura dos modelos comumente utilizados na identificação.

3.1 ESTRUTURA DOS MODELOS

A modelagem de sistemas dinâmicos lineares e não lineares diferencia-se pela complexidade do modelo. Alguns modelos só podem ser utilizados para sistemas com características restritas, enquanto que outros abrangem uma variedade considerável. Os principais modelos para sistemas dinâmicos lineares e não lineares são definidos a seguir.

3.1.1 Modelo auto-regressivo

O modelo auto-regressivo (AR) é representado por uma expressão autônoma, conforme

$$y(k) = a_1 y(k-1) + a_2 y(k-2) + \dots + a_{n_y} y(k-n_y), \quad (3.1)$$

sendo $y(k)$ representa a saída atual de um sistema, $y(k-n)$ são as saídas passadas de n amostras anteriores, a_n são os coeficientes do modelo e n_y é o maior atraso em y . Percebe-se que este

modelo não inclui nenhuma entrada ao sistema, por isso a expressão (3.1) é considerada autônoma - não depende de um sinal de entrada.

3.1.2 Modelo auto-regressivo com entrada exógena

Incluindo-se uma entrada na expressão (3.1), obtém-se a seguinte representação matemática:

$$y(k) = a_1 y(k-1) + \dots + a_{n_y} y(k-n_y) + b_1 u(k-1) + \dots + b_{n_u} u(k-n_u), \quad (3.2)$$

considerando u é o vetor de entrada, b são os coeficientes do modelo para a entrada e n_u é o maior atraso em u . Como há a inclusão do sinal de entrada no modelo, pode-se representar diversos sistemas de forma mais adequada que no caso dos modelos AR.

Ambos os modelos AR e ARX são simplificações de uma representação mais complexa, denominada de modelo auto-regressivo com média móvel e entradas exógenas, que é apresentada na próxima subseção, pois não levam em conta a influência do ruído e/ou de características não modeladas do sistema.

3.1.3 Modelo auto-regressivo com média móvel e entradas exógenas

Levando-se em conta o ruído, pode-se representar o sistema a partir de um modelo geral em tempo discreto (AGUIRRE, 2015)

$$y(k) = \frac{B(q)}{F(q) A(q)} u(k) + \frac{C(q)}{D(q) A(q)} v(k), \quad (3.3)$$

em que $v(k)$ é o ruído branco, q é o operador temporal de deslocamento em tempo discreto de forma que $y(k) q = y(k+1)$ e $A(q)$, $B(q)$, $C(q)$, $D(q)$ e $F(q)$ são polinômios descritos conforme:

$$A(q) = 1 - a_1 q^{-1} - \dots - a_{n_y} q^{-n_y}; \quad (3.4a)$$

$$B(q) = b_1 q^{-1} + \dots + b_{n_u} q^{-n_u}; \quad (3.4b)$$

$$C(q) = 1 + c_1 q^{-1} + \dots + c_{n_v} q^{-n_v}; \quad (3.4c)$$

$$D(q) = 1 + d_1 q^{-1} + \dots + d_{n_d} q^{-n_d}; \quad (3.4d)$$

$$F(q) = 1 + f_1 q^{-1} + \dots + f_{n_f} q^{-n_f}, \quad (3.4e)$$

sendo a_n , b_n , c_n , d_n e f_n são constantes e n_v , n_d e n_f são os maiores atrasos em v , d e f , respectivamente. Tomando-se $D(q) = F(q) = 1$ e $A(q)$, $B(q)$ e $C(q)$ polinômios arbitrários, obtém-se o modelo ARMAX:

$$y(k) = \frac{H(q)}{A(q)} u(k) + \frac{C(q)}{A(q)} v(k). \quad (3.5)$$

O modelo ARMAX, assim como o modelo ARX pertence à classe de modelos de erro na equação. No presente caso, o erro na equação é modelado como um processo de média móvel

(MA), e o ruído adicionado à saída

$$\frac{C(q)}{A(q)} v(k) \quad (3.6)$$

em geral é modelado como ruído branco filtrado pelo filtro ARMA, $C(q)/A(q)$.

Tais modelos, porém, representam um sistema dinâmico linear ou, alternativamente, um sistema dinâmico não linear em torno de um único ponto de operação, onde se pode aproximar por uma relação linear. A técnica de identificação mais utilizada para se obter os parâmetros dos modelos AR e ARX é o mínimos quadrados (Subseção 2.3.1). Alguns modelos não lineares são apresentados nas próximas subseções.

3.1.4 Modelo NARX e NARMAX

O modelo NARX (do inglês, *nonlinear autoregressive model with exogenous variables*) explica o valor da saída $y(k)$ de forma discreta no tempo em função de valores passados dos sinais de entrada e saída, como:

$$y(k) = F[y(k-1), \dots, y(k-n_y), u(k-\tau_d), \dots, u(k-n_u)], \quad (3.7)$$

sendo τ_d o tempo morto e F uma função não linear.

Já o modelo NARMAX consiste em um modelo NARX de um sistema com a inclusão de termos de ruído e é normalmente representada como

$$y(k) = F[y(k-1), \dots, y(k-n_y), u(k-\tau_d), \dots, u(k-n_u), e(k), e(k-1), \dots, e(k-n_e)], \quad (3.8)$$

em que $e(k)$ é o ruído e n_e é o maior atraso no modelo de ruído. Tal representação possui formas polinomiais e racionais, conforme é mostrado em (AGUIRRE, 2015).

As redes neurais também compreendem uma forma de modelar sistemas dinâmicos. Sua estrutura, no entanto, é muito complexa para ser expressa de forma semelhante às expressões apresentadas neste capítulo. A rede treinada é o próprio modelo que será utilizado para representar o sistema, sendo esta muito semelhante ao modelo NARX, devido à seu comportamento não linear. As redes neurais artificiais são abordadas em detalhes no Capítulo 4.

3.2 SINAIS DE EXCITAÇÃO

Características dinâmicas e estáticas de um sistema que não forem excitadas, não estarão presentes nos dados (AGUIRRE, 2015). Portanto, há de se escolher um sinal de entrada que seja capaz de excitar todas as características importantes do sistema. Para um sistema não linear, por exemplo, o sinal deve excitar o sistema como um todo, não somente em torno de um ponto de operação, sendo assim capaz de excitar as não linearidades. Ao mesmo tempo, deve-se excitar todas as frequências de interesse. Sinais capazes de tal tarefa são determinados persistentemente excitantes (PEs). Um sinal é persistentemente excitante se sua média e função de autocorrelação (FAC) existirem, e se sua matriz de covariância for não singular (AGUIRRE,

2015). Para identificar P parâmetros, o sinal de excitação deve ser PE de ordem P (LJUNG, 1986). Isso significa que o espectro/módulo do sinal na frequência deve ter ao menos P componentes não nulas. Um ruído branco ¹, por exemplo, é um sinal persistentemente excitante de ordem infinita. A FAC do ruído branco é não nula apenas para o instante atual, ou seja, o sinal atual é descorrelacionado com sinais passados.

Em (SCHMITZ; EMBRECHTS, 2017), é utilizada uma senoide de frequência variável (*sine sweep*) para a excitação de amplificadores valvulados ao realizar sua emulação por meio da técnica de *Hammerstein Kernels*. O sinal senoidal deve variar da menor até a maior frequência de interesse. A ordem de persistência de excitação de um *sine sweep* é igual ao número de frequências que este excita.

Um sinal de guitarra elétrica com diversas notas, acordes e diferentes técnicas de tocar foi utilizado como sinal de excitação em (SCHMITZ; EMBRECHTS, 2018) na identificação de amplificadores valvulados utilizando redes neurais *long short-term memory*. Diferentemente do *sine sweep*, o sinal provindo de uma guitarra elétrica possui menor ordem de persistência de excitação, por não abranger tantas frequências, porém, dependendo da intensidade que as notas forem tocadas o módulo do sinal será diferente e assim, excitará características do sistema que dependem do módulo do sinal de entrada, fato que é importante no caso de um sistema não linear. Uma forma de se obter a característica de variação do módulo do sinal de entrada de um sistema utilizando *sine sweeps*, que só variam na frequência, é utilizar um sinal composto de vários *sine sweeps* com amplitudes diferentes.

¹ Este sinal pode ser obtido por meio das funções *rand* e/ou *randn* do software Matlab ®

4 REDES NEURAIS ARTIFICIAIS

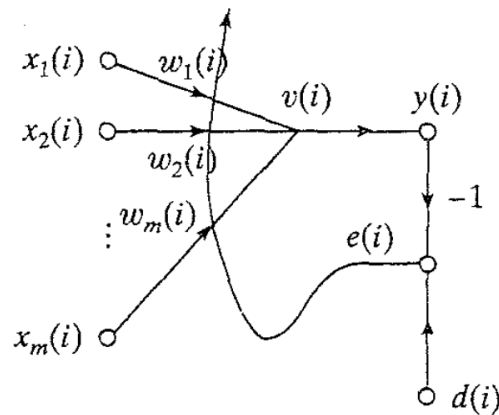
As redes neurais artificiais, conforme já introduzidas na Subseção 2.3.6, baseiam-se em sistemas neuronais (biológicos) para realizar funções relacionadas a processamento de dados ou imagem. Tais redes são abordadas nas mais diversas áreas da ciência, desde a simulação de sistemas até a classificação de padrões e processamento de imagem.

A utilização de redes neurais no âmbito da identificação de sistemas tem como motivação o fato de não ser necessário um conhecimento considerável sobre os processos físicos que ocorrem no sistema. Uma válvula é um componente bastante complexo e sua modelagem, conforme diversas obras da literatura abordadas no Capítulo 2, não é uma tarefa simples de ser realizada. Ao utilizar redes neurais, têm-se a possibilidade de obter um modelo fidedigno ao sistema necessitando basicamente de dados de entrada e saída, com base nos sinais de excitação expostos na Seção 3.2.

As redes neurais recorrentes (RNRs) são originadas de redes *multilayer perceptron* e possuem diferentes topologias cujas principais - tendo em vista o foco de identificação de sistemas - são abordadas neste capítulo. Também são vistas algumas das principais técnicas de treinamento das redes neurais *feedforward* e das RNRs, assim como seus principais problemas e possíveis soluções, visando a emulação de um sistema dinâmico altamente não linear - o pré-amplificador valvulado.

4.1 O NEURÔNIO ARTIFICIAL

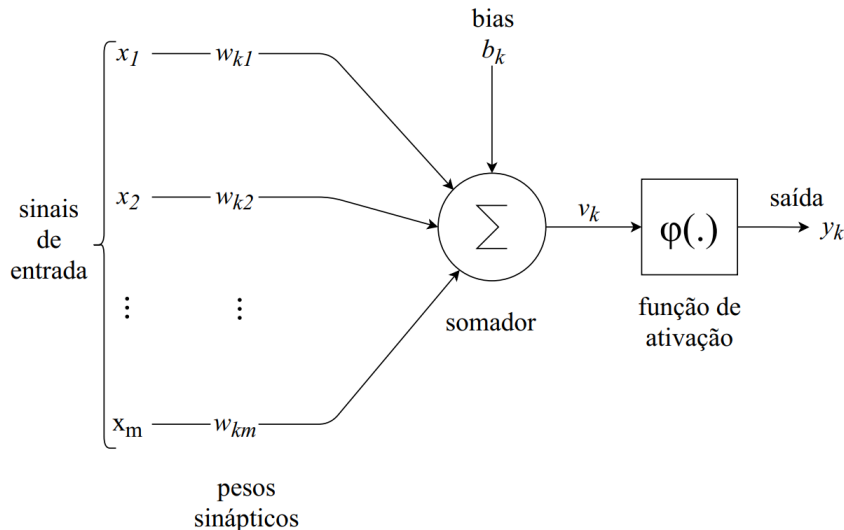
O *perceptron* foi apresentado em (ROSENBLATT, 1958) como um sistema nervoso hipotético, desenvolvido com o objetivo de ilustrar algumas propriedades fundamentais de sistemas inteligentes. Este modelo é descrito por Rosenblatt (1958) como um filtro adaptativo e exemplificado em tarefas de identificação de sistemas. De acordo com Haykin (1999), sua operação consiste em dois processos contínuos: filtragem e adaptação. A filtragem produz uma saída de acordo com um vetor de entrada e um sinal de erro, obtido comparando-se o valor da saída obtida e o valor da saída desejada. A adaptação consiste no ajuste dos pesos sinápticos do neurônio de acordo com o valor do erro. O modelo de um *perceptron* pode ser observado conforme a Figura 12.

Figura 12 – Modelo de um *perceptron*.

Fonte: (HAYKIN, 1999).

O *perceptron*, posteriormente, viria a originar o conceito de um neurônio. A diferença básica entre um neurônio e um *perceptron* é que este último não possui função de ativação, ou seja, sua saída consiste simplesmente no somatório das entradas ponderadas pelos pesos sinápticos. O modelo de um neurônio pode ser observado conforme a Figura 13.

Figura 13 – Modelo de um neurônio.



Fonte: adaptado de (HAYKIN, 1999).

Atribui-se ao neurônio uma função de ativação ϕ , sendo esta linear ou não linear, que será responsável por representar seu comportamento como processador de dados no sistema. Os sinais de entrada x do neurônio serão impostos a este de forma ponderada, com base nos pesos sinápticos w - valores escalares que multiplicarão o valor de cada entrada. O *bias* é um valor escalar aplicado diretamente ao neurônio que permite uma melhor adaptação da rede neural, aumentando ou diminuindo a entrada de uma função de ativação conforme a necessidade da

rede.

O campo local induzido v pode ser escrito conforme

$$v_k(i) = \sum_{n=1}^m w_{kn}(i) x_n(i) + b_k, \quad (4.1)$$

onde $w_{kn}(i)$ representa os m pesos sinápticos do neurônio de índice k e $x_n(i)$ são os sinais de entrada, ambos medidos no instante i . b_k é o valor do *bias* do neurônio, podendo este ser ponderado por um peso sináptico extra w_{kb} ou não. O treinamento adaptará o valor de outros pesos sinápticos para compensar o *bias* caso este não possua seu próprio peso.

A saída y do neurônio k , em um instante i , consiste, portanto, na aplicação do campo local induzido v à uma função de ativação φ , conforme a relação

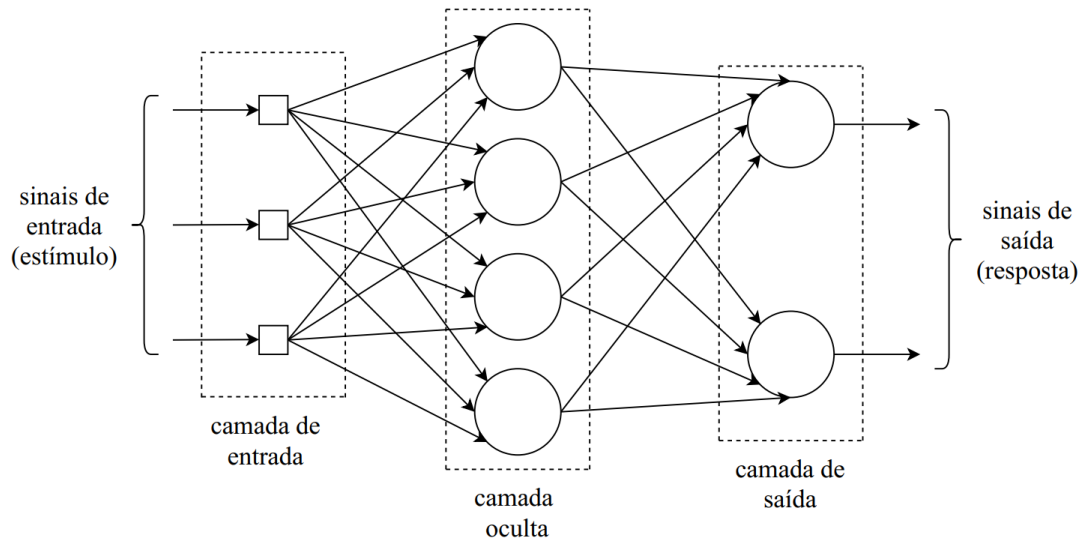
$$y_k(i) = \varphi(v_k(i)) = \varphi \left(\sum_{n=1}^m w_{kn}(i) x_n(i) + b_k \right). \quad (4.2)$$

4.2 MULTILAYER PERCEPTRON

De acordo com Haykin (1999), o *multilayer perceptron* (MLP) difere-se do *perceptron* em três pontos principais:

- o modelo de cada neurônio no *multilayer perceptron* inclui uma função de ativação não linear;
- a rede contém uma ou mais camadas de neurônios ocultos que não fazem parte da entrada ou saída;
- a rede possui um alto grau de conectividade, determinado pelas sinapses (pesos).

A Figura 14 mostra um exemplo de uma rede *multilayer perceptron* com três camadas, sendo elas a camada de entrada, camada oculta e camada de saída, contendo três entradas, quatro e dois neurônios, respectivamente. As expressões que descrevem o campo local induzido e a saída dos neurônios são iguais às apresentadas em (4.1) e (4.2), porém, para o caso de uma camada posterior à primeira camada oculta, as entradas x_n são as saídas dos neurônios da camada anterior.

Figura 14 – Exemplo de uma rede *multilayer perceptron*.

Fonte: o próprio autor, 2018.

4.3 REDES NEURAIIS RECORRENTES

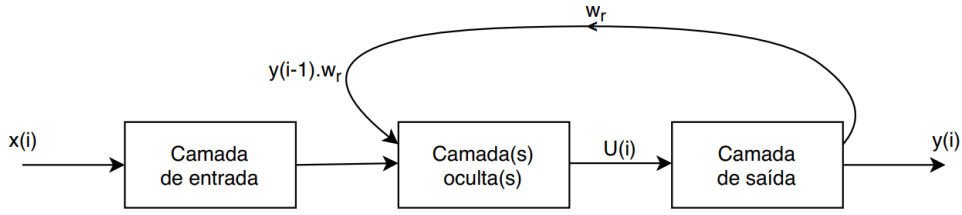
Redes neurais recorrentes (RNR) consistem em redes neurais artificiais com uma ou mais realimentações (HAYKIN, 1999). Essas realimentações podem ser locais - realimentação de um neurônio a ele mesmo - ou globais - de uma camada à outra.

De acordo com Haykin (1999), existem duas formas principais de utilização das redes neurais recorrentes: memória associativa e mapeamento entrada-saída. As realimentações possibilitam as redes neurais de obter representações de estado, devido ao seu comportamento temporal, tornando assim as redes neurais recorrentes boas opções para predição e modelagem de sistemas dinâmicos não lineares.

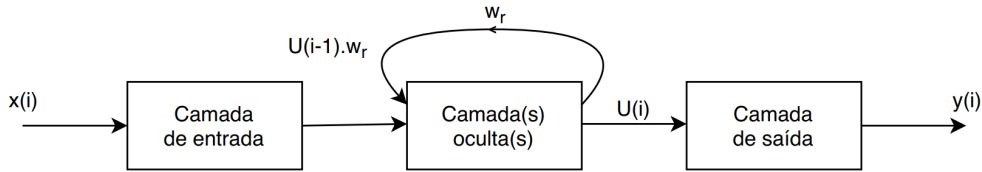
4.3.1 Redes de Jordan e Elman

Dentre as topologias das redes neurais recorrentes para aplicação em identificação de sistemas, duas são amplamente utilizadas por sua simplicidade: as redes de Jordan e de Elman. Estas são comumente referidas na literatura como RNRs *vanilla*, termo utilizado para definir algo padrão/simples. Tais topologias estão representadas na Figura 15.

Figura 15 – Redes neurais recorrentes de Jordan e Elman.



(a) Rede neural recorrente de Jordan



(b) Rede neural recorrente de Elman

Fonte: o próprio autor, 2019.

A topologia de Jordan foi apresentada em (JORDAN, 1986) e consiste em realimentações da saída da rede neural para a(s) camada(s) oculta(s). Para este caso, o campo local induzido de um neurônio oculto diferirá do que foi apresentado em (4.1) por um termo extra, de acordo com

$$v(i) = \sum_{k=1}^m [w_k(i) x_k(i)] + b w_b(i) + y(i-1) w_r(i), \quad (4.3)$$

sendo $y(i-1)$ é o valor da saída da rede neural no instante $i-1$ e $w_r(i)$ representa o peso sináptico da realimentação no instante i .

A proposta apresentada por Elman (1990) visa representar as dependências temporais do sistema por meio de realimentações internas. Assim, uma pequena modificação há de ser feita da relação (4.3), obtendo

$$v(i) = \sum_{k=1}^m [w_k(i) x_k(i)] + b w_b(i) + U(i-1) w_r(i), \quad (4.4)$$

em que $U(i-1)$ é a saída do neurônio realimentado no instante $i-1$.

A simplicidade de tais topologias possibilita aplicações em situações onde não haja muito poder de processamento ou, até mesmo, em tempo real. O maior problema com tais topologias de redes neurais recorrentes, no entanto, reside no treinamento. De acordo com Glorot e Bengio (2010), o maior problema das RNRs é sua incapacidade de aprender informações que possuam dependências temporais de longo prazo. Tal incapacidade é conhecida como *vanishing gradient* e é discutida em detalhes na Seção 4.4 deste trabalho.

4.4 TÉCNICAS DE APRENDIZADO DE REDES NEURAIIS RECORRENTES

As técnicas de aprendizado utilizadas em redes neurais recorrentes baseiam-se, diretamente ou indiretamente, no gradiente do valor do erro. O valor deste gradiente irá determinar quanto um parâmetro será alterado. Usualmente, os parâmetros livres durante o treinamento são os pesos sinápticos.

As técnicas que se baseiam de forma pura no gradiente do erro para atualização dos pesos sinápticos são o *back-propagation through time* e o *real-time recurrent learning*. Neste trabalho, só será discutida a técnica do *back-propagation through time*, devido a esta ser mais comumente utilizada na literatura atual, utilizar menos memória durante sua execução e ser mais facilmente utilizada como cálculo de derivadas durante o treinamento por filtro de Kalman estendido quando a rede neural recorrente possui mais de uma camada. Para a compreensão do aprendizado em RNRs, deve-se antes entender a forma que este é feito em redes neurais *feedforward multilayer perceptron*.

4.4.1 Back-propagation

O algoritmo de *back-propagation* é a forma mais comum e simples de realizar o treinamento de uma rede neural MLP, consistindo em propagar o sinal de erro da saída para a entrada da rede, ajustando o valor dos pesos sinápticos de acordo com o valor do gradiente do erro em cada neurônio.

Conforme (HAYKIN, 1999), considerando o valor do erro e de um neurônio j de saída da rede neural na amostra i , tem-se

$$e_j(i) = d_j(i) - y_j(i), \quad (4.5)$$

onde d é o sinal de saída desejado e y é o sinal de saída obtido pela rede.

O valor instantâneo da energia do erro do neurônio j é definido como $\frac{1}{2}e_j^2(n)$, sendo o valor instantâneo total $\mathcal{E}(n)$ a soma da energia do erro sobre todos os neurônios de saída, da forma

$$\mathcal{E}(n) = \frac{1}{2} \sum_{j \in C} e_j^2(n), \quad (4.6)$$

onde o conjunto C inclui todos os neurônios da camada de saída. A energia do erro médio quadrático pode ser obtida somando-se $\mathcal{E}(n)$ para todo i , com i variando de 1 a N , onde N é número total de amostras do sinal de saída, conforme

$$\mathcal{E}_{med.} = \frac{1}{N} \sum_{i=1}^N \mathcal{E}(n). \quad (4.7)$$

O objetivo principal do treinamento é reduzir a função custo $\mathcal{E}_{med.}$, o que é feito minimizando $\mathcal{E}(n)$ para cada amostra.

O ajuste de pesos é feito amostra por amostra, até que uma época - passagem por todas as amostras do conjunto de dados de treinamento - seja atingida. Ao atingir uma época, o erro médio

quadrático é medido. Caso este não seja inferior ao erro médio quadrático máximo, definido previamente, repete-se o treinamento durante mais uma época. O ajuste dos pesos é feito de acordo com os erros computados a cada amostra.

O algoritmo do *back-propagation* aplica uma correção $\Delta w_k(i)$ ao peso sináptico $w_j(i)$ proporcional a $\partial \mathcal{E}(i)/\partial w_j(i)$. Conforme a regra da cadeia do cálculo e usando as relações (4.1) e (4.2), esse gradiente pode ser escrito como

$$\frac{\partial \mathcal{E}(i)}{\partial w_k(i)} = \frac{\partial \mathcal{E}(i)}{\partial e_j(i)} \frac{\partial e_j(i)}{\partial y_j(i)} \frac{\partial y_j(i)}{\partial v_j(i)} \frac{\partial v_j(i)}{\partial w_j(i)}. \quad (4.8)$$

Derivando ambos os lados de (4.6) em relação a $e_j(i)$, obtém-se

$$\frac{\partial \mathcal{E}(i)}{\partial e_j(i)} = e_j(i). \quad (4.9)$$

Realizando o mesmo processo com (4.5), porém em relação a $y_j(i)$, pode-se obter

$$\frac{\partial e_j(i)}{\partial y_j(i)} = -1. \quad (4.10)$$

De forma semelhante, derivando (4.2) em relação a $v_j(i)$, tem-se

$$\frac{\partial y_j(i)}{\partial v_j(i)} = \phi'_j(v_j(i)), \quad (4.11)$$

onde o uso do apóstrofo indica derivada em relação ao argumento. Finalmente, derivando o campo local induzido - escrito semelhante a (4.1) porém para um neurônio j - em relação a $w_j(i)$, obtém-se

$$\frac{\partial v_j(i)}{\partial w_j(i)} = x_j(i). \quad (4.12)$$

Desta forma, pode-se reescrever a expressão (4.8) conforme

$$\frac{\partial \mathcal{E}(i)}{\partial w_k(i)} = -e_j(i) \phi'_j(v_j(i)) x_j(i). \quad (4.13)$$

A correção $\Delta w_k(i)$ aplicada ao peso sináptico $w_k(i)$ que aponta na direção oposta ao gradiente é definida como

$$\Delta w_j(i) = -\eta \frac{\partial \mathcal{E}(i)}{\partial w_j(i)}, \quad (4.14)$$

onde η é a taxa de aprendizado que deve ser definida pelo usuário. Utilizando (4.13) em (4.14), obtém-se

$$\Delta w_j(i) = \eta \delta_j(i) x_j(i), \quad (4.15)$$

onde $\delta_j(i)$ é chamado de gradiente local, definido como

$$\delta_j(i) = -\frac{\partial \mathcal{E}(i)}{\partial v_j(i)} = -\frac{\partial \mathcal{E}(i)}{\partial e_j(i)} \frac{\partial e_j(i)}{\partial y_j(i)} \frac{\partial y_j(i)}{\partial v_j(i)} = e_j(i) \phi'_j(v_j(i)), \quad (4.16)$$

que pode ser compreendido de forma abstrata como um vetor que indica mudanças necessárias nos pesos sinápticos para a redução do erro médio quadrático.

Existem duas formas de utilizar o algoritmo do *back-propagation* em redes MLP: atualizando os pesos a cada amostra de dados ou a cada batelada¹ de dados - conjunto de amostras de tamanho determinado. No caso do treinamento por *batch*, as derivadas ainda são calculadas a cada amostra, porém ao atualizar-se o pesos sinápticos, leva-se em conta os gradientes obtidos nas últimas K amostras, sendo K o tamanho de um *batch* de dados.

4.4.2 *Back-propagation through time*

A técnica do *back-propagation* aplicada à redes neurais recorrentes é denominada de *back-propagation through time* pois, com as realimentações, há de se considerar a dependência temporal causada por estas. Para que este tipo de treinamento seja aplicável à redes recorrentes, deve-se realizar o *unfolding in time* - ou desdobramento no tempo - da rede neural.

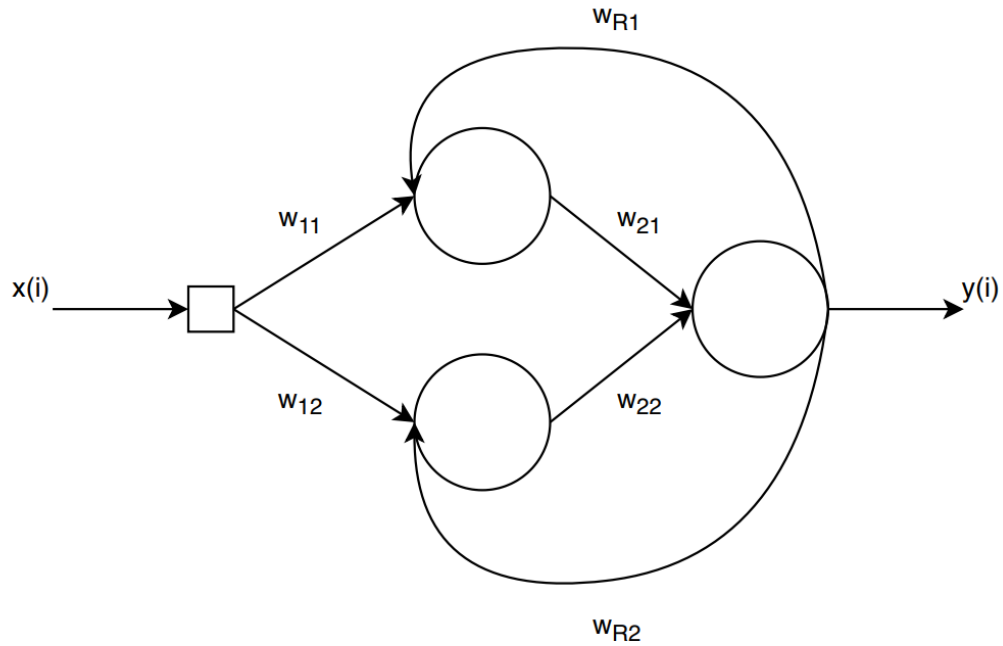
4.4.2.1 *Unfolding in time*

O *unfolding* consiste na representação de uma rede neural recorrente como uma rede neural *feedforward* como profundidade temporal limitada. Idealmente, esta profundidade temporal seria equivalente ao número total de amostras de treinamento, porém é necessário limitá-la para que o treinamento possa ser processado em tempo hábil.

Por questões didáticas, o *unfolding* será exemplificado para uma rede neural recorrente de uma única camada oculta, com dois neurônios. A ideia pode ser facilmente expandida para redes neurais com mais camadas, mais neurônios, e diferentes tipos de realimentações. Considera-se aqui uma rede neural recorrente, com realimentações de Jordan, dois neurônios em sua camada oculta e um neurônio na camada de saída, todos com função de ativação não linear, representada conforme a Figura 16.

¹ Termo comumente utilizado em inglês: *batch*.

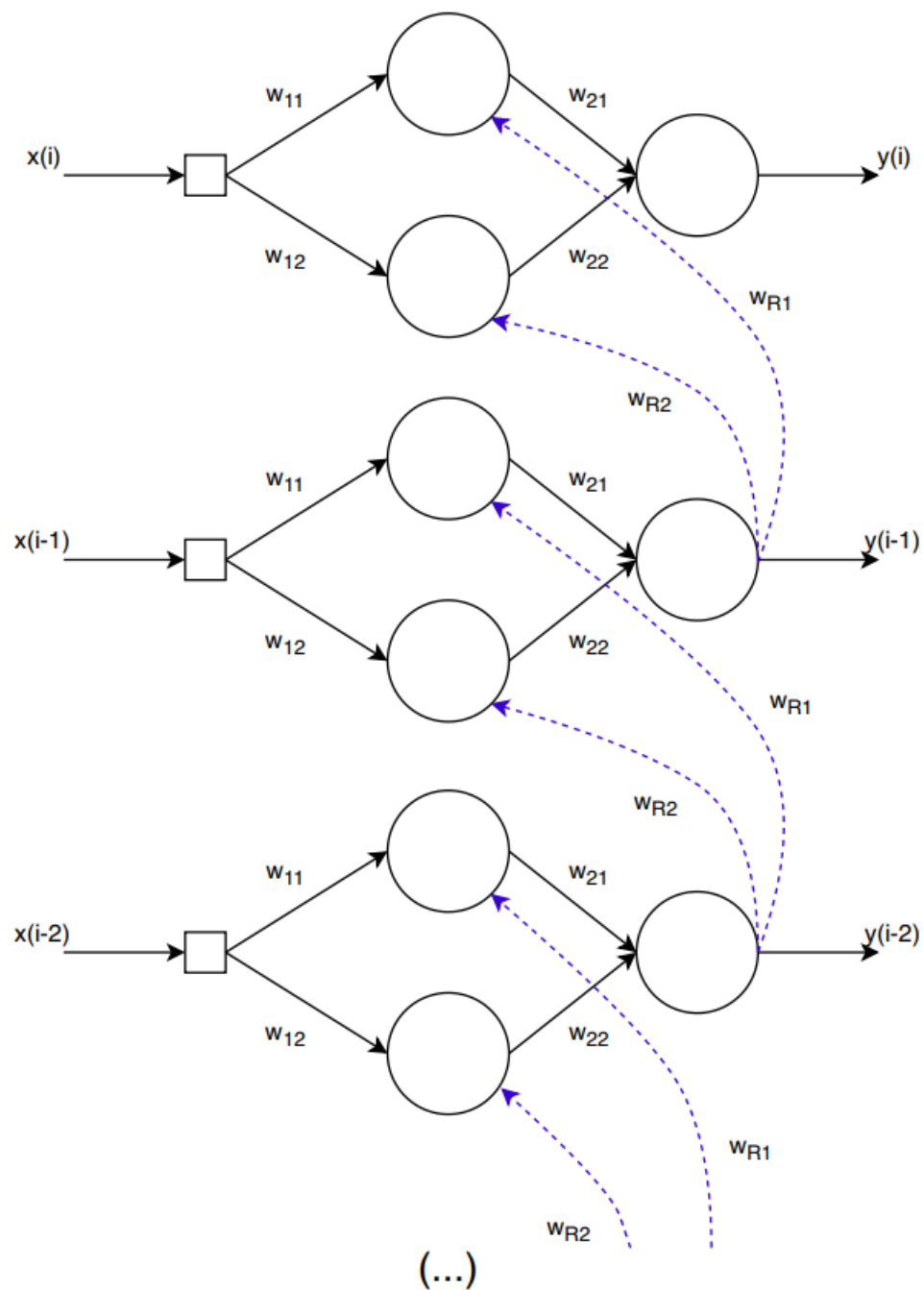
Figura 16 – Rede recorrente de Jordan com uma camada oculta.



Fonte: o próprio autor, 2019.

As realimentações representadas pelos pesos sinápticos w_{R1} e w_{R2} realizam, também, um atraso de uma amostra no valor de saída, causando a dependência temporal no sistema. Assim, representando a mesma rede neural acima mostrando explicitamente a dependência temporal, tem-se a rede *unfolded* conforme a Figura 17. Pode-se observar que a primeira camada superior desta figura possui índice i , portanto, para questões de equacionamento, esta será denominada como camada temporal h , ou a última camada, sendo h o número total de camadas temporais. Percebe-se também que os pesos sinápticos são os mesmos para todas as camadas temporais da rede. O número de camadas temporais, idealmente, é igual ao número de amostras total aplicadas à rede neural no treinamento.

Figura 17 – Rede recorrente de Jordan desdobrada no tempo.



Fonte: o próprio autor, 2019.

Para propagar o sinal de forma *feedforward* pela rede neural recorrente desdobrada no tempo (Algoritmo 1) é necessário calcular os campos locais induzidos, assim como saídas dos neurônios para cada camada temporal, para cada amostra, pois estes dados serão utilizados no treinamento. Necessita-se, assim, de uma memória de armazenamento equivalente ao quadrado do número de amostras disponíveis para treinamento multiplicado pelo número de neurônios na rede.

Algoritmo 1: Propagação *feedforward* em uma RNR *unfolded*

```

1 Para i = 1 até número de amostras
2   Para r = 1 até h
3     Se r = 1
4        $v(r) = \sum_{k=1}^m [w_k x_k(r)] + b w_b$ 
5        $y(r) = f(v(r))$ 
6     se não
7        $v(r) = \sum_{k=1}^m [w_k x_k(r)] + b w_b + y(r-1) w_r(i)$ 
8        $y(r) = f(v(r))$ 
9    $Y(i) = y(h)$ 

```

No Algoritmo 1, i é o índice de amostras dos dados, r indica a camada temporal atual, h representa o número total de camadas temporais e m é o número de entradas de um neurônio. Conforme pode-ser observar na Figura 17, a saída atual $Y(i)$ é dada pela última camada temporal $r = h$. Devido ao problema do *vanishing gradient*, informações de tempos passados distantes tem pouco efeito sobre o valor atual da saída do sistema. Assim, sugere-se a utilização de uma profundidade h de 10 a 20 (JAEGER, 2002).

Há também a opção de não limitar a profundidade da rede desdobrada no tempo, mas sim limitar o número de amostras por época, possibilitando que a rede neural assuma uma profundidade igual ao número total de amostras. Ambas as possibilidades de se utilizar o *back-propagation through time* aqui citadas são apresentadas na sequência.

4.4.2.2 Epochwise back-propagation through time

Separando-se o conjunto de dados de treinamento em épocas independentes, grandes o bastante para que possa-se perceber um padrão temporal, a função custo é dada por

$$\mathcal{E}_{total}(n_o, n_1) = \frac{1}{2} \sum_{i=n_o}^{n_1} \sum_{j \in \mathcal{A}} e_j^2(i), \quad (4.17)$$

onde $\mathcal{A}$ é o conjunto de índices de neurônios de saída em que valores desejados são definidos do conjunto de dados de treinamento, e $e_j(i)$ é o sinal de erro da saída desses neurônios.

Para calcular os gradientes do erro pela rede, utiliza-se o *epochwise backpropagation through time* (eBPTT) conforme:

- primeiro, realiza-se uma propagação *forward* dos dados de um *batch* (no intervalo (i_o, i_1)). Os valores das saídas dos neurônios, pesos sinápticos e dados de entrada/saída devem ser

salvos;

- realiza-se uma propagação do erro semelhante ao esquema do *backpropagation* comum (Subseção 4.4.1), porém incluindo as realimentações. O novo gradiente local

$$\delta_j(i) = -\frac{\partial \mathcal{E}_{total}(i_o, i_1)}{\partial v_j(i)} \quad (4.18)$$

é dado conforme

$$\delta_j(i) = \begin{cases} \phi'(v_j(i)) e_j(i) & \text{se } i = i_1 \\ \phi'(v_j(i)) [e_j(i) + \sum_{k \in \mathcal{A}} w_{jk} \delta_k(i+1)] & \text{se } i_o < n < i_1, \end{cases} \quad (4.19)$$

onde j é o neurônio em que o gradiente local está sendo calculado e k representa uma camada temporal superior, de onde a realimentação está vindo - o conjunto $\mathcal{A}$ são as realimentações que conectam-se ao neurônio j . Para que o equacionamento acima seja válido, assume-se que todos os neurônios possuem a mesma função de ativação ϕ . Caso contrário, deve-se alterar a derivada da função de ativação conforme cada neurônio;

- assim que os gradientes forem calculados até a camada temporal $i_o + 1$, o ajuste aplicado aos pesos é dado por

$$\Delta w_{ji} = -\eta \frac{\partial \mathcal{E}_{total}(i_o, i_1)}{\partial w_{ji}} = \eta \sum_{i=i_o+1}^{i_1} \delta_j(i) x_j(i-1), \quad (4.20)$$

onde w_{ji} é o peso sináptico que leva o estímulo x_j ao neurônio j .

4.4.2.3 Truncated backpropagation through time

A função custo a ser minimizada para realizar a atualização dos pesos a cada amostra é dada por

$$\mathcal{E}(i) = \frac{1}{2} \sum_{j \in \mathcal{A}} e_j^2(i). \quad (4.21)$$

O gradiente local é definido para o neurônio j como

$$\delta_j(r) = -\frac{\partial \mathcal{E}(r)}{\partial v_j(r)} \quad \forall j \in \mathcal{A} \quad \text{e} \quad i-h < r \leq i, \quad (4.22)$$

resultando em

$$\delta_j(r) = \begin{cases} \phi'(v_j(i)) e_j(i) & \text{se } r = i \\ \phi'(v_j(i)) \sum_{k \in \mathcal{A}} w_{kj} \delta_k(i+1) & \text{se } i-h < r < i. \end{cases} \quad (4.23)$$

Assim, a atualização dos pesos sinápticos é realizada de acordo com

$$\Delta w_{ji}(i) = \eta \sum_{r=i-h+1}^i \delta_j(r) x_j(r-1). \quad (4.24)$$

Para que w_{kj} seja utilizado em (4.23), deve-se utilizar um η pequeno o bastante para que se possa considerar que não há variação no peso sináptico entre cada camada temporal, somente entre cada amostra. Percebe-se que diferentemente da relação (4.19), para o caso do *backpropagation through time* truncado (tBPTT), em (4.23) o erro só é inserido no cálculo quando $r = h$, eliminando a necessidade de guardar a informação do erro a cada camada temporal, como é feito no eBPTT.

4.4.2.4 Vanishing gradient

A emulação de sistemas não lineares comumente será dinâmica no tempo, isto é, certa resposta desejada atual dependerá de informações (amostras) passadas. Essa dependência pode ser de poucas amostras de diferença (*short-term*) ou de muitas amostras de diferença (*long-term*).

Conforme Hochreiter e Schmidhuber (1997), supondo-se uma rede recorrente *fully connected*² e considerando que o erro está fluindo de um neurônio u a um neurônio j , o erro flui "de volta no tempo" durante q intervalos de tempo. Tal fato irá escalar o erro local conforme

$$\frac{\partial \delta_j(t-q)}{\partial \delta_u(t)} = \begin{cases} f'_j(v_j(t-1)) w_{uj} & \text{se } q = 1 \\ f'_j(v_j(t-q)) \sum_{l=1}^n \frac{\partial \delta_l(t-q+1)}{\partial \delta_u(t)} w_{lj} & \text{se } q > 1. \end{cases} \quad (4.25)$$

Com $l_q = j$ e $l_0 = u$, obtém-se

$$\frac{\partial \delta_j(t-q)}{\partial \delta_u(t)} = \sum_{l_1=1}^n \cdots \sum_{l_{q-1}=1}^n \prod_{m=1}^q f'_{lm}(v_{lm}(t-m)) w_{lm l_{m-1}}. \quad (4.26)$$

A soma dos n_{q-1} termos $\prod_{m=1}^q f'_{lm}(v_{lm}(t-m)) w_{lm l_{m-1}}$ determina o erro total que flui pela rede. Conforme (4.26), se

$$|f'_{lm}(v_{lm}(t-m)) w_{lm l_{m-1}}| > 1 \quad (4.27)$$

para todo m , então o maior produto aumenta exponencialmente com q , ou seja, o valor do erro tende a ser muito alto ("explode"). Erros conflitantes (muito diferentes) ocasionarão em pesos que oscilam demais e adquirem um comportamento instável. Da mesma forma, se

$$|f'_{lm}(v_{lm}(t-m)) w_{lm l_{m-1}}| < 1 \quad (4.28)$$

para todo m , o maior produto decresce exponencialmente com q , ou seja, o erro desaparece e nada pode ser aprendido em um tempo aceitável. Este é o *vanishing gradient* - gradiente que desaparece.

Para o caso de f_{lm} ser a função logística, com valor máximo igual a 1 e eixo abissal centrado na origem, cuja expressão é dada por

$$f_{lm}(x) = \frac{1}{1 + e^{-x}}, \quad (4.29)$$

² *Fully connected* significa que todos os neurônios e entradas estão conectados entre si, sem exceções. Tal forma de utilização de uma rede neural é usada com intenções didáticas ou de generalização da aplicação das teorias. Na prática, as conexões entre neurônios são limitadas (HOCHREITER; SCHMIDHUBER, 1997).

o maior valor que sua derivada pode assumir é de 0,25. Se a saída do neurônio anterior, cuja expressão é dada por

$$y_{lm-1} = f'_{lm-1} (net_{lm-1}(t - m)), \quad (4.30)$$

for constante e diferente de zero, então, conforme mostrado por Hochreiter e Schmidhuber (1997), $|f'_{lm}(v_{lm}(t - m)) w_{lm-1}|$ assume valores máximos onde

$$w_{lm-1} = \frac{1}{y_{lm-1}} \coth\left(\frac{1}{2}v_{lm}\right) \quad (4.31)$$

é zero para $|w_{lm-1}| \rightarrow \infty$, e é menor que 1 para $|w_{lm-1}| < 4$, ou seja, o valor absoluto máximo de pesos sinápticos é menor que 4. Assim, com a função de ativação logística, o erro tende a zero sempre que o módulo do valor dos pesos for inferior a 4, principalmente no início do treinamento (HOCHREITER; SCHMIDHUBER, 1997). O autor também apresenta a demonstração do *vanishing gradient* para o erro global, de forma semelhante à demonstração aqui exposta.

Três procedimentos são citados por Haykin (1999) para reduzir as dificuldades no treinamento devido ao *vanishing gradient*, sendo elas: aumentar o período temporal dos dados de entrada e saída; utilizar o filtro de Kalman estendido para um uso mais eficiente da informação durante o treinamento; utilizar métodos de otimização baseados nos métodos Quasi-Newton.

4.4.3 Filtro de Kalman estendido

A forma de treinamento contínua (atualizando os pesos a cada amostra) demonstra-se lenta para técnicas de aprendizado baseadas puramente em gradiente descendente (HAYKIN, 1999). Ao mesmo tempo, tais técnicas apresentam uma grande dificuldade no treinamento devido ao *vanishing gradient*, como exposto em 4.4.2.4. Uma das formas de solucionar tais problemas é utilizar o filtro de Kalman estendido (*extended Kalman filter*) como técnica de treinamento, tratando o aprendizado da rede neural como um problema de filtragem.

4.4.3.1 Filtro de Kalman

Considera-se um sistema dinâmico linear e em tempo discreto, representado na Figura 18, na forma de espaço de estados como:

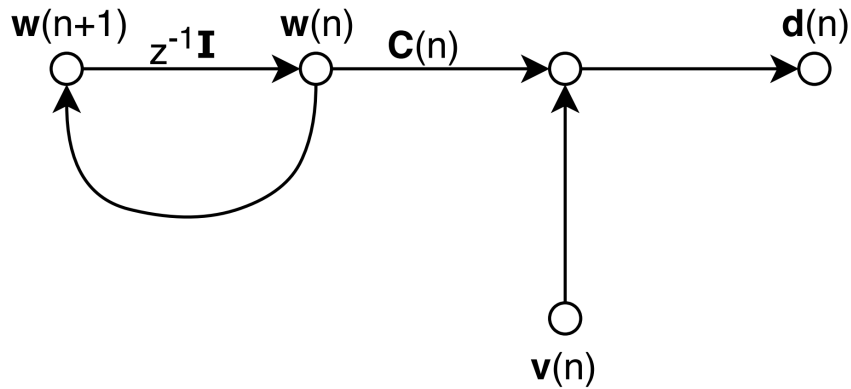
$$\mathbf{w}(n+1) = \mathbf{w}(n), \quad (4.32a)$$

$$\mathbf{d}(n) = \mathbf{C}(n) \mathbf{w}(n) + \mathbf{v}(n), \quad (4.32b)$$

onde n é o índice temporal, de acordo com a Figura 18, onde:

- $\mathbf{w}(n)$ é o vetor de estados do sistema;
- $\mathbf{d}(n)$ é o vetor de observação;
- $\mathbf{C}(n)$ é a matrix de medição;
- $\mathbf{v}(n)$ é o ruído de medição.

Figura 18 – Grafo de fluxo de sinal para um sistema dinâmico e linear.



Fonte: adaptado de (HAYKIN, 1999).

Na expressão (4.32a), duas simplificações foram feitas: foi assumido que não há ruído no processo e que a matriz de transição que leva os estados do tempo $n + 1$ para o tempo n é a matriz identidade.

O filtro de Kalman é um filtro cuja abordagem se dá no domínio do tempo, e não da frequência. Utiliza-se propriedades estatísticas de estimadores para resolver um problema de otimização, que minimiza o erro entre a predição e a medição para todo instante de tempo. Assim, pode-se definir o problema a ser resolvido pelo filtro de Kalman como: "*utilizando toda a informação observada, consistindo de um conjunto de vetores $\mathbf{d}(i)_{i=1}^n$, deve-se encontrar para cada $n \geq 1$ a estimação do vetor de estados $\mathbf{w}(i)$ que possui o menor erro médio quadrático.*". A solução do problema é baseada nas seguintes suposições:

1. O ruído de medição $\mathbf{v}(n)$ é um ruído branco de média nula com distribuição gaussiana, ou seja, cuja matriz de covariância é definida como

$$E[\mathbf{v}(n) \mathbf{v}^T(k)] = \begin{cases} \mathbf{R}(n), & \text{se } n = k \\ \mathbf{0}, & \text{se } n \neq k \end{cases} \quad (4.33)$$

2. O valor inicial dos estados, $\mathbf{w}(0)$, é descorrelacionado com $\mathbf{v}(n)$ para todo $n \geq 0$.

Conforme (HAYKIN, 1999), uma inovação é dada por

$$\boldsymbol{\alpha}(n) = \mathbf{d}(n) - \hat{\mathbf{d}}(n|n-1), \quad (4.34)$$

onde $\hat{\mathbf{d}}(n|n-1)$ é a estimação de menor erro médio quadrático de $\mathbf{d}(n)$, dados todos os valores passados que estão situados no vetor de observações desde o tempo $k = 1$ até $k = n - 1$. A inovação $\boldsymbol{\alpha}(n)$ pode ser compreendida como a nova informação contida em $\mathbf{d}(n)$ que não está disponível na predição $\hat{\mathbf{d}}(n|n-1)$. Em outras palavras:

1. O processo de inovação $\alpha(n)$ associado com $\mathbf{d}(n)$ é descorrelacionado com todas as observações passadas $\mathbf{d}(1), \mathbf{d}(2), \dots, \mathbf{d}(n-1)$, conforme

$$E[\alpha(n)\mathbf{d}^T(k)] = \mathbf{0} \text{ para } 1 \leq k \leq n-1; \quad (4.35)$$

2. As inovações consistem em uma sequência de vetores aleatórios descorrelacionados entre si, de acordo com

$$E[\alpha(n)\alpha^T(k)] = \mathbf{0} \text{ para } 1 \leq k \leq n-1; \quad (4.36)$$

3. Para cada $\mathbf{d}(n)$, há um $\alpha(n)$ correspondente. Assim, pode-se substituir uma sequência de dados observados correlacionados por dados descorrelacionados (inovações), deixando o algoritmo mais simples mas ainda sem perder informações.

Além das citadas acima, o filtro de Kalman possui mais três matrizes/quantidades:

- $\mathbf{K}(n|n-1)$: a matriz de covariância do erro, definida por

$$\mathbf{K}(n|n-1) = E[\boldsymbol{\varepsilon}(n|n-1)\boldsymbol{\varepsilon}^T(n|n-1)], \quad (4.37)$$

onde o erro de estados $\boldsymbol{\varepsilon}(n|n-1)$ é definido como

$$\boldsymbol{\varepsilon}(n|n-1) = \mathbf{w}(n) - \hat{\mathbf{w}}(n|n-1), \quad (4.38)$$

sendo que $\mathbf{w}(n)$ é o estado atual e $\hat{\mathbf{w}}(n|n-1)$ é a predição um passo a frente baseada em valores observados um em um instante $n-1$.

- $\Gamma(n)$: fator de conversão que relaciona a estimativa de erro filtrada $\mathbf{e}(n)$ às inovações $\alpha(n)$, conforme

$$\mathbf{e}(n) = \mathbf{R}(n) \Gamma(n) \alpha(n), \quad (4.39)$$

onde

$$\mathbf{e}(n) = \mathbf{d}(n) - \hat{\mathbf{d}}(n|n), \quad (4.40)$$

em que $\hat{\mathbf{d}}(n|n)$ é a estimativa do vetor de observações $\mathbf{d}(n)$ dadas todas as observações até o tempo n .

- $\mathbf{G}(n)$ é o ganho de Kalman, que determina a correção utilizada na estimação do estado.

O filtro de Kalman é sintetizado pelo Algoritmo 2.

Algoritmo 2: Filtro de Kalman

- 1 Para $n = 1, 2, 3, \dots$, calcular:
 - 2 $\Gamma(n) = [\mathbf{C}(n) \mathbf{K}(n, n-1) \mathbf{C}^T(n) + \mathbf{R}(n)]^{-1}$
 - 3 $\mathbf{G}(n) = \mathbf{K}(n, n-1) \mathbf{C}^T(n) \Gamma(n)$
 - 4 $\alpha(n) = \mathbf{y}(n) - \mathbf{C}(n) \hat{\mathbf{w}}(n|n-1)$
 - 5 $\hat{\mathbf{w}}(n+1|n) = \hat{\mathbf{w}}(n|n-1) + \mathbf{G}(n) \alpha(n)$
 - 6 $\mathbf{K}(n+1, n) = \mathbf{K}(n, n-1) - \mathbf{G}(n) \mathbf{C}(n) \mathbf{K}(n, n-1)$
-

4.4.3.2 Filtro de Kalman estendido

Como foi mostrado, o desenvolvimento da teoria do filtro de Kalman é dado sob sistemas lineares dinâmicos. Para ser possível desenvolvê-la no caso de uma rede neural recorrente, uma forma de linearização deve ser aplicada. Com isso em mente, pode-se reescrever as expressões (4.32) de forma generalizada:

$$\mathbf{w}(n+1) = \mathbf{w}(n), \quad (4.41a)$$

$$\mathbf{d}_o(n) = \mathbf{c}(\mathbf{w}(n), \mathbf{u}(n), \mathbf{r}(n)) + \mathbf{v}(n), \quad (4.41b)$$

onde $\mathbf{w}(n)$ são os estados, $\mathbf{u}(n)$ e $\mathbf{r}(n)$ são respectivamente os vetores de entrada e de atividades das recorrências ³, enquanto que $\mathbf{c}(\cdot)$ é a função não linear que representa a não linearidade da entrada para a saída da rede neural. $\mathbf{v}(n)$ é um ruído do sistema, branco, com distribuição gaussiana e matriz de covariância $\mathbf{R}(n)$.

Considerando a linearização de um sistema não linear cujos estados a serem adaptados pelo filtro de Kalman - no caso de uma rede neural recorrente, o vetor de pesos sinápticos $\mathbf{w} \in \mathbb{R}^W$, sendo W o número total de pesos - pode-se escrever a expressão (4.41b) como

$$\mathbf{d}(n) = \mathbf{C}(n) \mathbf{w}(n) + \mathbf{v}(n), \quad (4.42)$$

onde $\mathbf{C} \in \mathbb{R}^{1 \times W}$ é a matriz de medição do modelo linearizado do sistema. A linearização consiste em uma matriz de derivadas parciais da saída pelos W pesos sinápticos da rede, conforme

$$\mathbf{C}(n) = \begin{bmatrix} \frac{\partial \mathbf{c}}{\partial w_1} & \frac{\partial \mathbf{c}}{\partial w_2} & \cdots & \frac{\partial \mathbf{c}}{\partial w_W} \end{bmatrix}. \quad (4.43)$$

Para aplicações práticas, as derivadas da matriz de medição linearizada são calculadas utilizando o algoritmo do *back-propagation through time* de forma adaptada, modificando a expressão (4.16) conforme

$$\delta_j(n) = \frac{\partial y_j(n)}{\partial v_j(n)} = \phi'_j(v_j(n)). \quad (4.44)$$

Desta forma, os termos $\frac{\partial \mathbf{c}}{\partial \mathbf{w}}$ podem ser compreendidos como $\frac{\partial y}{\partial \mathbf{w}}$, uma vez que $\mathbf{v}$ é um ruído branco, e logo, descorrelacionado com $\mathbf{w}$, pode-se obter δ_j conforme

$$\delta_j(r) = \begin{cases} \phi'(v_j(i)) & \text{se } r = i \\ \phi'(v_j(i)) \sum_{k \in \mathcal{A}} w_{kj} \delta_k(i+1) & \text{se } i-h < r < i, \end{cases} \quad (4.45)$$

considerando a rede neural recorrente na sua forma *unfolded*, como é feito no algoritmo do BPTT truncado. Finalmente, pode-se adaptar o Algoritmo 2 para o caso do filtro de Kalman estendido,

³ Atividades das recorrências refere-se ao estado temporal da rede neural recorrente. Tal dependência temporal, conforme já citado anteriormente, é causada pelas realimentações - ou recorrências.

conforme apresentado no Algoritmo 3.

Algoritmo 3: Filtro de Kalman estendido

- 1 Para $n = 1, 2, 3, \dots$, calcular:
 - 2 $\mathbf{C}(n|n-1) = \frac{\partial \mathbf{y}(n)}{\partial \mathbf{w}(n|n-1)}$
 - 3 $\mathbf{\Gamma}(n) = [\mathbf{C}(n) \mathbf{K}(n, n-1) \mathbf{C}^T(n) + \mathbf{R}(n)]^{-1}$
 - 4 $\mathbf{G}(n) = \mathbf{K}(n, n-1) \mathbf{C}^T(n) \mathbf{\Gamma}(n)$
 - 5 $\boldsymbol{\alpha}(n) = \mathbf{d}(n) - \hat{\mathbf{d}}(n|n-1)$
 - 6 $\hat{\mathbf{w}}(n+1|n) = \hat{\mathbf{w}}(n|n-1) + \mathbf{G}(n) \boldsymbol{\alpha}(n)$
 - 7 $\mathbf{K}(n+1, n) = \mathbf{K}(n, n-1) - \mathbf{G}(n) \mathbf{C}(n) \mathbf{K}(n, n-1)$
-

Esta forma do *extended Kalman filter* é descrita na literatura como sendo global. Há também sua forma desacoplada, que divide as não linearidades c conforme o número de saídas. Para o caso aqui apresentado, é utilizado o filtro de Kalman estendido global, devido a uma única saída ser referência para o cálculo do erro. Também existem formas de desacoplar o filtro de Kalman para o caso de uma única saída visando obter maior velocidade na execução do algoritmo, que não serão abordadas neste trabalho, por questões de simplicidade, e são apresentadas em (HAYKIN, 1999).

4.5 LONG SHORT-TERM MEMORY

Uma outra forma de solucionar o problema de *vanishing gradient* das redes neurais recorrentes mais comuns (Jordan e Elman), a rede *long short-term memory* (LSTM) foi proposta em (HOCHREITER; SCHMIDHUBER, 1997). Tal rede é capaz de representar sistemas com dependências temporais de muitas amostras mesmo em casos onde há ruído nas medições e no próprio sistema. A principal característica da rede LSTM é a constância na forma que o erro flui pela rede durante o treinamento. Esta característica é abordada mais a fundo na sequência.

4.5.1 Constant error carousel

A ideia por trás da LSTM é que o erro flua de forma constante pela rede durante o processo de aprendizagem, evitando que o gradiente do erro utilizado para a atualização dos pesos assumam valores muito pequenos (*vanishing gradient*) ou muito grandes (*exploding gradient*).

Em um instante i , o erro e_j flui da saída para a entrada da rede neural no formato

$$e_j(i) = f'_j(v_j(i)) e_j(i+1) w_{jj}, \quad (4.46)$$

onde j é o índice que representa o neurônio em questão, f_j é a função de ativação do neurônio j , $v_j(i)$ é o campo local induzido do neurônio j no instante i , e w_{jj} é o peso sináptico que conecta o neurônio j com o neurônio $j+1$.

Para que o erro flua de forma constante, deve-se garantir que

$$f'_j(v_j(i)) w_{jj} = \text{constante}. \quad (4.47)$$

Conforme observado em (MOZER, 1989), uma constante temporal de valor unitário é apropriada para representar atrasos temporais de qualquer valor. Assim, considerando que a constante na relação (4.47) é igual a 1 e integrando-a em relação a $v_j(i)$, obtém-se

$$f_j(v_j(i)) = \frac{v_j(i)}{w_{jj}}, \quad (4.48)$$

para qualquer $v_j(i)$. Isso significa que f_j deve ser linear, devido a relação de proporcionalidade linear $v_j(i)/w_{jj}$, e a ativação do neurônio j deve manter-se constante, portanto:

$$y_j(i+1) = f_j(v_j(i+1)) = f_j(w_{jj} y_j(i)) = y_j(i). \quad (4.49)$$

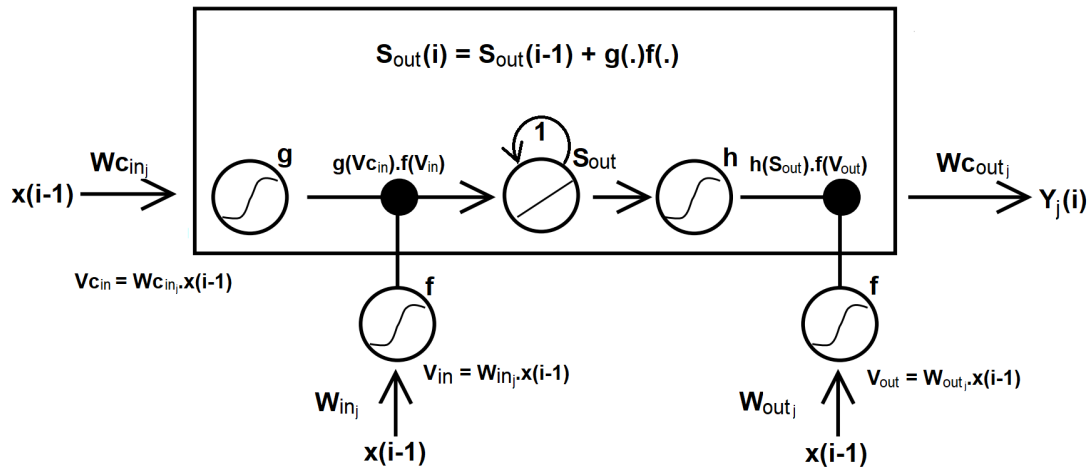
Tal constância na forma que o erro flui pela rede foi denominada por Hochreiter e Schmidhuber (1997) de *constant error carousel* (CEC) e é a principal característica da LSTM.

4.5.2 A célula de memória

A LSTM consiste em um agrupamento de células de memória com *gate units*, que podem ser analogamente entendidos como o "neurônio" deste tipo de rede. A célula da LSTM armazena informação e controla quais dados de entrada ou saídas são relevantes, por meio das *gate units*, estas que possuem função de ativação do tipo sigmoide - uma função não linear saturada para valores muito pequenos ou muito grandes, cuja derivada no seu ponto central é unitária. Uma *gate unit* de entrada é utilizada para proteger a informação contida na memória a dados de entrada irrelevantes. Da mesma forma, há uma *gate unit* na saída para proteger outras células LSTM de dados irrelevantes que foram armazenados na célula em questão.

A célula LSTM proposta em (HOCHREITER; SCHMIDHUBER, 1997) pode ser observada conforme a Figura 19. Percebe-se a presença das *gate units* de entrada (g e f), de saída (h) e da unidade central linear com uma realimentação unitária - o CEC, cuja saída é representada por *Sout*.

Figura 19 – Célula LSTM.



Fonte: adaptado de (HOCHREITER; SCHMIDHUBER, 1997).

Conforme proposto originalmente por Hochreiter e Schmidhuber (1997), as funções de ativação f , g e h são, respectivamente, a função logística

$$f(x) = \frac{1}{1 + \exp(-x)}, \quad (4.50)$$

uma função sigmoide semelhante à anterior, porém modificada, de acordo com

$$g(x) = \frac{4}{1 + \exp(-x)} - 2, \quad (4.51)$$

e uma função sigmoide semelhante à tangente hiperbólica,

$$h(x) = \frac{2}{1 + \exp(-x)} - 1. \quad (4.52)$$

Em suma, o funcionamento da célula centra-se na ideia de que sinais de erro presos no CEC não podem ser modificados, ou seja, não tendem a zero (*vanish*) e nem a infinito. A *gate unit* de saída será responsável por aprender quais erros devem ficar presos no CEC, enquanto que a *gate unit* de entrada deve aprender quando liberar os sinais de erro para o CEC.

A estrutura de rede LSTM é formada por camadas de células de memória, assim como a estrutura das redes recorrentes *vanilla*, porém, no lugar dos neurônios têm-se as células. Como a LSTM é uma RNR, o treinamento pode ser realizado da mesma forma que as rede *vanilla*, conforme as técnicas apresentadas na Seção 4.4. A forma de realizar o desdobramento da rede no tempo é análoga à forma apresentada na Subseção 4.4.2.1, somente será mais complexa devido a própria complexidade da estrutura da célula de memória.

5 PROJETO E SIMULAÇÃO DO PRÉ-AMPLIFICADOR VALVULADO

A simulação de um sistema torna seu estudo mais conveniente, possibilitando que mais testes possam ser realizados em tempo hábil. O objetivo da simulação neste caso é testar as capacidades das redes neurais ao emular as formas de onda de um pré-amplificador valvulado, reduzindo assim o número de testes experimentais necessários para a avaliação da estrutura das redes.

Como a válvula triodo é um componente pouco utilizado nos dias atuais, não há, por padrão, a presença de um modelo dessa em softwares de simulação, sendo necessário, com base na literatura, implementar um modelo presente na literatura que possa ser representado por meio dos componentes já disponíveis no programa. Portanto, a modelagem de uma válvula triodo em um software de simulação e a simulação do pré-amplificador valvulado são realizados neste capítulo.

5.1 MODELO DE SIMULAÇÃO

Duas escolhas devem ser feitas ao simular o modelo da válvula: qual modelo matemático e qual software de simulação devem ser utilizados. Para o modelo, optou-se por utilizar o que há de mais recente que teve relevância na literatura: o modelo de Dempwolf (DEMPWOLF; ZÖLZER, 2011). Quanto ao software de simulação, a princípio, prioriza-se a fácil acessibilidade ao programa, sem muitas restrições.

O objetivo principal desta simulação é obter formas de onda semelhantes ao sistema real, porém não necessariamente iguais. As redes neurais são sistemas relativamente robustos no quesito complexidade dos sinais quando estas são utilizadas para a identificação, portanto pequenas diferenças entre o modelo matemático e o sistema real podem ser consideradas irrelevantes em testes iniciais. A seguir são apresentadas as equações do modelo escolhido e sua implementação via *PowerSIM*®.

5.1.1 Modelo de Dempwolf e sua simulação

Um novo modelo equivalente para válvulas triodo do tipo 12AX7 é apresentado por Dempwolf e Zölzer (2011), com equações que representam o sistema de forma contínua e diferenciável em todos os pontos, diferentemente de modelos anteriores como o modelo de Koren (KOREN, 2003), facilitando a sua implementação em softwares de simulação. Tais equações referem-se à corrente no anodo, no catodo e na grade, dependendo das tensões grade-catodo (V_{gk}) e anodo-catodo (V_{ak}).

A corrente no catodo e na grade são dadas, respectivamente por

$$I_k = G \cdot \left(\log \left(1 + \exp \left(C \cdot \left(\frac{1}{\mu} \cdot V_{ak} + V_{gk} \right) \right) \right) \cdot \frac{1}{C} \right)^\gamma, \quad (5.1)$$

e

$$I_g = G_g \cdot \left(\log(1 + \exp(C_g \cdot V_{gk})) \cdot \frac{1}{C_g} \right)^\xi + I_{g0}, \quad (5.2)$$

onde G , μ , γ , C , G_g , ξ , C_g , I_{g0} são parâmetros das válvulas obtidos por meio de medições para que o modelo seja adaptado tipo de triodo em questão e são descritos em (DEMPWOLF; ZÖLZER, 2011). A corrente do anodo I_a é simplesmente dada pela diferença entre a corrente do catodo e a corrente da grade.

Para válvulas 12AX7, os parâmetros do modelo foram obtidos em (DEMPWOLF; ZÖLZER, 2011) e são dados conforme a Tabela 1. RSD-1, RSD-2 e EHX-1 referem-se à marca das válvulas. O número após as letras em caixa alta refere-se a ser uma válvula diferente, porém do mesmo fabricante.

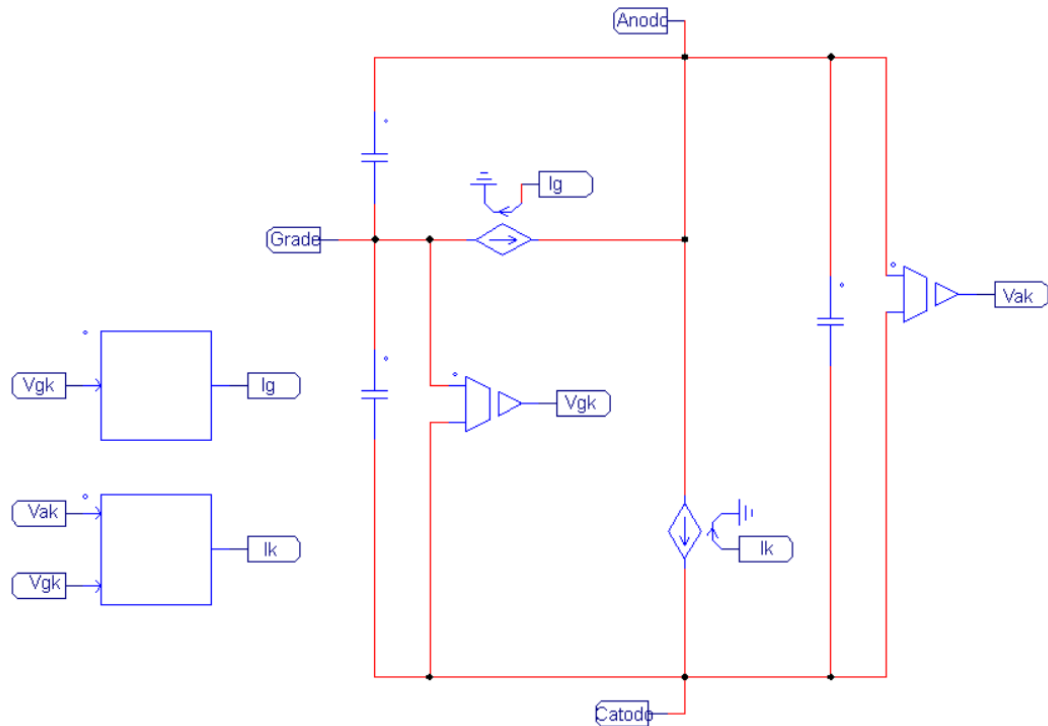
Tabela 1 – Parâmetros do modelo de Dempwolf para três diferentes válvulas 12AX7.

	RSD-1	RSD-2	EHX-1
G	2,242E-3	2,173E-3	1,371E-3
μ	103,2	100,2	86,9
γ	1,26	1,28	1,349
C	3,40	3,19	4,56
G_g	6,177E-4	5,911E-4	3,263E-4
ξ	1,314	1,358	1,156
C_g	9,901	11,76	11,99
$I_{g0}(A)$	8,014E-8	4,527E-8	3,917E-8

Fonte: adaptado de (DEMPWOLF; ZÖLZER, 2011).

Para a simulação do modelo, optou-se por utilizar os parâmetros da coluna EHX-1, pois as válvulas representadas pela abreviação RSD são, conforme o autor, mais antigas, e as válvulas utilizadas na construção do protótipo do pré-amplificador valvulado deste trabalho são novas.

As capacitâncias parasitas observadas por Dempwolf e Zölzer (2011) são: para anodo-catodo, 0,9 pF; para grade-catodo, 2,3 pF; e para anodo-grade, 2,4 pF. O modelo da válvula foi construído no software *PowerSIM*® e pode ser observado conforme a Figura 20. As expressões (5.1) e (5.2) foram implementadas por meio de blocos de programação na linguagem C, cujas entradas são as tensões anodo-catodo e grade-catodo. O resultado destas equações é aplicado à fontes dependentes de corrente. As capacitâncias parasitas foram inclusas para assimilar de forma mais fidedigna a resposta do modelo simulado ao modelo matemático usado como base.

Figura 20 – Modelo de Dempwolf de uma válvula 12AX7 no software *PowerSIM*®.

Fonte: o próprio autor, 2019.

Para que fosse possível simular este modelo precisou-se utilizar um passo de cálculo de $1 \cdot 10^{-08}$ s, devido ao mesmo não ter sido otimizado em questões de processamento computacional. Tal passo de cálculo é bastante pequeno e resulta em uma simulação demorada. No entanto, a utilização de uma simulação, mesmo que não otimizada, é mais conveniente do que testar a capacidade das redes neurais com base no protótipo real.

O passo de cálculo pequeno resulta em sinais com muitos dados e assim, o treinamento das redes neurais leva muito mais tempo do que seria necessário. Uma solução encontrada para tal é de reamostrar o sinal para um passo de cálculo inferior inferior ¹ ao utilizado no software de simulação. Optou-se, portanto, por reamostrar o sinal com uma frequência de cinco a dez vezes maior que a frequência do sinal, mantendo sua integridade e também respeitando a relação de Nyquist.

5.2 PROJETO DO PRÉ-AMPLIFICADOR

Para o início do projeto, foram definidos que seriam utilizados três estágios de ganho valvulados, e um estágio de *tonestack* ², totalizando quatro estágios valvulados. Para isto, são necessárias duas válvulas com dois triodos cada, sendo que cada triodo consistirá no "núcleo" de

¹ A reamostragem do sinal pode ser realizada no software *Matlab*® por meio da função *resample*.

² *Tonestack* é o conjunto de filtros de agudos, médios e graves em que os valores de frequências de corte destes podem ser modificados pelo usuário por meio de potenciômetros.

cada estágio.

Durante a explicação dos circuitos, pode-se perceber que muitos valores foram escolhidos visando maximizar o ganho e/ou diminuir a atenuação. Estas escolhas foram feitas pois somente duas válvulas triodo 12AX7 foram utilizadas no projeto, totalizando quatro estágios, sendo três de ganho. A necessidade de um ganho consideravelmente alto se dá pelo aumento da amplitude das harmônicas, principalmente da harmônica de segunda ordem, esta que é ocasionada pela não linearidade presente nas válvulas e, conseqüentemente, proporcional ao ganho.

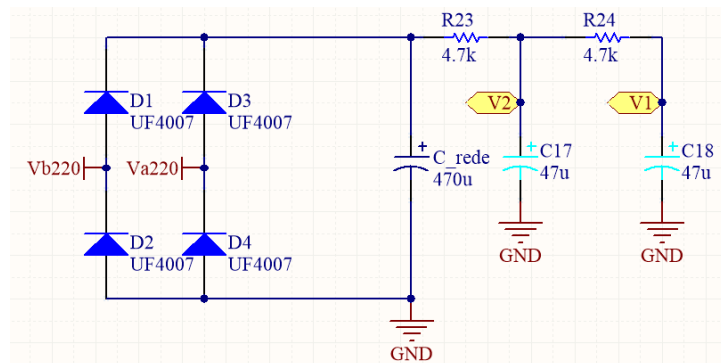
5.2.1 Alimentação

O circuito de alimentação do pré-amplificador valvulado inicia com dois transformadores 220/12 V *back-to-back*. Tal forma de ligação dos transformadores garante tensões de 12 e 220 V isoladas, garantindo a isolamento do usuário quanto à rede.

A tensão do anodo das válvulas geralmente situa-se entre 200 e 500 V, com corrente contínua. Nos *datasheets*, os valores nominais dos triodos são obtidos com uma alimentação no anodo de 250 V. Para este projeto, foi escolhida uma tensão de aproximadamente 310 V, devido a facilidade de retificar a tensão da rede e filtrá-la para alimentar os anodos. Ao ligar o circuito de alimentação nas válvulas, espera-se uma queda de tensão em torno de 20 - 30 V devido ao circuito resistivo formado pelo resistor do anodo e pela resistência interna do triodo.

A tensão isolada de 220 V é retificada por diodos UF4007, e filtrada por um capacitor eletrolítico de 470 μF . A qualidade desta forma de onda é então melhorada com dois filtros passa-baixas de primeira ordem em série, cada um contendo um resistor de 4,7 k Ω e um capacitor de 47 μF , com uma frequência de corte de 0,72 Hz. Tal baixa frequência de corte justifica-se pela necessidade de uma alimentação quase que puramente DC no anodo das válvulas triodos. Caso esta tensão alterne demais, bastante ruído será adicionado ao sistema. Tal parte do circuito de alimentação pode ser observado de acordo com a Figura 21.

Figura 21 – Alimentação dos anodos.

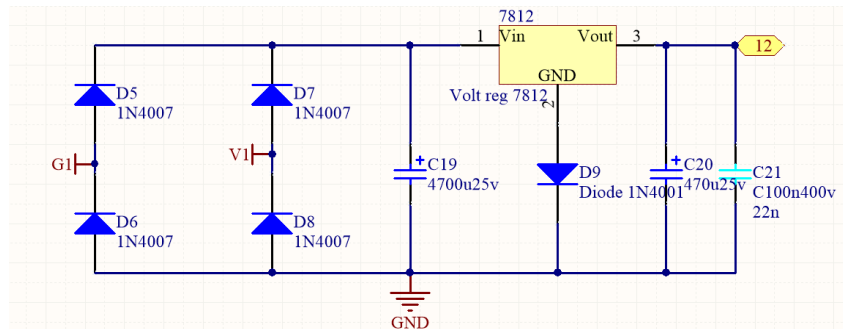


Fonte: o próprio autor, 2019.

Vale ressaltar que os estágios de ganho iniciais são mais sensíveis ao ruído, devido a possuírem uma relação sinal ruído reduzida. Portanto, a tensão mais filtrada e, conseqüentemente,

menor, está alimentando os anodos do primeiro e segundo triodo (primeira válvula). Para alimentar o filamento das válvulas, foi optado por utilizar uma alimentação de 12,3 V DC. Inicialmente, foi tentado utilizar 12 V AC, porém, havia muito ruído audível em 60 Hz. Portanto, decidiu-se por retificar e filtrar a tensão de 12 V alternada, de acordo com o circuito presente da Figura 22.

Figura 22 – Alimentação dos filamentos.



Fonte: o próprio autor, 2019.

A tensão de 12 V AC obtida no centro da ligação *back-to-back* dos transformadores é retificada por diodos 1N4007. Seu filtro é formado por um capacitor de $4700\ \mu F$, um regulador linear de 12 V com um diodo 1N4007 no seu pino central, para que possam ser obtidos os 12,3 V comumente utilizados na alimentação de filamentos de triodos, e finalmente, dois capacitores, um de $470\ \mu F$ e outro de $100\ nF$.

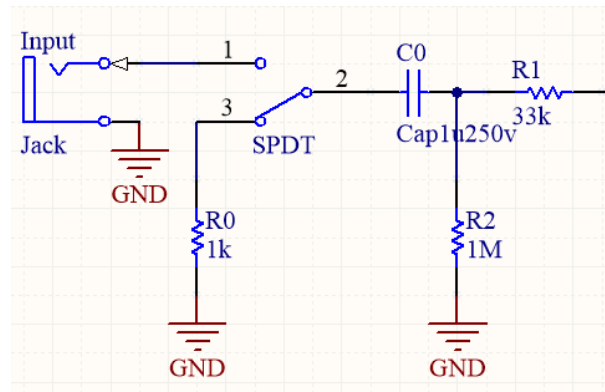
Esta tensão de 12,3 V DC alimenta, em série, os dois filamentos presentes em cada válvula. Tal alimentação também é utilizada para dois LEDs, um verde e um vermelho, cada um indicando o canal 1 e o canal 2, respectivamente, do pré-amplificador. A configuração de cada canal é explicada na subseção 5.2.6.1.

5.2.2 Entrada do pré-amplificador

A entrada do pré-amplificador pode ser observada conforme a Figura 23. Os sinais adentram o pré-amplificador por meio de um cabo P10 vindo diretamente da guitarra ou após um pedal de efeitos, conforme a necessidade do usuário. Uma chave SPDT seleciona entre o sinal de entrada ou um resistor (R0) aterrado de $1\ k\Omega$, servindo como *standby* para quando o pré-amplificador estiver ligado e não sendo tocado, evitando ruídos indesejáveis proveniente de interferências da guitarra e cabos.

O capacitor C0 evita níveis DC que possam provir de pequenos surtos nos captadores da guitarra (sejam estes ativos ou passivos) ou de pedais de efeitos utilizados anteriormente ao amplificador. Seu valor de capacitância não é relevante, desde que não seja pequeno demais ou grande demais. Um valor médio utilizado é de $1\ \mu F$.

Figura 23 – Entrada do pré-amplificador.



Fonte: o próprio autor, 2019.

Devido ao aquecimento da grade (*grid*) da válvula, alguns elétrons são emitidos, ocasionando em uma carga positiva da mesma, a não ser que haja um caminho de para recuperar a carga (negativa) perdida, o *pull down*. O resistor R2 é responsável por prover este caminho. Seu valor deve ser elevado para evitar o desvio do sinal proveniente da guitarra para o terra, porém, há um valor máximo informado nos *datasheets* das válvulas triodo, usualmente em torno de $2\text{ M}\Omega$. Assim, foi escolhido o valor de $1\text{ M}\Omega$.

O resistor R1 serve para evitar que possíveis surtos de corrente danifiquem as válvulas. Os valores comumente utilizados em amplificadores britânicos variam entre 33 e 68 $k\Omega$. Foi optado por utilizar o valor de 33 $k\Omega$ por atenuar menos o sinal de entrada.

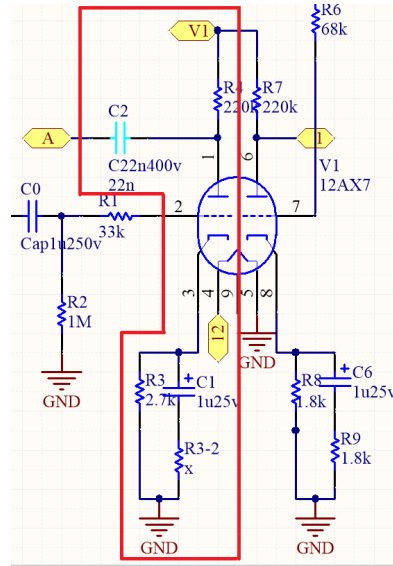
5.2.3 Primeiro estágio de ganho

O primeiro estágio de ganho pode ser observado, selecionado em vermelho, na Figura 24. O sinal adentra a válvula por meio da grade, pino 2, onde controlará o fluxo de elétrons entre anodo (pino 1) e catodo (pino 3). Para o resistor do anodo do primeiro estágio, foi escolhido 220 $k\Omega$. Assim, com a alimentação de 310 V, obtém-se uma corrente máxima para o triodo de 1,4 mA. Desenha-se então a reta de carga, conforme as curvas presentes no datasheet, como apresentado na Figura 25 em vermelho.

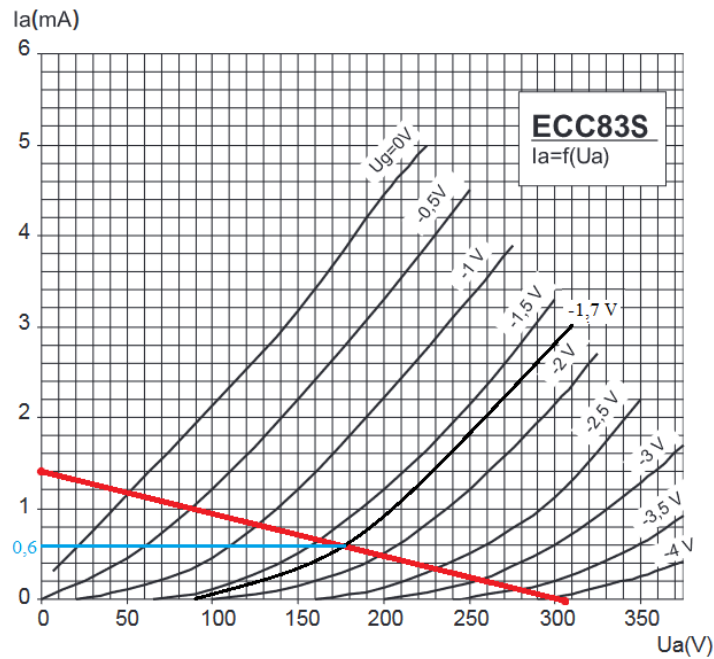
Há diversas formas de se polarizar um triodo. Neste projeto, será utilizada a polarização por catodo, consistindo em um ajuste entre o resistor em série com o catodo e o resistor em série com o anodo. É recomendável que o estágio inicial tenha um *bias* centralizado por possuir uma capacidade de *headroom*³ maior. Assim, define-se um ponto de operação em torno de -1,7 V. Considerando a reta de carga para o resistor de 220 $k\Omega$, obtém-se uma corrente de aproximadamente 0,6 mA. Assim, o resistor do cátodo será de $1,7\text{ V} / 0,6\text{ mA}$, resultando em 2,83 $k\Omega$. O resistor comercial mais próximo é de 2,7 $k\Omega$, portanto este será utilizado.

³ *Headroom* é um termo que refere-se a diferença entre a máxima potência de um amplificador de áudio e a potência média do sinal de saída para determinado volume.

Figura 24 – Primeiro estágio de ganho.



Fonte: o próprio autor, 2019.

Figura 25 – Reta de carga para o resistor de 220 k Ω .

Fonte: adaptado de (JJ ELECTRONIC,).

Um capacitor eletrolítico é adicionado em paralelo com o resistor do catodo para manter a tensão do mesmo mais constante, evitando *feedback*⁴ e permitindo que todo o ganho seja utilizado. O valor do capacitor deve ter um valor elevado para que o ganho mantenha-se máximo para todas as frequências audíveis. O resistor de valor "x" situado em série com o capacitor C1

⁴ O *feedback* da corrente do catodo é causado quando a tensão no catodo da válvula é modulada juntamente com o sinal de áudio na grade, resultando em ganho e distorção não linear reduzidos.

não foi utilizado, portanto, um *jumper* foi soldado no local do mesmo.

O ganho de um estágio valvulado triodo para qualquer frequência é dado por

$$A = \frac{-\mu R_a}{R_a + r_a} \cdot \sqrt{\frac{1 + (2\pi f R_k C_k)^2}{[1 + \frac{R_k(\mu+1)}{R_a + r_a}]^2 + (2\pi f R_k C_k)^2}}, \quad (5.3)$$

Onde R_a é o resistor do anodo, R_k é o resistor do catodo em paralelo com o capacitor C_k , μ é o fator de amplificação, dado pela variação de tensão do anodo dividida pela variação da tensão grade-catodo da válvula, r_a é a resistência do anodo, calculada pela variação de tensão no anodo dividida pela variação de corrente no mesmo. f pode ser qualquer frequência, mas para realizar o cálculo do ganho, é utilizada a frequência *half boost*, que refere-se à frequência em que o ganho está na metade entre ganho mínimo e ganho máximo, dada por

$$f_{(halfboost)} = \frac{1}{2\pi R_k C_k} \cdot \sqrt{1 + \frac{R_k(\mu + 1)}{2(R_a + r_a) + \frac{1}{2}R_k(\mu + 1)}}. \quad (5.4)$$

Assumindo uma variação de tensão grade-catodo de 1 V pico-a-pico, é obtida uma variação de tensão no anodo de aproximadamente 70 V. Assim, é possível calcular um fator de amplificação μ de 70. A variação de corrente no anodo obtida para tal variação de tensão grade-catodo é de 0,36 mA, obtendo assim uma resistência do anodo r_a de aproximadamente 194,5 k Ω .

O ganho de tensão do primeiro estágio, portanto, de acordo com a relação 5.3, situa-se em -30,179⁵. Entre os estágios, encontram-se filtros e componentes de acoplamento. Esses circuitos são comentados nas subseções 5.2.7, 5.2.8 e 5.2.9.

5.2.4 Segundo estágio de ganho

O segundo estágio de ganho encontra-se destacado na Figura 26.

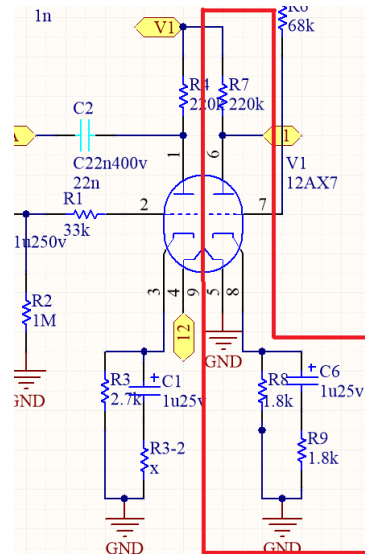
Os cálculos realizados no primeiro estágio repetem-se para todos os três estágios de ganho, portanto estes serão descritos mais diretamente, assim como certos cálculos minoritários serão omitidos. Caso o leitor tenha interesse em realizar os cálculos, deve-se seguir o procedimento realizado na Subseção 5.2.3.

Percebe-se que este estágio assimila-se muito ao primeiro. A maior diferença é dada pelo valor do resistor do catodo, assim como um resistor de mesmo valor em série com o capacitor do catodo, alterando a frequência *half boost*. Para o cálculo da mesma, basta considerar o dobro da resistência do catodo. Este resistor não interfere no cálculo do ganho.

Considerando a mesma reta de carga presente na Figura 25, pois o resistor do anodo tem o mesmo valor, pode-se escolher um ponto de operação para este estágio. Supondo uma curva

⁵ O valor é negativo devido à inversão de fase realizada pela válvula operando como catodo comum. O ouvido humano não é capaz de perceber esta diferença, portanto o ganho prático é equivalente ao módulo desse valor.

Figura 26 – Segundo estágio de ganho.



Fonte: o próprio autor, 2019.

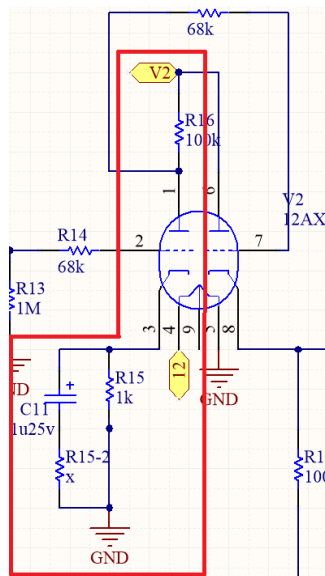
de carga de -1,3 V (um pouco mais quente ⁶ que o *bias* anterior), o ponto de intersecção entre esta e a reta de carga resultam em uma corrente de aproximadamente 0.78 mA. Tais valores sugerem um resistor do catodo de 1,67 k Ω . Por disponibilidade, o resistor utilizado foi de 1,8 k Ω . Inicialmente foi utilizado um *bias* mais frio, com um resistor na faixa de 22 k Ω , porém, um ruído bastante agudo tomava conta da sonoridade a partir de um certo ganho. Tal fenômeno não era esperado. Este foi resolvido mudando o *bias* do segundo estágio para um mais quente, assim como modificações foram feitas nos filtros entre os estágios de ganho. Para uma variação de tensão grade-catodo de 1 V, têm-se um fator de amplificação de 83, e uma resistência do anodo de aproximadamente 230,5 k Ω , resultando em uma frequência *half boost* de 50,161 Hz e em um ganho de tensão de -32,111.

5.2.5 Terceiro estágio de ganho

O terceiro estágio de ganho, selecionado em vermelho, pode ser visto na Figura 27. Neste caso, um resistor de 100 k Ω foi utilizado no anodo do triodo, visando um estágio com um *bias* mais quente e menor ganho, para ser utilizado no canal mais limpo como estágio característico para a distorção. A Figura 28 mostra a nova reta de carga, para o resistor de 100 k Ω . O fator de amplificação obtido para o terceiro estágio é de 68, com uma resistência do anodo de aproximadamente 104,5 k Ω . A frequência *half boost* situa-se em 90,332 Hz, enquanto o ganho de tensão situa-se em -26,305.

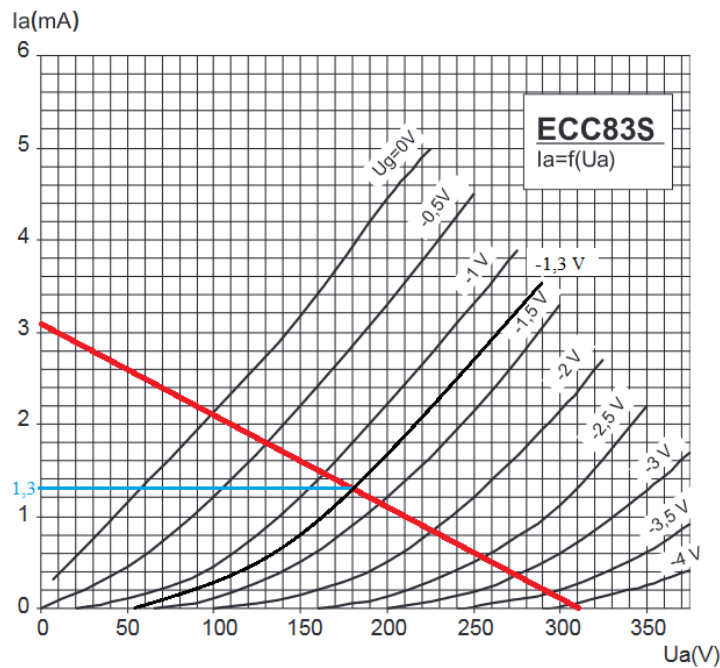
⁶ Os termos quente e frio utilizados para o *bias* das válvulas referem-se à características sonoras referentes ao tipo de distorção gerada, dependendo da posição do ponto de operação nas curvas de carga. Caso situado à esquerda da curva de -1,5 V, diz-se que o *bias* é mais quente, enquanto que para pontos após a curva de -1,5 V, o *bias* é dito mais frio. O "calor" presente na sonoridade faz referência, principalmente, à quantidade de graves e ao tipo de agudo. Caso o som seja quente, os graves estarão mais presentes e o agudo será mais "limpo", enquanto que no caso frio, os graves serão reduzidos e o agudo será mais "sujo", como se uma leve distorção fosse adicionada

Figura 27 – Terceiro estágio de ganho.



Fonte: o próprio autor, 2019.

Figura 28 – Reta de carga para o resistor de 100 kΩ.



Fonte: adaptado de (JJ ELECTRONIC,).

5.2.6 Tonestack e saída

A posição do *tonestack* foi definida como sendo o último estágio. Caso este fosse posicionado anteriormente a algum estágio de ganho, a modificação dos agudos, médios e graves poderia ser atenuada (no caso da presença de filtros nos estágios de ganho) ou amplificada pelo

às frequências mais altas.

5.2.6.1 Seleção de canais

Dois canais podem ser utilizados no pré-amplificador projetado. O primeiro canal consiste na seguinte cadeia de circuitos: entrada, estágio de ganho 1, circuito entre estágios 1 e 3, estágio de ganho 3, *tonestack*, saída. Enquanto isso, o segundo canal consiste em: entrada, estágio de ganho 1, circuito entre estágios 1 e 2, estágio de ganho 2, circuito entre estágios 2 e 3, estágio de ganho 3, *tonestack* e saída.

Percebe-se que a principal diferença entre os dois canais se dá pelo *bypass* do segundo estágio de ganho no primeiro canal, ocasionando em um som que pode variar de um som limpo até uma sonoridade razoavelmente distorcida. O segundo canal, por sua vez, varia de uma sonoridade razoavelmente distorcida à uma sonoridade com um quantidade de distorção elevada.

No entanto, pré-amplificadores de alto ganho costumam consistir em uma maior quantidade de estágios de ganho (comumente cinco), com uma alternância de *bias* entre quente e frio. Como só duas válvulas (quatro triodos) foram utilizados neste projeto, foi optado por manter um *bias* levemente quente, pois o *bias* mais frio torna-se mais interessante somente para ganhos mais elevados⁷.

5.2.6.2 Saída do pré-amplificador

Na saída encontram-se um potenciômetro para atenuar o volume, um capacitor de acoplamento de 22 ηF , para eliminar possíveis níveis DC presentes no sinal, e uma saída via cabo P10 para ser direcionada à jusante de um amplificador de potência classe D.

Recomenda-se, para futuras modificações, a inclusão de um amplificador operacional TL08X entre o potenciômetro TREBLE e o divisor resistivo, operando como *buffer*, para evitar que o potenciômetro de volume afete os valores das frequências de corte do *tonestack*, principalmente para o filtro de agudos, assim como o potenciômetro de agudos afete o volume.

O resistor R20, deve ser escolhido de forma que o valor máximo na saída deste pré-amplificador seja 10-20 % menor que o valor de pico da portadora do modulador PWM presente em um amplificador de potência classe D. É desejável que estes valores sejam calculados e medidos com base nos valores práticos de tensão de saída, para que haja maior precisão.

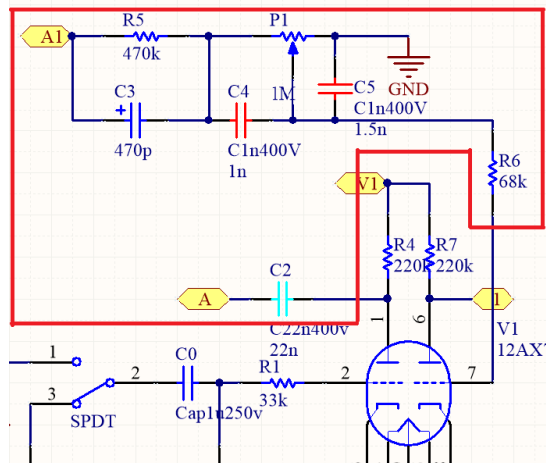
5.2.7 Circuito entre estágios 1 e 2

O circuito presente entre o primeiro e o segundo estágio de ganho está selecionado em vermelho na Figura 30.

Inicialmente, o sinal sai do primeiro estágio e passa por um capacitor de acoplamento de 22 ηF , para eliminar possíveis níveis DC. Após isso, o sinal passa por um filtro composto pelo resistor R5, capacitores C3, C4 e C5, e potenciômetro P1. Neste filtro, o capacitor C1 e

⁷ Como qualidade da distorção e qualidade sonora de amplificadores de guitarra elétrica são fatores muito subjetivos, as opiniões aqui expressas representam somente a opinião do autor deste relatório e não devem ser tomadas como verdade absoluta.

Figura 30 – Circuito entre o estágio 1 e 2.



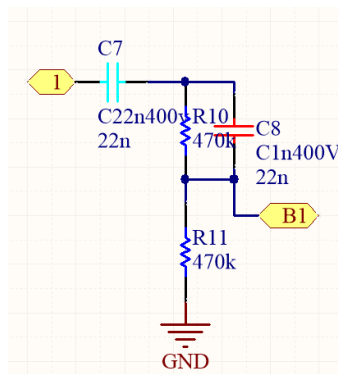
Fonte: o próprio autor, 2019.

C2 servem como *treble boost*, ou seja, passa altas. No entanto, na prática foi percebido um excesso de agudos. Tal excesso foi solucionado modificando o valor dos capacitores C3 e C4, empiricamente, e incluindo o capacitor C5. O resistor R6 serve para evitar que surtos de corrente danifiquem a grade do triodo do segundo estágio. O potenciômetro P1 é usado como atenuador de ganho. Este potenciômetro só é funcional no segundo canal do pré-amplificador, sendo responsável pelo controle de ganho.

5.2.8 Circuito entre estágios 2 e 3

O circuito entre o segundo e terceiro estágio está marcado em vermelho na Figura 31. Este circuito consiste em um capacitor de acoplamento de 22 nF e de um filtro passa altas. Os resistores atuam como divisor resistivo de tensão para que o sinal entre os estágios seja atenuado, evitando um ganho excessivo e possíveis ruídos.

Figura 31 – Circuito entre o estágio 2 e 3.

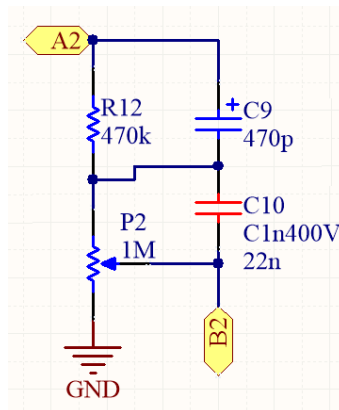


Fonte: o próprio autor, 2019.

5.2.9 Circuito entre estágios 1 e 3

Caso o canal 1 seja selecionado, a saída do primeiro estágio não passará pelo segundo estágio, esta será desviada por meio de um circuito, conforme mostrado na Figura 32, para o terceiro estágio.

Figura 32 – Circuito entre o estágio 1 e 3.



Fonte: o próprio autor, 2019.

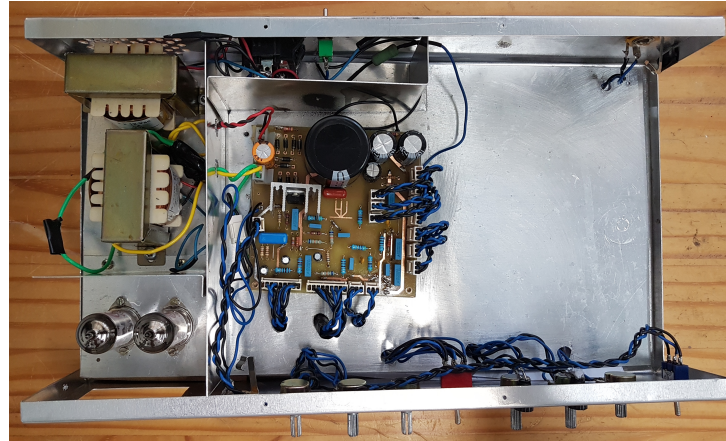
O capacitor de acoplamento de 22 nF também está presente, porém não foi mostrado na Figura 32. O filtro é muito semelhante ao filtro entre os estágios 1 e 2, porém este não possui o capacitor desviando altas frequências para o terra. No entanto, os valores de C9 e C10 foram modificados de forma empírica para alcançar a sonoridade desejada pelo autor. O potenciômetro P2 é responsável pelo controle de ganho quando o amplificador está selecionado para operar no primeiro canal.

5.3 RESULTADOS DO PROTÓTIPO

O pré-amplificador foi construído e montado em uma caixa de alumínio. A Figura 33 apresenta o protótipo em uma visão superior, com a caixa aberta. Nesta figura, pode-se perceber facilmente os transformadores (*back-to-back*), as duas válvulas triodo e a placa do pré-amplificador com os componentes soldados. Os *jacks* de entrada e saída, assim como potenciômetros e *switches*, encontram-se presos ao painel frontal e traseiro. Um espaço extra foi deixado na caixa de alumínio para uma futura inserção de um amplificador classe D. A esquemática completa encontra-se na Figura 39, enquanto que o *layout* completo da placa pode ser observado na Figura 40.

Para um sinal de entrada de 1 V pico-a-pico, com frequência de 1 kHz, o sinal de saída para os estágios de ganho 1, 2 e 3, para o canal 2 (de maior ganho), com o potenciômetro de ganho em aproximadamente 100 %, é apresentado conforme as Figuras 34, 35 e 36. Os resultados foram documentados somente para o canal de alto ganho (segundo canal), pois este é o principal

Figura 33 – Protótipo do pré-amplificador valvulado.

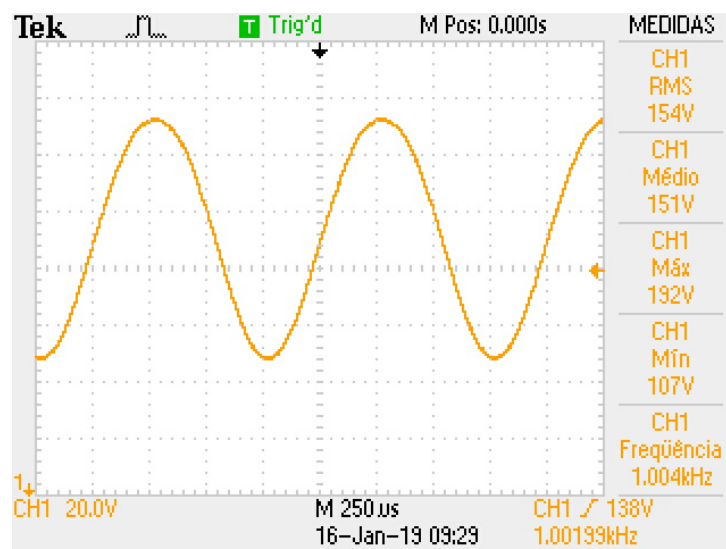


Fonte: o próprio autor, 2019.

canal de interesse do trabalho, devido a elevada presença de componentes harmônicas pares no sinal ocasionado por sua distorção.

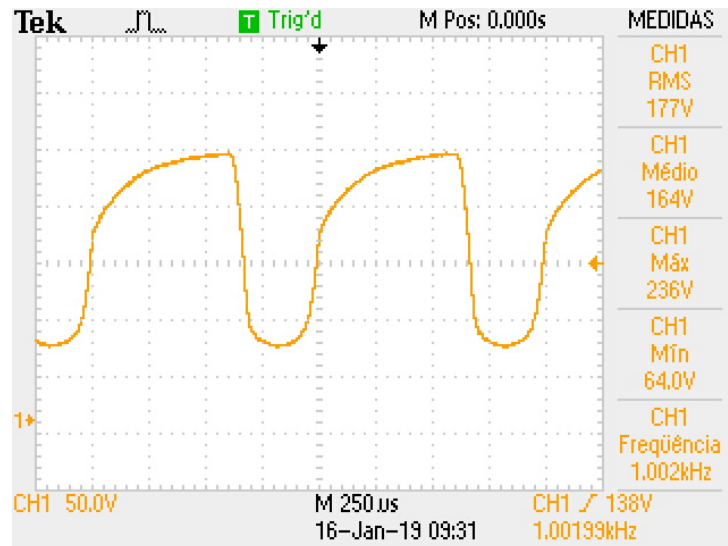
Finalmente, o sinal de saída do pré-amplificador para o canal 2, com potenciômetro de ganho em 100 % e os potenciômetros de médio, grave e agudo em 50 %, é dado conforme a Figura 37. Realizando a FFT do sinal por meio do *software* Matlab, obteve-se as harmônicas conforme a Figura 38. Percebe-se a dominância da segunda e quarta harmônica sobre as demais, assim como uma baixa amplitude da terceira harmônica. A obtenção de tais componentes harmônicas valida o projeto, pois este era o objetivo inicial da construção deste pré-amplificador valvulado. Assim, afirma-se que o protótipo produz uma distorção bastante não linear, tornando-se um bom objeto de estudos para emulações e identificação de sistemas não lineares.

Figura 34 – Sinal de saída do primeiro estágio.



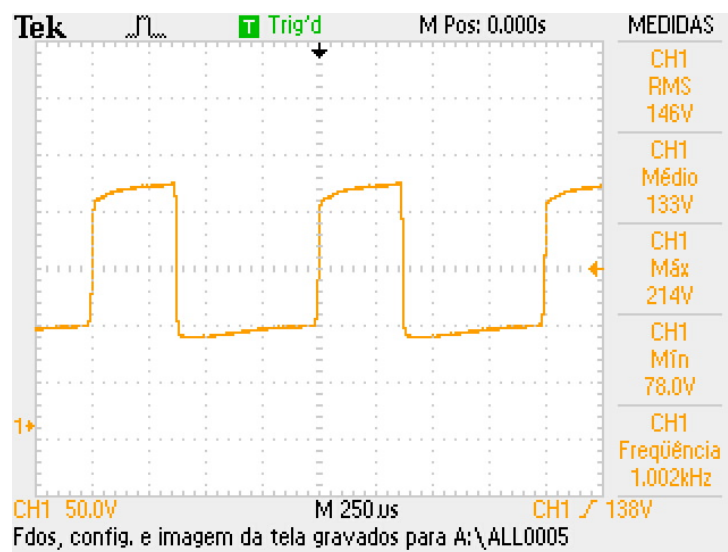
Fonte: o próprio autor, 2019.

Figura 35 – Sinal de saída do segundo estágio.



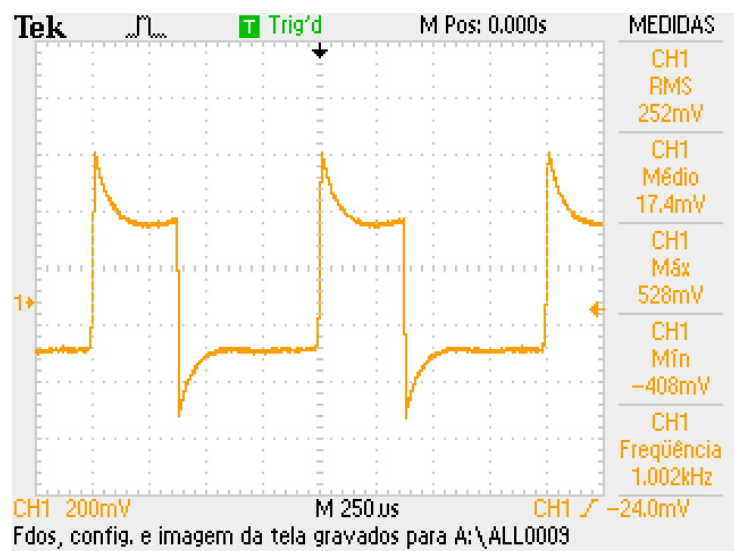
Fonte: o próprio autor, 2019.

Figura 36 – Sinal de saída do terceiro estágio.



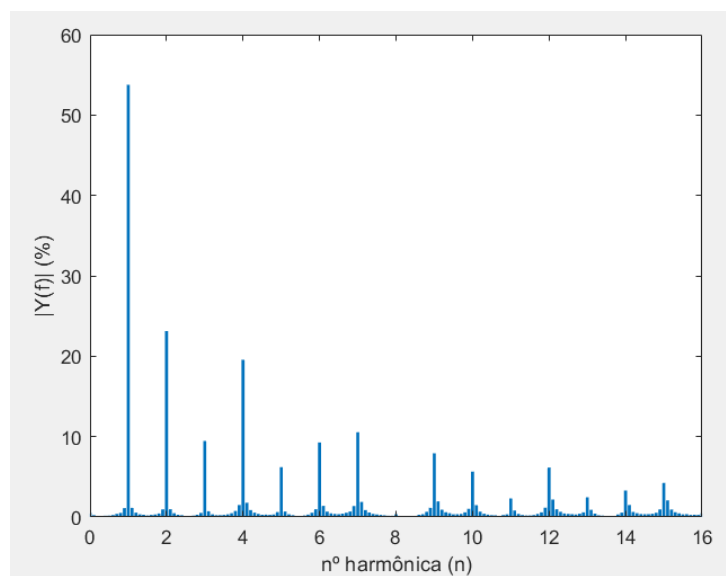
Fonte: o próprio autor, 2019.

Figura 37 – Sinal de saída do pré-amplificador valvulado.



Fonte: o próprio autor, 2019.

Figura 38 – FFT do sinal de saída do pré-amplificador valvulado.



Fonte: o próprio autor, 2019.

Fonte: o próprio autor, 2019.

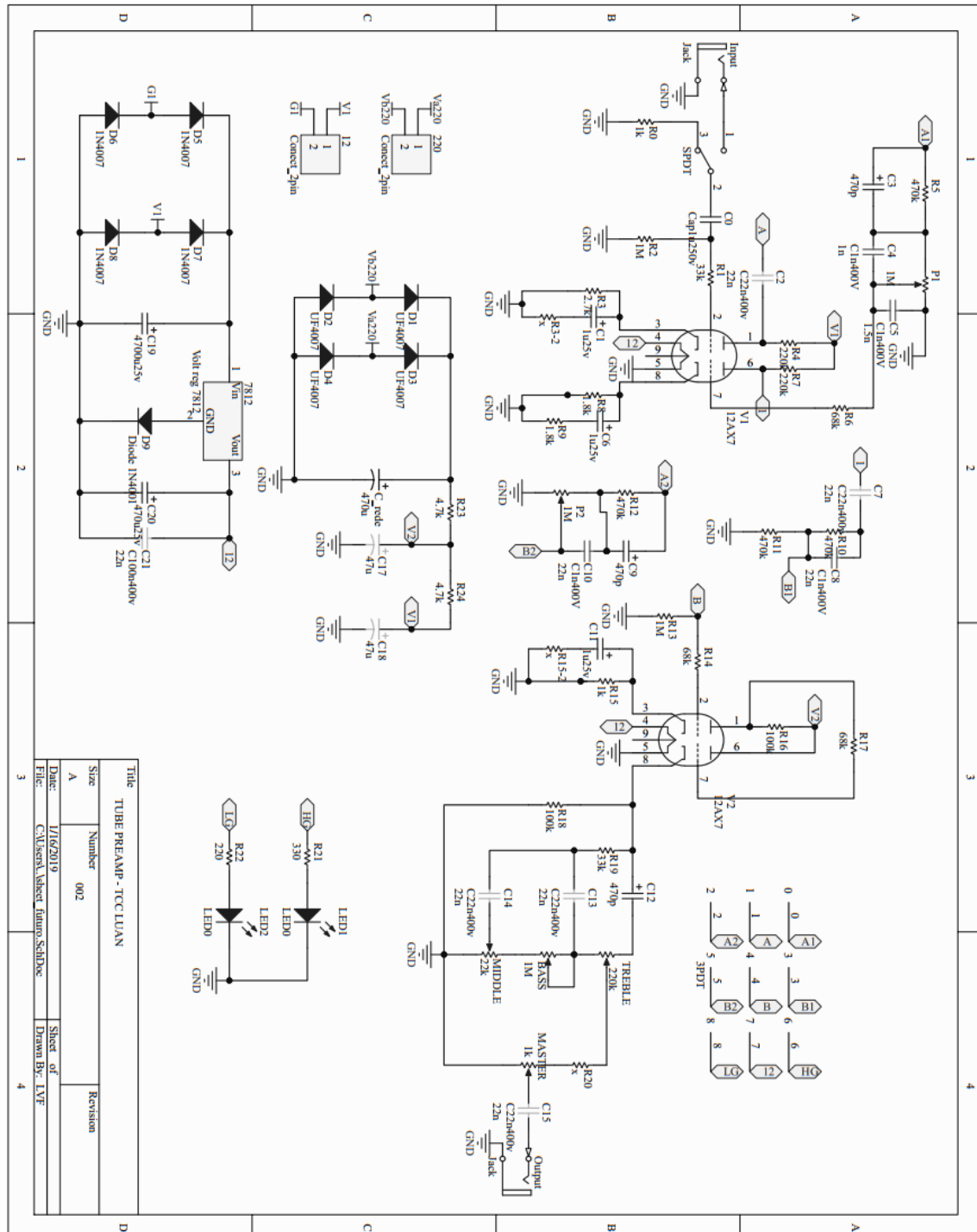
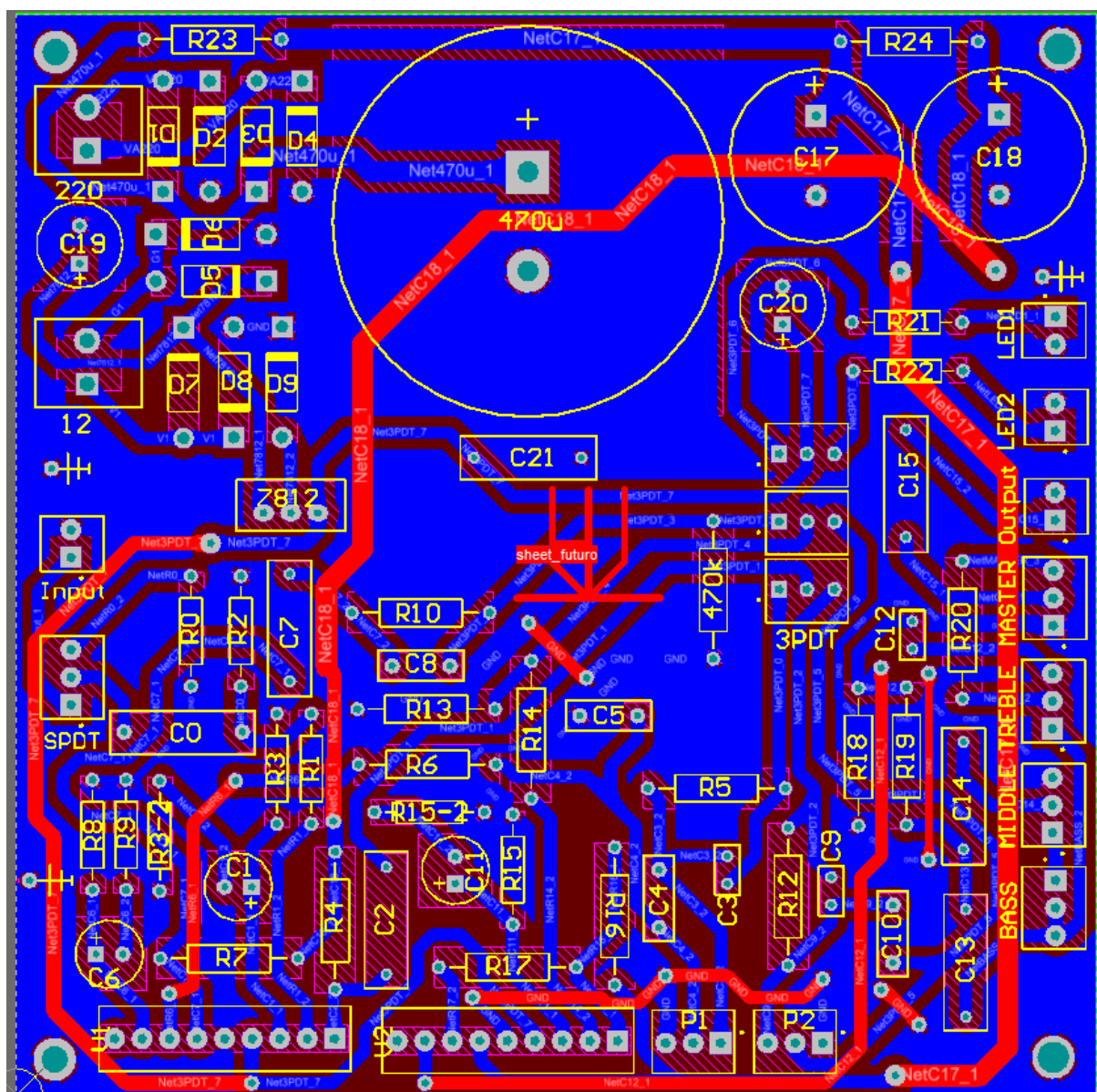


Figura 40 – Layout da PCB do pré-amplificador valvulado.



Fonte: o próprio autor, 2019.

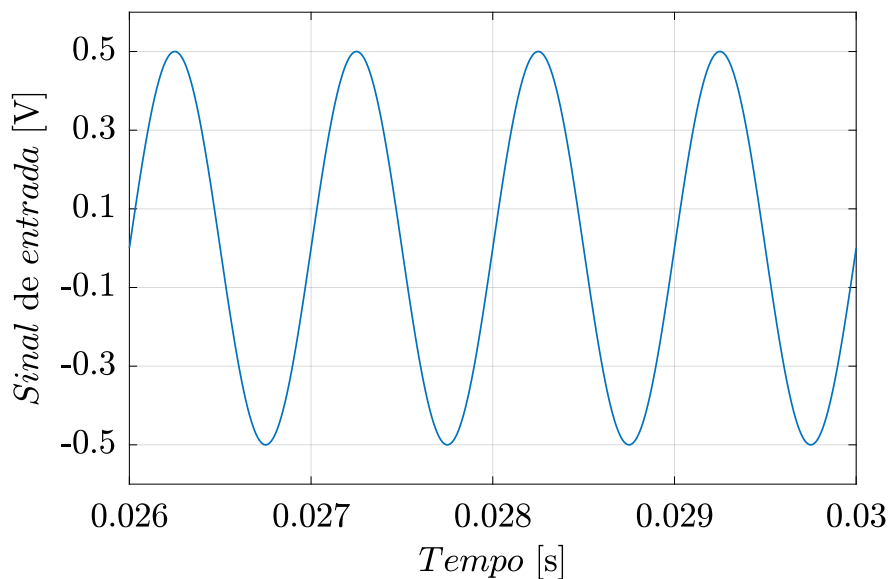
5.4 SIMULAÇÃO DO PRÉ-AMPLIFICADOR

O projeto do pré-amplificador valvulado foi realizado com base em (BLENLOWE, 2009) e foi exposto na Seção 5.2. Conforme este projeto e utilizando o modelo descrito na Subseção 5.1.1, o pré-amplificador valvulado foi elaborado no software *PowerSIM*® para uma situação de baixo ganho - de acordo com o canal 1 do projeto - e uma situação de alto ganho - equivalente ao canal 2 do projeto. Seus respectivos circuitos implementados no *PowerSIM*® são mostrados nos Apêndices A e B.

Cada canal é analisado inicialmente aplicando-se um sinal senoidal de 1 kHz de frequência e 1 V de tensão pico-a-pico (Figura 41) na entrada do pré-amplificador. É fixado um único ganho para cada canal por questões de simplicidade - no canal de alto ganho, fixa-se ganho máximo, enquanto que no canal de baixo ganho, utiliza-se um ganho próximo ao mínimo, visando a obtenção de um sinal com menos componentes harmônicas e assim, podendo obter uma diferença considerável na emulação dos dois canais.

Neste capítulo, o termo *tonestack* é mencionado diversas vezes. *Tonestack* compreende o filtro linear de graves, médios e agudos composto por capacitores, resistores e potenciômetros que situa-se após o último estágio (de ganho unitário) do pré-amplificador valvulado, em ambos os canais. O *tonestack* é abordado mais profundamente na Subseção 5.2.6 do projeto do pré-amplificador valvulado disposto no Apêndice 5.2.6.

Figura 41 – Sinal de entrada do pré-amplificador simulado.

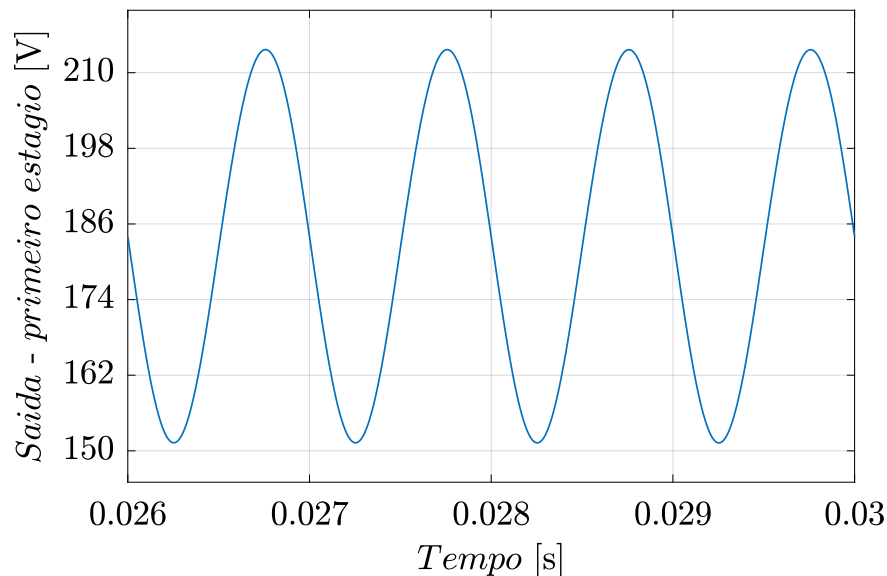


Fonte: o próprio autor, 2019.

5.4.1 O canal de baixo ganho

Aplicando o sinal de entrada da Figura 41 no pré-amplificador configurado para o canal 1 (baixo ganho), obtém-se a forma de onda na saída do primeiro estágio de ganho (anodo) conforme a Figura 42. Percebe-se que o sinal foi invertido e amplificado. Sua forma de onda é quase uma senoide perfeita, visto que a intenção do primeiro estágio de ganho não é distorcer, e sim amplificar o sinal para que os próximos estágios de ganho distorçam-no. O nível DC presente na forma de onda se dá pelo anodo ser alimentado por uma fonte de tensão DC de 311 V em série com um resistor de 220 k Ω . Percebe-se que um nível DC semelhante estará presente em todas as formas de onda de saídas de estágios de ganho, pois estes são medidos antes dos capacitores e filtros de acoplamento.

Figura 42 – Saída do primeiro estágio de ganho para o canal 1.



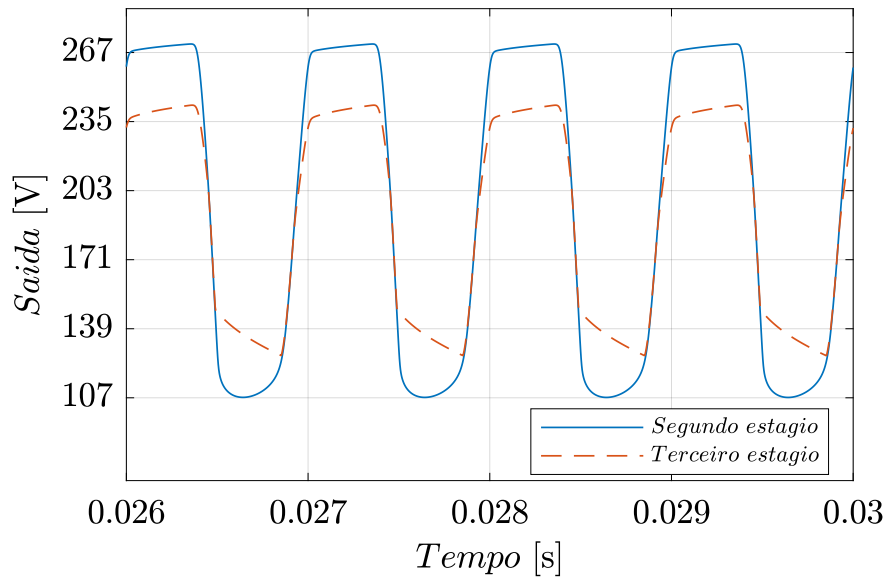
Fonte: o próprio autor, 2019.

Entre o primeiro e o segundo estágio de ganho, situam-se filtros de graves e agudos e um potenciômetro de ganho logarítmico, este que foi posicionado em 10 % de sua resistência total, para que fosse obtido um sinal final com baixo a médio ganho. A forma de onda da saída do segundo triodo, no anodo, pode ser observada conforme a linha contínua, em azul, da Figura 43. O sinal teve sua fase invertida e foi consideravelmente mais distorcido.

Após o segundo estágio de ganho, têm-se um terceiro estágio cuja função é de aumentar a potência do sinal para que esta não seja completamente dissipada quando passar pelo *tonestack*. A forma de onda de saída deste estágio (linha tracejada, em laranja, da Figura 43) é obtida no catodo. O ganho do triodo operando nesta forma é unitário e a fase do sinal se mantém. A forma do sinal do terceiro estágio é principalmente alterada, da sua forma original presente na saída do segundo estágio, pela presença do *tonestack* a jusante. Retirando estes filtros do circuito,

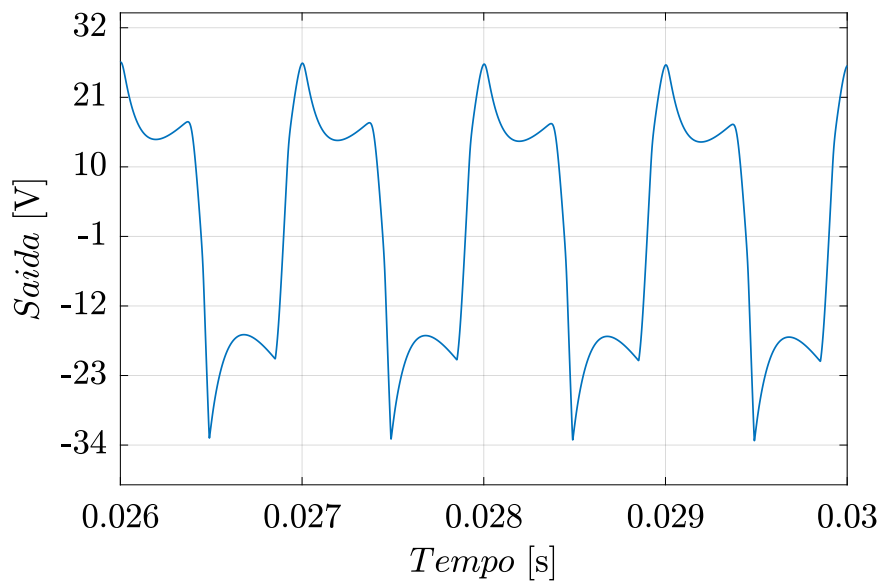
obtem-se formas de onda basicamente idênticas, visto que o ganho do último triodo é unitário e a fase do sinal não é invertida.

Figura 43 – Saída do segundo e terceiro estágios de ganho para o canal 1.



Fonte: o próprio autor, 2019.

Figura 44 – Saída do pré-amplificador para o canal 1.



Fonte: o próprio autor, 2019.

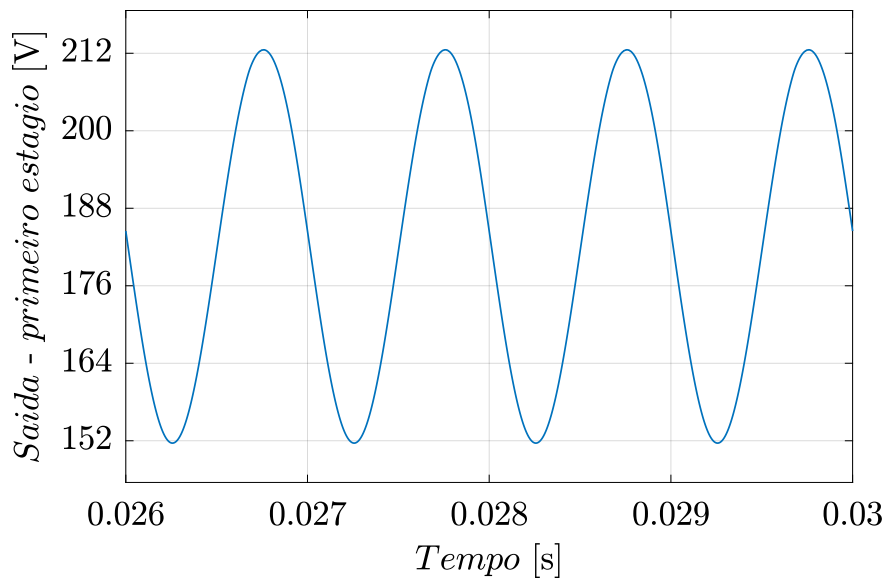
Por fim, o sinal de saída do pré-amplificador para o canal de baixo ganho é dado conforme a Figura 44. A fase do sinal se manteve e este foi filtrado pelo *tonestack* e pelo capacitor de

acoplamento da saída, eliminando o nível de tensão em corrente contínua ⁸. Os potenciômetros do *tonestack* foram posicionados aproximadamente em 50 % de sua resistência total. Percebe-se que a presença dos filtros modificou consideravelmente a parte superior e inferior da forma de onda da saída, porém mantendo uma característica de simetria de meia onda, o que era esperado visto que os filtros presentes no *tonestack* são lineares.

5.4.2 O canal de alto ganho

Ao aplicar o sinal de entrada observado na Figura 41 no pré-amplificador de alto ganho (canal 2), a forma de onda obtida no anodo do primeiro estágio de ganho é dada conforme a Figura 45. Percebe-se que a forma de onda foi invertida e amplificada.

Figura 45 – Saída do primeiro estágio de ganho para o canal 2.

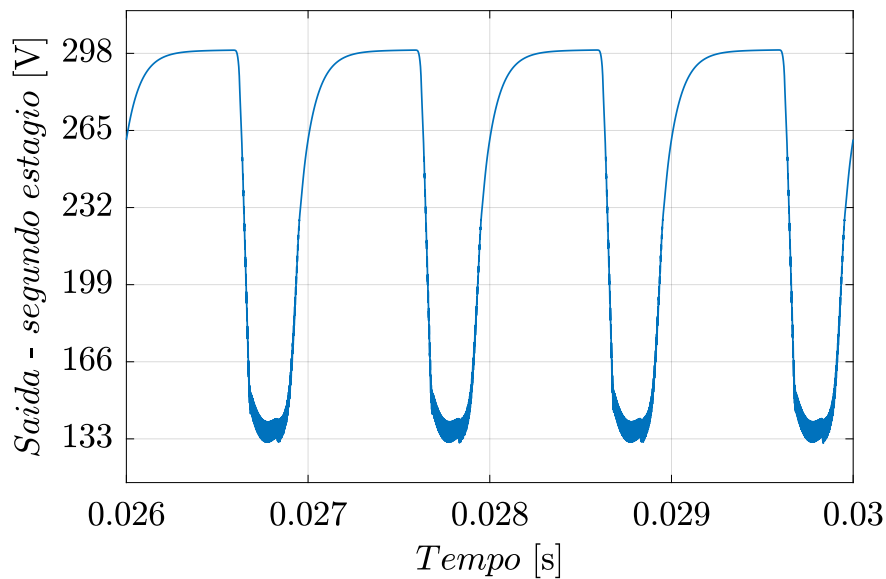


Fonte: o próprio autor, 2019.

A saída do segundo estágio de ganho é dada pela Figura 46 e faz referência ao anodo do segundo triodo do circuito. É facilmente perceptível que o sinal está mais distorcido, invertido e com um formato característico da distorção das válvulas - altamente não linear. O segundo estágio é o triodo com mais ganho do circuito e está presente somente no canal 2, como pode ser observado no projeto exposto no Apêndice ???. Entre o primeiro e o segundo estágio, nos filtros de acoplamento, está presente o potenciômetro de ganho. Este, por sua vez, está posicionado para obter máximo ganho.

⁸ Do inglês, *direct current* (DC).

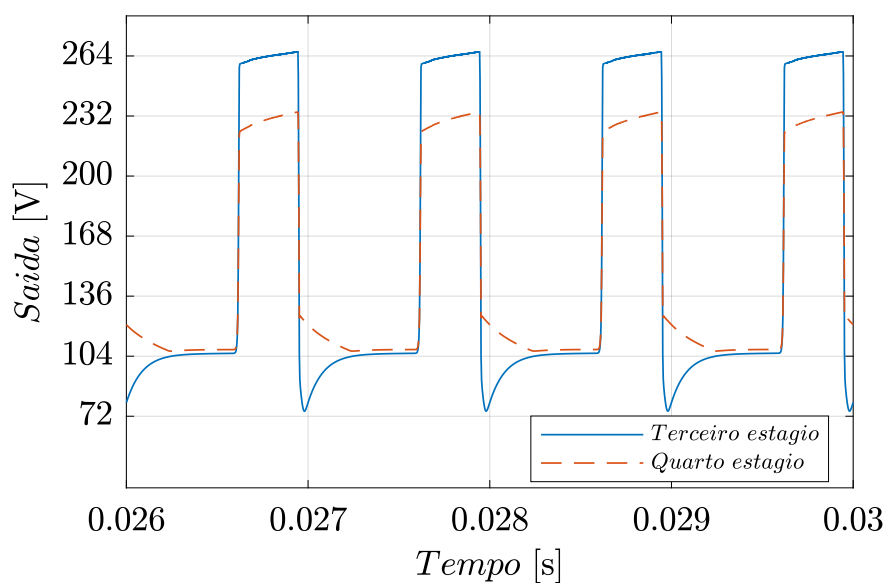
Figura 46 – Saída do segundo estágio de ganho para o canal 2.



Fonte: o próprio autor, 2019.

O terceiro estágio compreende mais um estágio comum de ganho, onde sua saída (no anodo) pode ser observada conforme a forma de onda contínua, em azul, na Figura 47. Pode-se verificar que o sinal foi invertido e o ganho deste estágio aproximou a forma de onda de uma forma retangular.

Figura 47 – Saída do terceiro e quarto estágios de ganho para o canal 2.



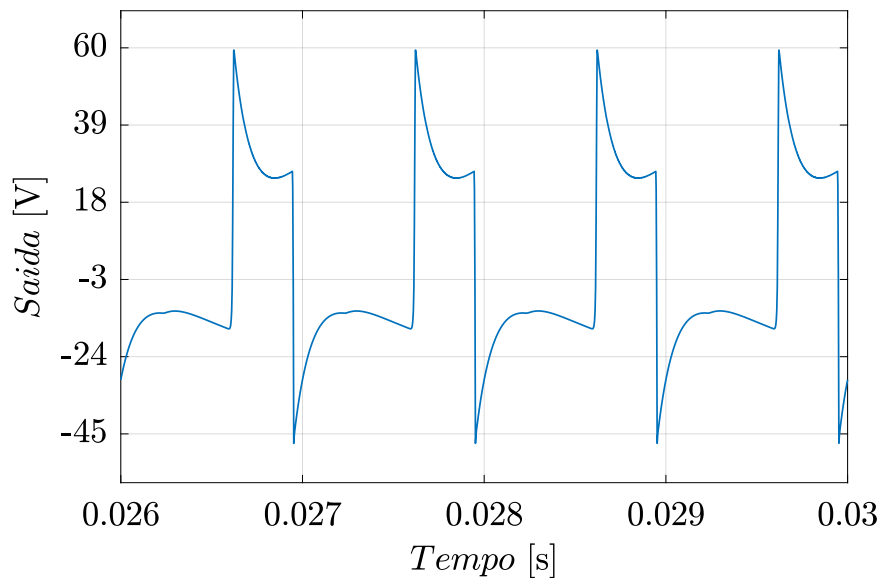
Fonte: o próprio autor, 2019.

Os filtros presentes entre cada estágio são responsáveis pela filtragem de graves e agudos durante a etapa de distorção. O resultado da inclusão de tais componentes pode ser observado nas partes superior e inferior, principalmente inferior, da forma de onda da Figura 47.

O quarto triodo tem como objetivo incrementar a potência do sinal para que este passe pelo *tonestack* mantendo sua integridade. Neste estágio, a saída do triodo é dada no catodo, portanto, o ganho é unitário e a fase do sinal é mantida. Este sinal pode ser observado na Figura 47 como a forma de onda tracejada, em laranja. A diferença entre a forma de onda da saída do quarto estágio para a saída do terceiro se dá, principalmente, pela presença do *tonestack* após o último estágio, mesmo a forma de onda sendo obtida anteriormente ao filtro, esta é modificada pela presença dos capacitores a jusante.

Finalmente, a forma de onda da saída do pré-amplificador é dada pela Figura 48. Esta forma de onda refere-se ao sinal obtido na saída do quarto estágio após passar pelo *tonestack* e ser filtrado por um capacitor de acoplamento. Percebe-se a forte influência dos filtros de graves, médios e agudos no sinal. Os potenciômetros destes filtros, lineares e logarítmicos, foram posicionados em aproximadamente 50 % do seu posicionamento para obter-se a forma de onda da Figura 48.

Figura 48 – Saída do pré-amplificador para o canal 2.



Fonte: o próprio autor, 2019.

6 DEFINIÇÃO DA ESTRUTURA DA REDE NEURAL

Para que se possa realizar a emulação do pré-amplificador valvulado, há de se definir a estrutura (número de camadas, neurônios, função de ativação e taxa de aprendizagem) da rede neural. Na literatura, não há um teste específico que defina a melhor estrutura de uma rede neural para uma situação qualquer, sendo comumente citada a experiência do projetista da rede como o principal fator para definir tais parâmetros.

Diante disso, objetiva-se neste capítulo definir um teste para ser utilizado como base para a escolha da estrutura das redes neurais recorrentes. Espera-se que esse teste possa ser utilizado em trabalhos futuros como base para definição de parâmetros, facilitando o projeto de redes neurais e assim, abstraindo a influência da experiência do projetista no resultado.

6.1 TESTES PARA DEFINIÇÃO DA ESTRUTURA DAS REDES NEURAIIS

Os testes aqui desenvolvidos baseiam-se em uma situação específica para definir os parâmetros da rede neural. No caso da identificação de sistemas, onde o sistema é identificado com base em sinais de entrada e saída, os testes são realizados para uma parcela de um sinal obtido pelo sistema em questão. Preferencialmente, deve-se utilizar conhecimentos prévios sobre o sistema para melhorar a qualidade dos parâmetros obtidos. Por exemplo, sabe-se que os sinais provenientes das notas de uma guitarra elétrica são senoidais, portanto aplica-se um sinal senoidal de frequência e amplitude fixa - cuja amplitude e frequência está contida dentro da faixa que uma guitarra produziria - para que se obtenha, na saída do pré-amplificador, um sinal característico que seria obtido pelo sistema caso uma guitarra fosse utilizada em sua entrada. Este tipo de identificação não é de interesse prático pois não possui grande generalidade, mas será aqui utilizada para definir os parâmetros que são estipulados pelo usuário/projetista da rede neural.

Os testes foram realizados somente para a rede neural recorrente treinada com filtro de Kalman estendido, pois a utilização das outras redes - a rede recorrente treinada com *back-propagation through time* e a rede *long short-term memory* - ocasionaria em um tempo total de execução de alguns dias. Tal demora foi considerada inviável durante o tempo disponível para realizar este trabalho, visto que tomaria um tempo que seria utilizado em outros pontos essenciais.

6.1.1 Definição de faixa de parâmetros

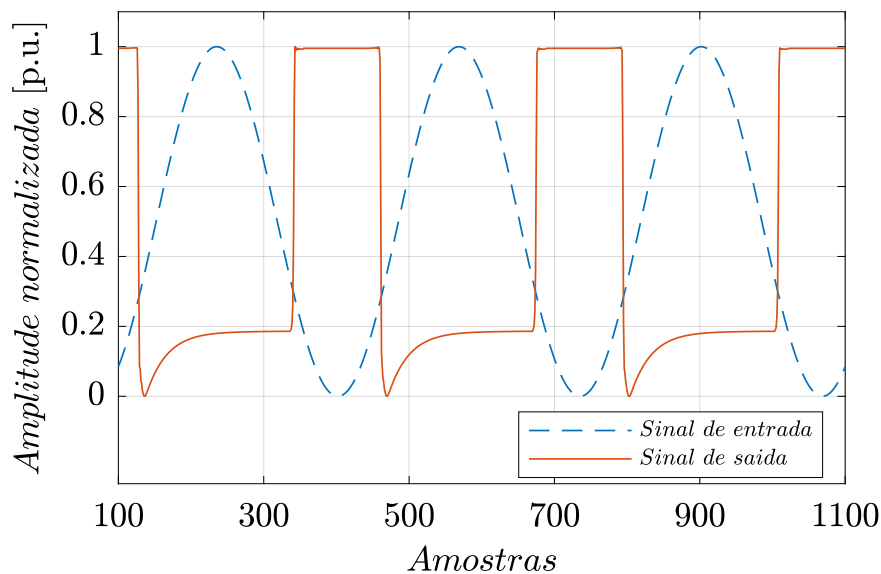
O objetivo deste primeiro teste é encontrar uma faixa onde os parâmetros a serem definidos são ótimos - ou próximos disso. Para realizá-lo com êxito, há de se ter uma noção inicial sobre os parâmetros que serão utilizados.

Visto que o *tonestack* consiste em um filtro puramente linear e pode ser facilmente identificado por meio de técnicas de identificação de sistemas lineares, ou mesmo implementado

por meio de equações diferenciais que representem seu comportamento físico, este foi retirado do circuito para se obter o sinal que é utilizado nos testes. Possibilita-se, desta forma, que a rede esteja "focada" na emulação de um sistema altamente não linear, sem uma parte extra para dificultar a identificação. Na prática, a rede neural seria aplicada em um processador digital de sinais, por exemplo, onde a implementação de equações que representem o *tonestack* seria facilmente realizada, portanto a sua inclusão na identificação do pré-amplificador valvulado não é essencial a este trabalho.

Aplicando um sinal senoidal com 1 kHz de frequência e 0,5 V de amplitude (linha azul tracejada na Figura 49) no pré-amplificador simulado, para o canal de alto ganho, obtém-se o sinal de saída conforme a linha contínua laranja presente na Figura 49. Tais sinais foram re-amostrados de acordo com o procedimento descrito na Subseção 5.1.1 buscando um menor tempo de execução dos testes, assim como foi limitada a quantidade de épocas - iterações que o treinamento realiza até que os pesos sejam atualizados - em cem, visto que o algoritmo de treinamento baseado no filtro de Kalman estendido converge em poucas épocas - vinte a trinta.

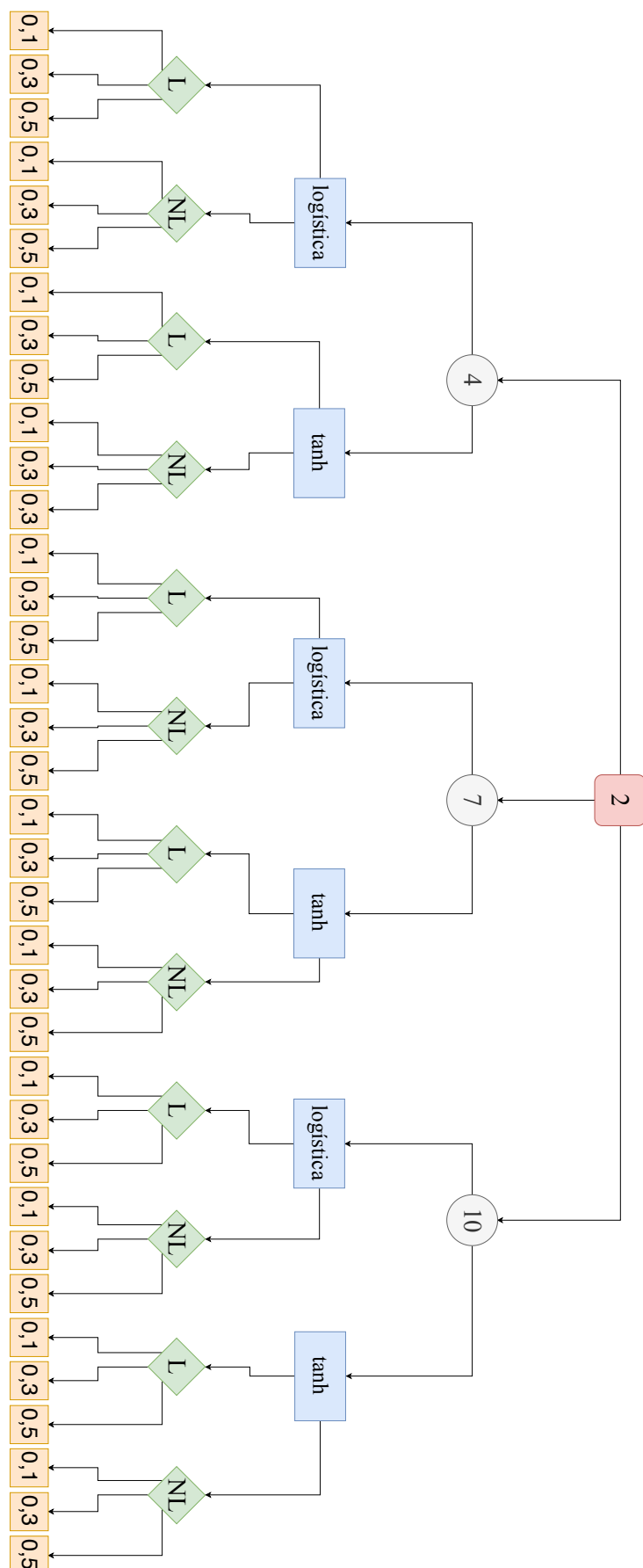
Figura 49 – Sinais utilizados no primeiro teste de definição de parâmetros da rede neural.



Fonte: o próprio autor, 2019.

O primeiro teste consiste em uma árvore de simulações (Figura 50) onde são variados: número de camadas, número de neurônios, tipo de função de ativação, qual o tipo da função de ativação do neurônio da saída (linear ou não linear) e taxa de aprendizado, respectivamente representados na Figura 50 pelas cores: vermelho, cinza, azul, verde e laranja, onde L e NL significa, respectivamente, linear e não linear. São analisados os valores de MSE resultantes, para a identificação do sinal escolhido, para definir qual a estrutura que obteve o melhor resultado.

Figura 50 – Árvore de testes de parâmetros.



Fonte: o próprio autor, 2019.

Anteriormente à execução deste teste, de forma empírica, percebeu-se que uma rede neural recorrente de uma única camada obtém um erro substancialmente maior que uma rede de duas camadas. Para fornecer ao leitor uma noção quantitativa, em 12 execuções das redes neurais de uma camada variando o número de neurônios, função de ativação, tipo do neurônio de saída e taxa de aprendizado com objetivo de identificar o sinal da Figura 49, obteve-se, na média, um MSE de 0,57. Realizando 12 execuções semelhantes para duas camadas, obteve-se uma média dos MSEs de 0,02. Portanto, como a execução do teste é demorada, as redes de uma única camada foram retiradas do mesmo.

Os pesos sinápticos iniciais da rede neural deverão ser fixos (e iguais) para todas as situações, visando testar a variação de parâmetros sem que o valor inicial dos pesos influencie na escolha. Executando o teste de acordo com a árvore presente na Figura 50, obteve-se os erros médios quadráticos apresentados na Tabela 2. A célula destacada em verde mostra o menor MSE obtido.

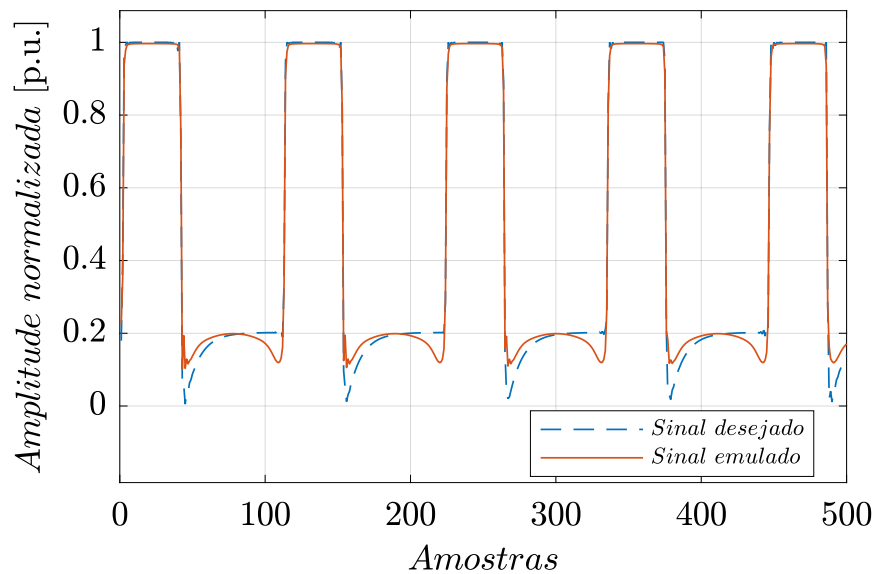
Tabela 2 – MSE de validação das redes neurais para o teste inicial de parâmetros.

Ativação; saída; taxa	Quatro neurônios	Sete neurônios	Dez neurônios
<i>Logística; L; 0,5</i>	0,0053311	0,0049155	0,0023418
<i>Logística; L; 0,3</i>	0,0071163	0,0020328	0,0030335
<i>Logística; L; 0,1</i>	0,0384778	0,0040699	0,0073336
<i>Logística; NL; 0,5</i>	0,0015043	0,0010965	0,0000539
<i>Logística; NL; 0,3</i>	0,0010684	0,0011424	0,001074
<i>Logística; NL; 0,1</i>	0,0011238	0,0008744	0,0011344
<i>Tanh; L; 0,5</i>	0,3754941	0,0067177	0,1643468
<i>Tanh; L; 0,3</i>	0,8908207	0,0133660	0,0074725
<i>Tanh; L; 0,1</i>	0,0287255	0,2992241	0,2276541
<i>Tanh; NL; 0,5</i>	0,2882785	0,4440789	0,2629054
<i>Tanh; NL; 0,3</i>	0,2680444	0,4861853	0,2845783
<i>Tanh; NL; 0,1</i>	0,2664526	0,2611053	0,2760634

Fonte: o próprio autor, 2019.

Percebe-se conforme a Tabela 2 que a estrutura que obteve o menor MSE no teste foi a rede com duas camadas, sete neurônios em cada camada, função de ativação logística, neurônio de saída não linear e taxa de aprendizagem de 0,1. O sinal desejado (sinal tracejado em azul) e o sinal resultante da validação da rede neural de menor MSE (sinal contínuo em laranja) podem ser observados na Figura 51. O sinal emulado mostra que a rede é capaz de identificar as características não lineares do sinal que definem sua razão cíclica diferente de 0,5, porém não é completamente capaz de representar as harmônicas que caracterizam a parte inferior da forma de onda do sinal.

Figura 51 – Sinal desejado e emulado pela rede neural de menor MSE no teste de parâmetros.



Fonte: o próprio autor, 2019.

6.1.2 Refinamento dos parâmetros para definição final da estrutura

Com base nos resultados obtidos no teste realizado na Subseção 6.1.1, será realizado um novo teste, muito semelhante, porém para uma nova faixa de valores para os parâmetros. Sabendo, conforme a Tabela 2 que a rede de duas camadas, sete neurônios, função de ativação logística, neurônio de saída não linear e taxa de aprendizagem 0,1 obteve o menor MSE, são variados os parâmetros conforme:

- número de camadas: fixado em 2;
- número de neurônios: de 5 a 9;
- função de ativação: logística;
- função de ativação do neurônio de saída: não linear;
- taxa de aprendizagem: 0,1 a 0,4.

Os valores de MSE obtidos no segundo teste são expostos na Tabela 3, onde a célula destaca em verde mostra o menor erro obtido e η representa a taxa de aprendizagem. Para todas as redes desse caso, foram fixadas duas camadas, função de ativação logística e neurônio de saída não linear.

Tabela 3 – MSE de validação das redes neurais para o teste de refinamento de parâmetros.

Número de neurônios	$\eta = 0,4$	$\eta = 0,3$	$\eta = 0,2$	$\eta = 0,1$
5	0,0011189	0,0012110	0,0011584	0,0033259
6	0,0031683	0,0023474	0,0012233	0,0013892
7	0,0012057	0,0011335	0,0012803	0,0010941
8	0,0012512	0,0011097	0,0011487	0,0012807
9	0,0011502	0,0013045	0,0012705	0,0011853

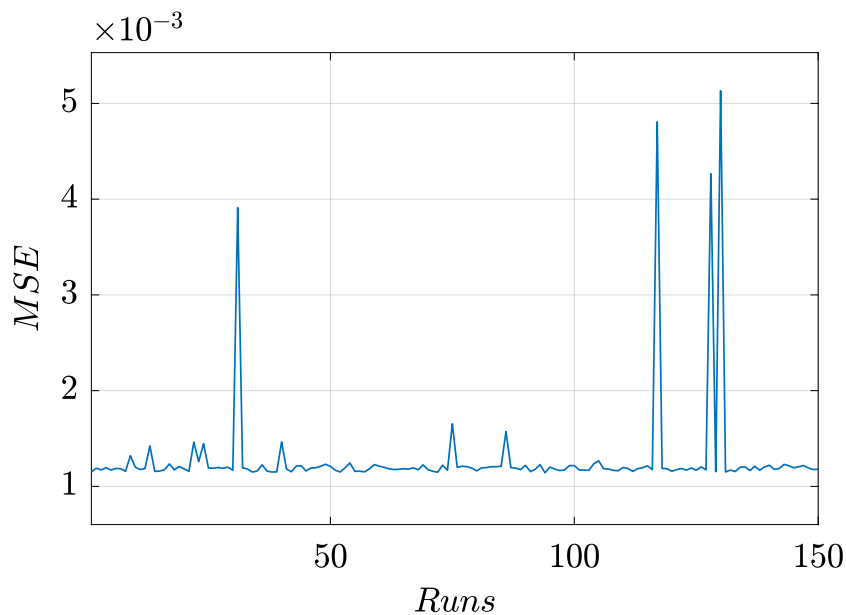
Fonte: o próprio autor, 2019.

Define-se portanto, a estrutura da rede neural recorrente com treinamento por filtro de Kalman estendido para realizar a emulação do pré-amplificador valvulado: duas camadas, sete neurônios por camada, função de ativação logística, neurônio de saída não linear e taxa de aprendizagem de 0,1.

6.1.3 Análise de convergência da estrutura final

Definida a estrutura da rede neural conforme subseção anterior, pode-se testar a convergência da estrutura final diante de diferentes pesos sinápticos iniciais. Iniciando os pesos de forma aleatória, com distribuição gaussiana ¹, a rede neural é executada 150 vezes, obtendo valores de MSE conforme mostrados de forma gráfica na Figura 52, com um desvio padrão de $8,983194 \cdot 10^{-6}$, onde o termo *runs* refere-se às execuções da rede.

Figura 52 – MSEs das 150 execuções da rede neural com diferentes pesos iniciais.



Fonte: o próprio autor, 2019.

¹ No software *Matlab*®, pode-se gerar valores aleatórios com distribuição gaussiana por meio da função *randn*.

Percebe-se que os valores de MSE obtidos ficaram próximos e o valor de desvio padrão foi baixo em relação ao menor valor de MSE obtido, aproximadamente três ordens de grandeza menor. Portanto, considera-se que a rede é convergente diante de diferentes condições iniciais para os pesos sinápticos, podendo então esta ser utilizada nas emulações.

7 RESULTADOS EXPERIMENTAIS

Definida a estrutura da rede neural recorrente com treinamento por filtro de Kalman estendido conforme o procedimento exposto no Capítulo 6, é assumida a mesma estrutura para as três redes neurais utilizadas na emulação do pré-amplificador, de forma a se tentar obter uma comparação direta entre elas.

Foram aplicados três sinais ao pré-amplificador valvulado, por simulação e no próprio protótipo. São estes: ruído-branco, senoide de frequência variável e sinal de guitarra. Os dados obtidos foram identificados pelas três redes neurais: rede neural recorrente treinada com *back-propagation through time* (RNN-BPTT); rede neural recorrente treinada com *extended Kalman filter* (RNN-EKF); e rede *long short-term memory* treinada com *back-propagation through time* (B-LSTM). Para a rede LSTM, foi utilizada a implementação desenvolvida em (SCHMITZ; EMBRECHTS, 2018), por ser a aplicação de maior êxito até o momento na literatura no quesito emulação de amplificadores valvulados. Na RNR *vanilla*¹ é utilizado o tBPTT, enquanto que na LSTM é utilizado o eBPTT. Durante o capítulo, ambos serão referidos somente como BPTT por simplicidade.

Neste capítulo, é descrito o processo de tratamento de dados utilizado para adequar os sinais adquiridos por simulação ou do protótipo às redes neurais, assim como são expostos os resultados obtidos com base nestes.

7.1 DADOS DE IDENTIFICAÇÃO

Os dados obtidos via simulação e pelo protótipo são re-amostrados utilizando a função *resample* do software *Matlab*® para a redução do número de amostras do conjunto de dados e um menor tempo de execução dos algoritmos de treinamento. Para seguir os padrões comumente utilizados para gravação e edição de áudio, a nova frequência de amostragem foi mantida sempre superior a 22,050 kHz².

Na maioria dos casos de simulação, devido a curta duração dos sinais, pode-se utilizar frequências de amostragem da ordem de 1 MHz. No caso dos sinais obtidos do protótipo, após a re-amostragem, padronizou-se a frequência de amostragem de 25 kHz. A seguir, é descrito o processo de obtenção dos sinais para o caso de simulação e real.

7.1.1 Dados de simulação

Para que fosse possível obter uma situação de simulação convergente, foi necessário utilizar um passo de cálculo de $1 \cdot 10^{-08}$ s devido a não otimização da implementação do modelo da válvula presente na Subseção 5.1.1. Padronizou-se a aplicação de sinais com duração

¹ Termo utilizado na literatura estrangeira para definir redes neurais recorrentes simples.

² As frequências mais comumente utilizadas para gravações em estúdio são de 22,050 kHz, 44,100 kHz, 88,200 kHz, 96,000 kHz e 192,000 kHz.

de 120 ms no modelo de simulação do pré-amplificador apresentado na Seção 5.4. O tempo observado durante a execução das simulações variou de 40 minutos a uma hora para cada treinamento.

O sinal de 120 ms é composto de uma senoide de 1 kHz e amplitude de 1 V com duração de 20 ms para que o sistema se acomodasse diante do transitório inicial - e, portanto, seu tempo de carga não afete os sinais que deseja-se obter. Os restantes 100 ms do sinal são compostos por um dos sinais de identificação: ruído branco ou *sine sweep*. Para o caso de simulação, não foi possível aplicar o sinal de guitarra com duração mínima para realizar a identificação³ devido ao tempo necessário para executá-lo e à capacidade de armazenamento necessária para salvar os dados obtidos, portanto, este sinal não foi abordado nos resultados de simulação.

Desta forma, foi padronizada a aplicação de sinais de ruído branco gaussiano e senoides de frequência variável com três diferentes amplitudes: 0,1 V, 0,3 V e 0,5 V, visto que a distorção altamente não linear do pré-amplificador valvulado terá diferentes respostas para as diferentes amplitudes. A definição da utilização de tais amplitudes foi feita previamente, de forma empírica, por meio de simulações isoladas.

A senoide de frequência variável foi gerada pela função *chirp* do software *Matlab*® variando a frequência de 60 Hz a 3500 Hz, visto que havia somente 100 ms de tempo disponível para a variação de frequência. A expansão para maiores frequências pode ser facilmente realizada e provavelmente obteria resultados semelhantes, visto que a corda mais aguda da guitarra, na afinação padrão (afinada em mi - 88,4 Hz), soa a 329,628 Hz. Considerando duas oitavas acima (nota mais aguda de uma guitarra com 24 casas), obtém-se uma frequência de 1318,512 Hz. Considerando que as notas mais agudas serão tocadas menos frequentemente, assume-se que a maior parte da resposta harmônica do pré-amplificador às notas de guitarra estará contida dentro da faixa de 60 Hz a 3500 Hz.

Os dados obtidos possuem 12 milhões de amostras, para 120 ms de duração, com frequência de amostragem de 100 MHz. Assim, re-amostra-se utilizando uma relação de 1/150, obtendo um sinal de 80 mil amostras. Retirando-se as amostras referentes ao transitório inicial, obtém-se um sinal final de 66,6 mil amostras, amostrado em 1 MHz.

7.1.2 Dados do protótipo

Para obter os dados do protótipo, um notebook foi utilizado como gerador de funções. A saída de áudio do notebook foi conectada à entrada do pré-amplificador valvulado. O sinal de entrada foi gerado no software *Matlab*® e os sinais de entrada e saída para a identificação foram adquiridos, no protótipo, de forma simultânea, por meio de um osciloscópio da *Tektronix*® modelo MDO3014 em modo de detecção de evento por *trigger*. Para os sinais do tipo ruído branco com distribuição gaussiana e *sine sweep* foi coletada uma janela de 400 ms, com um

³ A duração mínima para realizar a identificação consiste no tempo necessário para várias notas serem toadas no sinal. (SCHMITZ; EMBRECHTS, 2018) sugere que seja utilizado um sinal de 10 a 20 segundos.

milhão de amostras. Para os sinais de guitarra, foi definida uma janela contendo dez segundos de áudio.

Todos os sinais, de ruído branco, de *sine sweep* e de guitarra obtidos foram reamostrados para uma frequência de amostragem final de aproximadamente 25 kHz para serem utilizados no treinamento das redes neurais. As amplitudes dos sinais de identificação aplicados ao pré-amplificador real (protótipo) seguem o mesmo procedimento da Subseção 7.1.1. Nenhum filtro de ruído foi adicionado para compensar ruído de medição ou inerente ao sistema real, visto que é de interesse deste estudo comparar as redes neurais, incluindo sua capacidade de lidar com ruído branco ou colorido.

7.2 RESULTADOS EXPERIMENTAIS DE IDENTIFICAÇÃO

As redes RNN-BPTT e RNN-EKF foram implementadas no software *Matlab* ®, enquanto que a rede B-LSTM foi implementada em *Python*, utilizando o *framework Tensorflow* ®, conforme a implementação apresentada em (SCHMITZ; EMBRECHTS, 2018). Os códigos de implementação das redes neurais, assim como todos os dados obtidos do protótipo estão disponíveis em (FIORIO, 2019b).

As redes recorrentes também foram implementadas nas linguagens C e *Python* para que fosse averiguado em qual linguagem estas rodariam em menor tempo, porém demonstraram-se mais rápidas de serem executadas na sua versão implementada em *Matlab* ®. Tal fato provavelmente deve-se à otimização das funções do software *Matlab* ® inerente ao próprio programa. Nenhum tipo de otimização de velocidade de processamento de código ou *framework* foram utilizadas nas outras linguagens de programação para estas redes.

Ao treinar as redes neurais, dois valores de erro são obtidos: erro de treinamento e erro de validação. O erro de treinamento somente é utilizado para ter noções de como a rede está lidando com os dados e com o treinamento, ou em casos de aprendizado em tempo real, em que o erro de treinamento torna-se o erro "real". O erro de validação aqui referido foi obtido aplicando o mesmo sinal utilizado no treinamento na rede de validação. O motivo do uso do mesmo sinal se dá pela forma de treinamento da rede. Como esta é treinada para uma profundidade temporal pré-definida, têm-se uma rede com uma resposta temporal diferente para treinamento e validação, assim como deve-se analisar sua capacidade de generalização. Enquanto a rede de treinamento possui uma capacidade temporal limitada, a rede de validação possui tantas camadas temporais quanto número de amostras disponíveis nos dados de validação. Portanto, há de se analisar a capacidade da rede de, após o treinamento, emular o sinal para o qual esta foi treinada. Caso a emulação deste sinal não seja razoavelmente aceitável, não há por que testar a capacidade de generalização da rede neural com sinais diferentes, pois a mesma não obterá êxito.

Os erros médios quadráticos para as saídas de validação das redes RNN-BPTT, RNN-EKF e B-LSTM, com base nos dados simulados, podem ser observados conforme a Tabela 4. Os nomes dos sinais de treinamento estão abreviados na ordem: tipo do sinal (*White-noise* ou

Sine-sweep), nível de ganho (*highgain* - HG - ou *lowgain* - LG) e amplitude (0,1; 0,3; 0,5). O sinal de ruído branco com amplitude de 0,1 V não foi utilizado em simulação por possuir um tempo transitório muito acentuado, dificultando sua obtenção.

Tabela 4 – MSE de validação das redes neurais para dados de simulação.

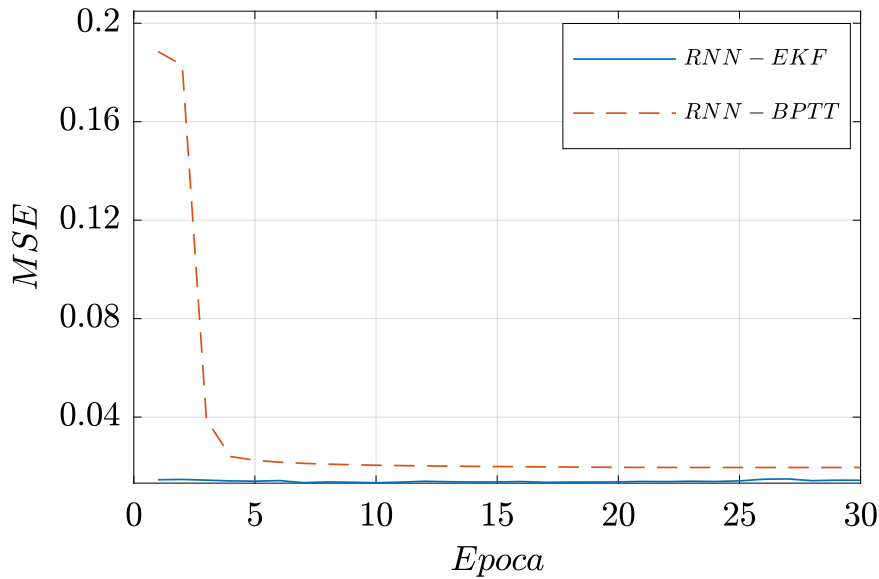
Sinal de treinamento	RNN-BPTT	RNN-EKF	B-LSTM
<i>Sine-sweep LG 01</i>	0,0066407	0,0031842	0,0000003
<i>Sine-sweep LG 03</i>	0,0115381	0,0078598	0,0000008
<i>Sine-sweep LG 05</i>	0,0195536	0,0143541	0,0000009
<i>Sine-sweep HG 01</i>	0,0696815	0,1729353	0,0000539
<i>Sine-sweep HG 03</i>	0,0390683	0,0891628	0,0000200
<i>Sine-sweep HG 05</i>	0,3373789	0,0393861	0,0000139
<i>White-noise LG 01</i>	0,0220963	0,0146429	0,0001847
<i>White-noise LG 03</i>	0,0200459	0,0244008	0,0001310
<i>White-noise LG 05</i>	0,0402776	0,0339782	0,0001577
<i>White-noise HG 03</i>	0,1147576	0,0866533	0,0075056
<i>White-noise HG 05</i>	0,1125819	0,1198915	0,0019836

Fonte: o próprio autor, 2019.

Percebe-se claramente a superioridade da LSTM em relação às outras redes, o que era esperado, visto que foi exposto em diversas obras - (HOCHREITER; SCHMIDHUBER, 1997), (SCHMITZ; EMBRECHTS, 2018), (CATALA, 2017), etc - o fato de sua estrutura lidar muito bem com sistemas não lineares onde há dependência temporal de curto e longo prazo. Comparando as duas redes neurais recorrentes onde somente a forma de treinamento foi modificada, percebe-se que em sete dos onze casos a rede com treinamento por filtro de Kalman estendido obteve menor MSE. Contudo, a diferença de erro obtida é pequena, dificultando uma conclusão sólida sobre a diferença entre ambos os algoritmos de treinamento.

Tais fatos indicam que mesmo mudando a forma de treinamento, a rede neural recorrente em sua forma pura (ou *vanilla*, como é referida na literatura) não possui estrutura para emular sistemas cuja dependência temporal de longo prazo é considerável, devido ao *vanishing gradient*. No entanto, uma das vantagens observadas ao utilizar o *extended Kalman filter* foi o tempo de convergência. Para a maior parte dos casos, a rede treinada pelo filtro de Kalman estendido foi capaz atingir um valor de MSE próximo ao da convergência em apenas uma época. A Figura 53 ilustra este fenômeno, comparando o MSE de validação das duas formas de treinamento das RNRs comuns para o sinal de *sine-sweep* com baixo ganho e amplitude de 0,3 V.

Figura 53 – MSE de validação das redes RNN-EKF e RNN-BPTT para um *sine-sweep* de amplitude 0,3 V e baixo ganho.



Fonte: o próprio autor, 2019.

Para o protótipo real, onde há ruído de medição e inerente ao sistema, há de se utilizar a matriz de covariância do ruído $\mathbf{R}$. Para obtê-la foi observada a transformada rápida de Fourier dos sinais utilizados na identificação. Percebeu-se que a partir de aproximadamente 40 kHz não haviam mais componentes harmônicas relevantes. Assim, isolou-se o ruído do sinal filtrando-o por meio de um filtro passa altas do tipo *Chebyshev* de décima ordem, utilizando a função *cheby1* do software *Matlab*®, com uma frequência de corte de 40 kHz, e obteve-se a covariância do mesmo.

Utilizando os dados reais do protótipo montado em laboratório, foram obtidos novos valores de MSE de validação para as três redes neurais - agora incluindo os sinais de guitarra para baixo ganho (Guitarra LG) e para alto ganho (Guitarra HG) - dados conforme a Tabela 5.

Novamente, a rede *long short-term memory* obtém resultados muito superiores às outras redes recorrentes. Neste caso no entanto, os dados possuem uma quantidade considerável de ruído de medição e ruído inerente ao sistema. Esperava-se que a rede recorrente treinada com *extended Kalman filter* obtivesse um resultado superior ao observado na rede com *back-propagation through time* devido ao filtro de Kalman estendido lidar melhor com ruído, por basear-se na média e na covariância do sinal para realizar a estimação dos pesos sinápticos. Porém, a relação de situações em que a rede RNN-EKF saiu-se melhor quando comparada à RNN-BPTT foi menor que para o caso anterior: sete de quatorze. Novamente, mesmo que uma melhor escolha da matriz de covariância do ruído possivelmente melhoraria os resultados da rede treinada pelo EKF, acredita-se que o maior problema esteja na topologia da rede neural, e não na sua estrutura.

Visando uma maior generalidade na identificação do sistema, juntou-se os sinais de

Tabela 5 – MSE de validação das redes neurais para dados do protótipo.

Sinal de treinamento	RNN-BPTT	RNN-EKF	B-LSTM
<i>Sine-sweep LG 01</i>	0,0189667	0,2426349	0,0000273
<i>Sine-sweep LG 03</i>	0,0361303	0,0251071	0,0000146
<i>Sine-sweep LG 05</i>	0,0493229	0,0330314	0,0000130
<i>Sine-sweep HG 01</i>	0,0418579	0,0746764	0,0000817
<i>Sine-sweep HG 03</i>	0,0626707	0,1838646	0,0000572
<i>Sine-sweep HG 05</i>	0,0452895	0,1425609	0,0000783
<i>White-noise LG 01</i>	0,0124474	0,0238145	0,0001708
<i>White-noise LG 03</i>	0,0287379	0,0344735	0,0000780
<i>White-noise LG 05</i>	0,0356893	0,0871564	0,0001098
<i>White-noise HG 01</i>	0,0984908	0,0978312	0,0009683
<i>White-noise HG 03</i>	0,1096698	0,1094005	0,0013827
<i>White-noise HG 05</i>	0,1239540	0,1210592	0,0018615
<i>Guitarra LG</i>	0,0063042	0,0070350	0,0000653
<i>Guitarra HG</i>	0,0275544	0,0269354	0,0065570

Fonte: o próprio autor, 2019.

sine-sweeps com 0, 1, 0, 3 e 0,5 V de amplitude, dos dados do protótipo, em um único sinal, para cada um dos ganhos, para utilizá-lo como sinal de treinamento das redes neurais. O mesmo processo foi realizado para os sinais de ruído branco com distribuição gaussiana. Realizando o treinamento para estes sinais e utilizando-os na validação da rede neural, obtém-se os MSEs conforme a Tabela 6.

Tabela 6 – MSE de validação das redes neurais para os sinais compostos.

Sinal de treinamento	RNN-BPTT	RNN-EKF	B-LSTM
<i>Sine-sweeps HG</i>	0,0960710	0,0934958	0,0000732
<i>Sine-sweeps LG</i>	0,0490289	0,0236506	0,0000229
<i>White-noises HG</i>	0,0825189	0,0819639	0,0043263
<i>White-noises LG</i>	0,0146918	0,0321154	0,0000653

Fonte: o próprio autor, 2019.

Aplicando os sinais de guitarra de alto e baixo ganho nas redes neurais treinadas conforme os sinais presentes na Tabela 6 e, também utilizando os próprios sinais de guitarra como treinamento, a Tabela 7 é obtida.

Com base nos valores de MSE apresentados na Tabela 7, percebe-se que a pior situação foi quando tentou-se emular o pré-amplificador para o canal de alto ganho utilizando ruído branco como sinal de identificação. No geral, as situações com alto ganho obtiveram maior erro, o que era esperado visto que quanto maior o ganho, mais não linear é a resposta da válvula a um sinal de entrada. Em todos os casos, a rede LSTM mostrou-se superior em questões de MSE.

Tais resultados apontam que para o caso da emulação de áudio altamente não linear (pré-

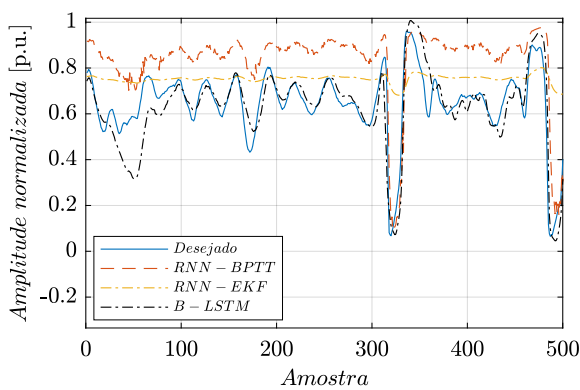
Tabela 7 – MSE das redes neurais, treinadas por sinais compostos e de guitarra, para um sinal de guitarra.

Sinal de treinamento	RNN-BPTT	RNN-EKF	B-LSTM
<i>Sine-sweeps HG</i>	0,0641137	0,0591059	0,0000622
<i>Sine-sweeps LG</i>	0,0458974	0,0211141	0,0000245
<i>White-noises HG</i>	0,0833399	0,0819821	0,0042050
<i>White-noises LG</i>	0,0248930	0,0476392	0,0000668
<i>Guitarra HG</i>	0,0275544	0,0269354	0,0065570
<i>Guitarra LG</i>	0,0063042	0,0070350	0,0000653

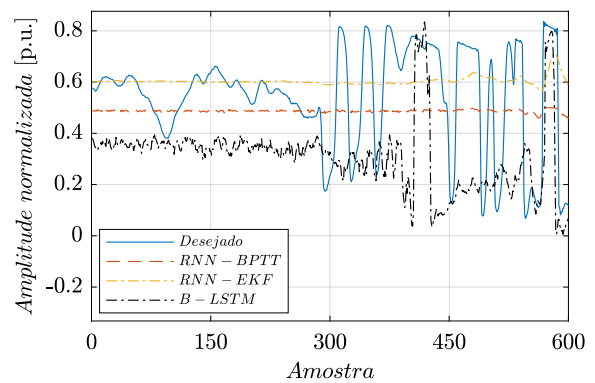
Fonte: o próprio autor, 2019.

amplificador valvulado), a complexidade da identificação sobrepõe-se à técnica de aprendizagem. Mesmo que a estrutura das redes RNR *vanilla* fosse modificada, espera-se que o resultado se mantenha similar, visto que o problema pode estar na topologia das redes, e não na estrutura. Salienta-se, porém, que a influência da matriz $\mathbf{R}$ do treinamento por EKF ainda carece de maior exploração. Como exposto pela literatura e demonstrado na Subseção 4.4.2.4, o *vanishing gradient* aparenta ser o principal problema das redes neurais recorrentes comuns, este que é corrigido utilizando uma topologia desenvolvida especificamente para que o problema dos gradientes seja suprimido - a rede *long short-term memory*.

Para uma melhor compreensão do leitor, um trecho do sinal de guitarra emulado por cada rede neural e do sinal desejado são expostos de forma gráfica da Figura 54 à 56. Tais resultados gráficos são divididos de acordo com o sinal de excitação e qual canal de ganho foi utilizado.

Figura 54 – Trecho do sinal emulado utilizando os *sine-sweeps* no treinamento.

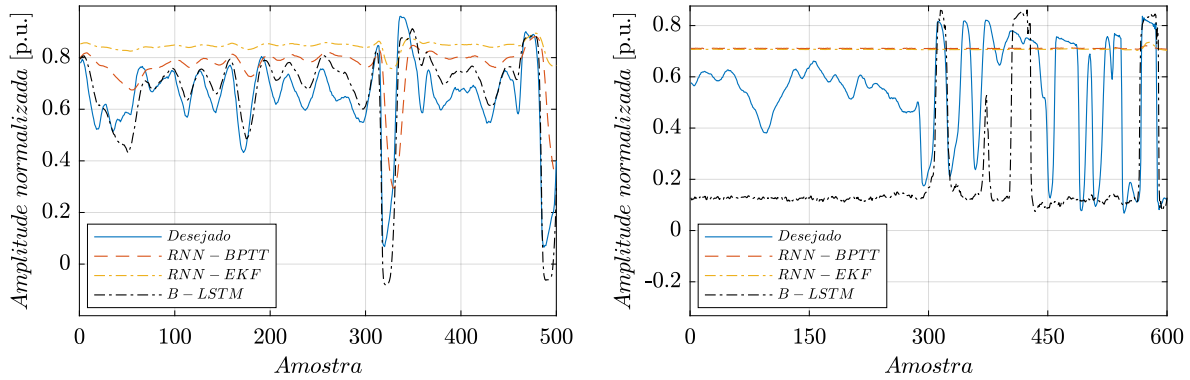
(a) Para o canal de baixo ganho.



(b) Para o canal de alto ganho.

Fonte: o próprio autor, 2019.

Figura 55 – Trecho do sinal emulado utilizando os sinais de ruído no treinamento.

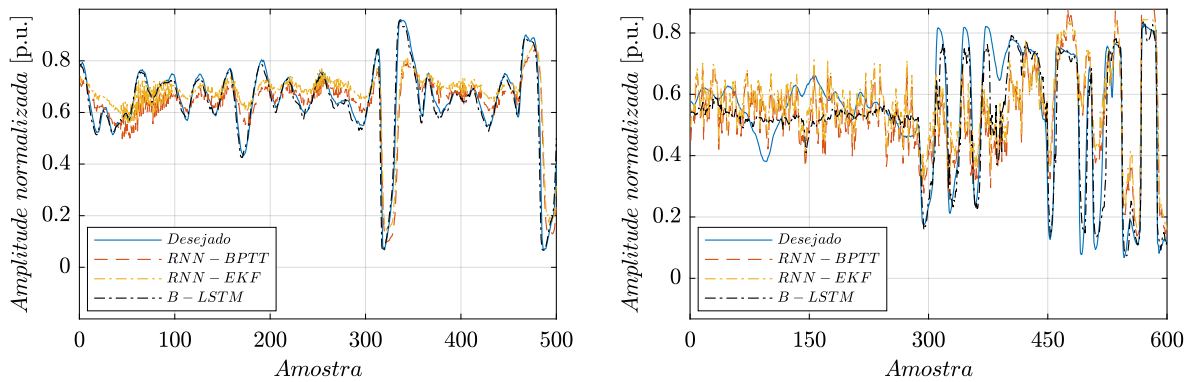


(a) Para o canal de baixo ganho.

(b) Para o canal de alto ganho.

Fonte: o próprio autor, 2019.

Figura 56 – Trecho do sinal emulado utilizando os sinais de guitarra no treinamento.



(a) Para o canal de baixo ganho.

(b) Para o canal de alto ganho.

Fonte: o próprio autor, 2019.

8 CONCLUSÃO

Neste trabalho, três redes foram utilizadas na emulação de um pré-amplificador valvulado para guitarra elétrica: uma rede neural recorrente treinada pelo algoritmo de *back-propagation through time* (RNN-BPTT); uma rede neural recorrente treinada pelo filtro de Kalman estendido (RNN-EKF); e uma rede *long short-term memory* treinada com *back-propagation through time* (B-LSTM). Foram utilizados dados de simulação, onde foi implementado um modelo da válvula em um software de simulação. Os dados reais do pré-amplificador foram obtidos com base no protótipo projetado e construído. A simulação e o protótipo são detalhados no Capítulo 5. Um teste para a definição da estrutura das RNRs foi elaborado neste trabalho, justificando a escolha dos parâmetros das redes. Os sinais de identificação utilizados foram uma senoide de frequência variável, um ruído branco de distribuição gaussiana e um sinal de guitarra elétrica - este último somente para o caso real.

Observou-se nos resultados que os MSEs obtidos pela B-LSTM foram de mil vezes menores que os MSEs obtidos pelas RNRs *vanilla*. Acredita-se que este fato deve-se ao problema do *vanishing gradient* estar presente durante o treinamento das topologias convencionais de redes neurais recorrentes, o que é corrigido na topologia da LSTM. A utilização do EKF no lugar do BPTT nas redes *vanilla* durante o treinamento ocasionou em resultados semelhantes, onde não pode-se inferir de forma sólida qual das técnicas se saiu melhor nas identificações de forma geral. No entanto, o algoritmo do EKF mostrou-se mais rápido na convergência do treinamento. Os resultados são expostos de forma quantitativa no Capítulo 7.

Uma análise quanto a matriz de covariância do ruído de medição ($\mathbf{R}$) do EKF mais aprofundada deve ser feita, em trabalhos futuros, visando-se obter um melhor resultado. Da mesma forma, sugere-se aplicar o treinamento por filtro de Kalman estendido em uma rede LSTM para que se possa comparar ambos os algoritmos de treinamentos em uma situação onde a topologia da rede neural não sofre de problemas como o *vanishing gradient*. Também pode ser realizada a comparação das técnicas de treinamento aqui utilizadas com um algoritmo bio-inspirados de otimização (como os algoritmos genéticos, p.e.), podendo-se analisar sua influência no *vanishing gradient* em redes neurais recorrentes *vanilla*. Sugere-se, por fim, buscar métodos de cálculo de erro que abstraíam a defasagem do sinal emulado, visto que tal fenômeno, para um único sinal sonoro, não é percebido pelo ouvido humano.

Diante do que foi apresentado, percebeu-se que o principal problema das redes neurais recorrentes comuns na emulação de um sistema não linear reside, principalmente, na topologia destas, que ocasiona no *vanishing gradient* durante o treinamento. A utilização da LSTM mostrou-se válida e com resultados muito superiores às outras redes, com maior complexidade estrutural e custo computacional.

REFERÊNCIAS

- AGUIRRE, L. *Introdução à Identificação de Sistemas – Técnicas Lineares e Não-Lineares Aplicadas a Sistemas Reais*. Editora UFMG, 2015. ISBN 9788570415844. Disponível em: <<https://books.google.com.br/books?id=f9IwE7Ph0fYC>>. Citado 7 vezes nas páginas 23, 32, 36, 43, 44, 45 e 46.
- T. Araya e A. Suyama. *Sound effector capable of imparting plural sound effects like distortion and other effects*. 1996. 5570424. Acesso em 26/09/2018. Disponível em: <<https://patents.google.com/patent/US5570424A/en>>. Citado na página 33.
- BERTOLDI, B. Trabalho de conclusão de curso (Bacharelado em Engenharia Elétrica - Área: Eletrônica de Potência), *Estudo de um amplificador classe D destinado à guitarra elétrica*. 2016. Citado na página 26.
- BLENCOWE, M. *Designing Valve Preamps for Guitar and Bass*. [S.l.]: Merlin Blencowe, 2009. Citado 2 vezes nas páginas 26 e 86.
- BOSS. *Katana-50 Guitar Amplifier*. 2017. Boss' Website. Acesso em 21/09/2018. Disponível em: <<https://www.boss.info/us/products/katana-50/>>. Citado na página 29.
- BRAGA, N. C. *A História do transistor*. c.a. 2010. Instituto NCB. Acessado em 21/09/2018. Disponível em: <<http://www.newtoncbraga.com.br/index.php/artigos/67-historia-da-eletronica/407-ahistoria-do-transistor.html>>. Citado na página 25.
- CATALA, O. D. T. *Audio Effects Emulation with Neural Networks*. 2017. Disponível em: <<http://hdl.handle.net/10251/88860>>. Citado na página 104.
- COVERT, J.; LIVINGSTON, D. L. A vacuum-tube guitar amplifier model using a recurrent neural network. In: *2013 Proceedings of IEEE Southeastcon*. [S.l.: s.n.], 2013. p. 1–5. ISSN 1558-058X. Citado 2 vezes nas páginas 37 e 38.
- DEMPWOLF, K.; ZÖLZER, U. A physically-motivated triode model for circuit simulations. In: . [S.l.: s.n.], 2011. Citado 2 vezes nas páginas 67 e 68.
- DUNKEL, W. et al. The fender bassman 5f6-a family of preamplifier circuits—a wave digital filter case study. In: RAJMIC, P.; RUND, F.; SCHIMMEL, J. (Ed.). *Proceedings of the 19th International Conference on Digital Audio Effects*. [S.l.]: DAFx, 2016. p. 263–270. ISBN 2413-6700. Awarded best paper number three. Citado na página 34.
- ELMAN, J. L. Finding structure in time. *COGNITIVE SCIENCE*, v. 14, n. 2, p. 179–211, 1990. Citado na página 51.
- FARINA, A.; ARMELLONI, E. *Emulation of Not-Linear, Time-Variant Devices by the Convolution Technique*. 2005. Citado 2 vezes nas páginas 35 e 36.
- FIORIO, L. V. *Relatório Técnico cmear-2019-01-001*. 2019. Relatório técnico de projeto. Nenhuma citação no texto.
- FIORIO, L. V. *Repositório de códigos e dados para o trabalho intitulado "Emulação de um pré-amplificador valvulado utilizando redes neurais artificiais"*. 2019. Disponível em: <https://github.com/LVF784/rnn_preamp_tcc>. Citado na página 103.

- FJESTAD, Z. *Trash or Treasure: A Pre-Fender Fender*. 2016. PREMIERGuitar. Acesso em 18/09/2018. Disponível em: <<https://www.premiarguitar.com/articles/23834-trash-or-treasure-a-pre-fender-fender>>. Citado na página 27.
- FRYETTE, S. A brief history of electric guitar amps, part one. *Fryette Amplification*, mar. 2016. Acesso em 18/09/2018. Citado na página 25.
- GLOROT, X.; BENGIO, Y. Understanding the difficulty of training deep feedforward neural networks. In: *In Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS'10)*. Society for Artificial Intelligence and Statistics. [S.l.: s.n.], 2010. Citado na página 51.
- HAMM, R. O. Vacuum tubes vs. transistors - is there an audible difference? *Audio Engineering Society, New York*, 1972. Citado 2 vezes nas páginas 23 e 31.
- HAYKIN, S. *Neural Networks: A Comprehensive Foundation*. Prentice Hall, 1999. ISBN 9780780334946. Disponível em: <<https://books.google.com.br/books?id=3-1HPwAACAAJ>>. Citado 9 vezes nas páginas 24, 47, 48, 49, 50, 52, 60, 61 e 64.
- HELMHOLTZ, H. von. *On the sensations of tone as a physiological basis for the theory of music*. 3. ed. Longmans, Green, e Co., 1895. Acesso em 27/09/2018. Disponível em: <<https://archive.org/details/onsensationsof00helmrich>>. Citado na página 31.
- HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. v. 9, p. 1735–80, 12 1997. Citado 7 vezes nas páginas 24, 59, 60, 64, 65, 66 e 104.
- HOLZMANN, G. Reservoir computing : a powerful black-box framework for nonlinear audio processing. In: . [S.l.: s.n.], 2009. Citado na página 39.
- HORNIK, K.; STINCHCOMBE, M.; WHITE, H. Multilayer feedforward networks are universal approximators. *Neural Networks*, v. 2, n. 5, p. 359 – 366, 1989. ISSN 0893-6080. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0893608089900208>>. Citado 2 vezes nas páginas 24 e 37.
- JAEGER, H. *A tutorial on training recurrent neural networks, covering BPTT, RTRL, EKF and the "echo state network" approach*. 2002. German National Research Center for Information Technology. G. Citado na página 57.
- JJ ELECTRONIC. *ECC83S*. [S.l.]. Disponível em: <https://www.jj-electronic.com/images/stories/product/preamplifying_tubes/pdf/ecc83s.pdf>. Citado 2 vezes nas páginas 73 e 76.
- JORDAN, M. Serial order: a parallel distributed processing approach. technical report, june 1985-march 1986. 5 1986. Citado na página 51.
- KARJALAINEN, M.; PAKARINEN, J. Wave digital simulation of a vacuum-tube amplifier. In: *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*. [S.l.: s.n.], 2006. v. 5, p. V–V. ISSN 1520-6149. Citado 2 vezes nas páginas 33 e 34.
- KEMPERAMPS. *The Kemper Profiler*. c.a. 2016. Kemper Amps' Website. Acessado em 21/09/2018. Disponível em: <<https://www.kemper-amps.com/ profiler/overview>>. Citado na página 30.

KEUNINCKX, L.; DANCKAERT, J.; SANDE, G. V. der. Real-time audio processing with a cascade of discrete-time delay line-based reservoir computers. *Cogn Comput*, 2017. ISSN 1866-9956. Disponível em: <<https://doi.org/10.1007/s12559-017-9457-5>>. Citado na página 39.

KOREN, N. L. *Improved vacuum tube models for SPICE simulations*. 2003. Norman Koren's website. Acesso em 18/10/2018. Disponível em: <http://www.normankoren.com/Audio/Tubemodspice_article.html>. Citado 2 vezes nas páginas 34 e 67.

LJUNG, L. *System identification: theory for the user*. [S.l.]: Prentice-Hall, Inc. Upper Saddle River, NJ, USA ©1986, 1986. Citado 2 vezes nas páginas 43 e 46.

LOTS, I. S.; STONE, L. Perception of musical consonance and dissonance: an outcome of neural synchronization. *Journal of The Royal Society Interface*, The Royal Society, v. 5, n. 29, p. 1429–1434, 2008. ISSN 1742-5689. Disponível em: <<http://rsif.royalsocietypublishing.org/content/5/29/1429>>. Citado na página 31.

MACAK, J.; SCHIMMEL, J. Real-time guitar tube amplifier simulation using an approximation of differential equations. In: *DAFX-10*. [S.l.: s.n.], 2010. Citado 2 vezes nas páginas 34 e 35.

MACAK, J.; SCHIMMEL, J. Real-time guitar preamp simulation using modified blockwise method and approximations. *EURASIP Journal on Advances in Signal Processing*, v. 2011, n. 1, p. 629309, Feb 2011. ISSN 1687-6180. Disponível em: <<https://doi.org/10.1155/2011/629309>>. Citado na página 35.

MACKSEY, K.; WOODHOUSE, W. *The Penguin Encyclopedia of Modern Warfare: 1850 to the present day*. [S.l.]: Viking, 1991. Citado na página 25.

MARSHALL. *The History of: JCM 800 - The Story of a True Amp Legend*. 2017. Marshall Amplification. Acesso em 19/09/2018. Disponível em: <<https://marshall.com/live-for-music/history-of-jcm800>>. Citado na página 28.

MARSHALL. *JCM800 2203-01 Owners Manual*. [S.l.], 2018. Disponível em: <<https://marshall.com/marshall-amps/products/amps/vintage-reissues/jcm800-2203>>. Citado na página 27.

MESA/BOOGIE. *Overview: Over 40 Years of Randall Smith's Innovation in One Amp*. 2018. Mesa Boogie's Website. Acesso em 20/09/2018. Disponível em: <<http://www.mesaboogie.com/amplifiers/electric/mark-series/mark-five/index.html>>. Nenhuma citação no texto.

MESA/BOOGIE. *Power Tubes - 6L6 STR 440 - Matched pair*. 2018. Mesa/Boogie Web Store. Acesso em 20/09/2018. Disponível em: <<https://store.mesaboogie.com/categories/shop/tubes.html>>. Citado 2 vezes nas páginas 26 e 28.

MILLET, P. *The sound of distortion*. 2004. The European Triode Festival. Citado na página 30.

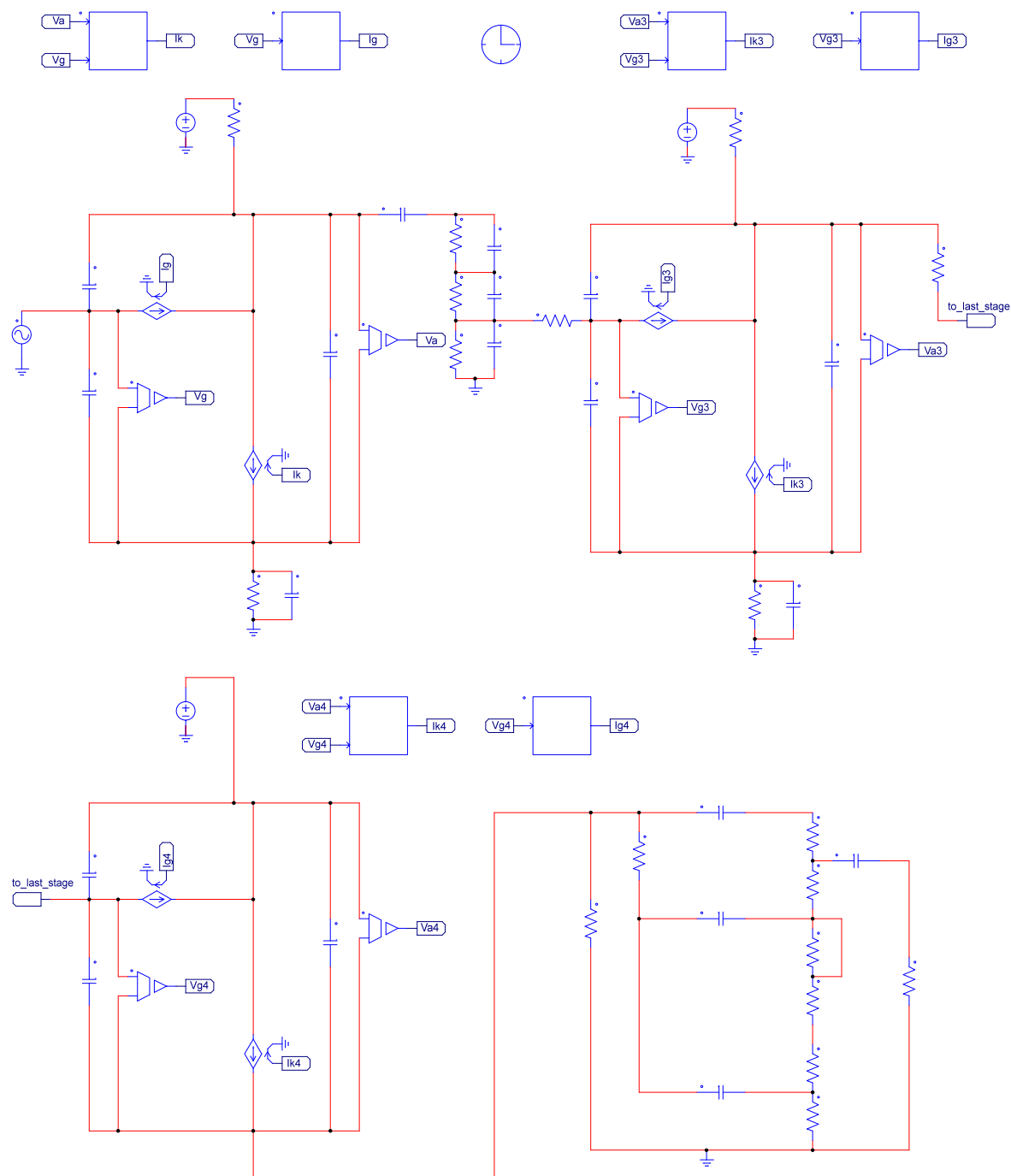
MOZER, M. A focused backpropagation algorithm for temporal pattern recognition. *Complex Systems*, v. 3, 01 1989. Citado na página 65.

MUSICCRITIC. *Top 10 Best Solid State Amps*. 2018. Music Critic's Website. Acesso em 21/09/2018. Disponível em: <<https://musiccritic.com/equipment/amplifiers/best-solid-state-amps/?ved=0ahUKEwihopSemczdAhVBIpAKHXujDHsQypMCCCcwGA>>. Citado 2 vezes nas páginas 28 e 29.

- OLIVEIRA, T.; BARRETO, G.; PASQUAL, A. Review of digital emulation of vacuum-tube audio amplifiers and recent advances in related virtual analog models. *INFOCOMP*, v. 12, n. 1, p. 10–23, 2013. ISSN 1982-3363. Disponível em: <<http://www.dcc.ufla.br/infocomp/index.php/INFOCOMP/article/view/369>>. Citado 2 vezes nas páginas 32 e 33.
- PAKARINEN, J.; KARJALAINEN, M. Enhanced wave digital triode model for real-time tube amplifier emulation. *IEEE Transactions on Audio, Speech, and Language Processing*, v. 18, n. 4, p. 738–746, May 2010. ISSN 1558-7916. Citado na página 33.
- PAKARINEN, J.; YEH, D. T. A review of digital techniques for modeling vacuum-tube guitar amplifiers. *Computer Music Journal*, v. 33, n. 2, p. 85–100, June 2009. ISSN 0148-9267. Citado 3 vezes nas páginas 23, 26 e 32.
- ROSENBLATT, F. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, nov. 1958. Citado na página 47.
- SCHMITZ, T.; EMBRECHTS, J. Real time emulation of parametric guitar tube amplifier with long short term memory neural network. *ARXIV*, 2018. Citado 7 vezes nas páginas 24, 40, 46, 101, 102, 103 e 104.
- SCHMITZ, T.; EMBRECHTS, J.-J. Hammerstein kernels identification by means of a sine sweep technique applied to nonlinear audio devices emulation. *J. Audio Eng. Soc.*, v. 65, n. 9, p. 696–710, 2017. Disponível em: <<http://www.aes.org/e-lib/browse.cfm?elib=19200>>. Citado 2 vezes nas páginas 36 e 46.
- SOLLO. *Amplificadores Valvulados X Transistorizados*. 2016. Sollos Amps' Website. Acesso em 21/09/2018. Disponível em: <<http://www.solloamps.com.br/blog/2016/05/10/amplificadores-valvulados-x-transistorizados/>>. Citado na página 28.
- SUNDBERG, J. E. F.; LINDQVIST, J. Musical octaves and pitch. *The Journal of the Acoustical Society of America*, v. 54, n. 4, p. 922–929, 1973. Disponível em: <<https://doi.org/10.1121/1.1914347>>. Citado na página 30.
- WHITE, P. *Boss Katana Series*. 2017. Sound on Sound's Website. Acesso em 21/09/2018. Disponível em: <<https://www.soundonsound.com/reviews/boss-katana-series>>. Citado na página 29.
- YEH, D. T.; ABEL, J. S.; SMITH, J. O. Automated physical modeling of nonlinear audio circuits for real-time audio effects—part i: Theoretical development. *IEEE Transactions on Audio, Speech, and Language Processing*, v. 18, n. 4, p. 728–737, May 2010. ISSN 1558-7916. Citado na página 34.

APÊNDICES

**APÊNDICE A – CIRCUITO DO CANAL DE BAIXO GANHO DO
PRÉ-AMPLIFICADOR VALVULADO IMPLEMENTADO NO SOFTWARE
POWERSIM®**



**APÊNDICE B – CIRCUITO DO CANAL DE ALTO GANHO DO
PRÉ-AMPLIFICADOR VALVULADO IMPLEMENTADO NO SOFTWARE
POWERSIM®**

