CENTRO UNIVERSITÁRIO FEI

JULIANO ALVES DE OLIVEIRA

EFEITO DE EVENTOS ÚNICOS EM TRANSISTORES MOS: CLASSIFICAÇÃO DOS EVENTOS VIA REDES NEURAIS PROFUNDAS

São Bernardo do Campo

JULIANO ALVES DE OLIVEIRA

EFEITO DE EVENTOS ÚNICOS EM TRANSISTORES MOS: CLASSIFICAÇÃO DOS EVENTOS VIA REDES NEURAIS PROFUNDAS

Dissertação apresentada ao Centro Universitário FEI, como parte dos requisitos necessários para obtenção do título de Mestre em Engenharia Elétrica. Orientado pelo Prof. Dr. Renato Giacomini.

São Bernardo do Campo

2021

Oliveira, Juliano Alves de.

Efeito de eventos únicos em transistores MOS: Classificação dos eventos via redes neurais profundas / Juliano Alves de Oliveira. São Bernardo do Campo, 2021.

155 p. : il.

Dissertação - Centro Universitário FEI. Orientador: Prof. Dr. Renato Camargo Giacomini. Coorientador: Prof. Dr. Marco Antônio Assis de Melo.

1. Single Event Effect. 2. 3N163. 3. MOSFET. 4. Deep Learning. I. Giacomini, Renato Camargo, orient. II. Título.

Elaborada pelo sistema de geração automática de ficha catalográfica da FEI com os dados fornecidos pelo(a) autor(a).

universitário

APRESENTAÇÃO DE DISSERTAÇÃO ATA DA BANCA EXAMINADORA

Programa de Pós-Graduação Stricto Sensu em Engenharia Elétrica

Aluno: Juliano Alves de Oliveira

Título do Trabalho: Efeito de eventos únicos em transistores mos: classificação dos eventos via redes neurais profundas.

Área de Concentração: Nanoeletrônica e Circuitos Integrados

Orientador: Prof. Dr. Renato Camargo Giacomini

Data da realização da defesa: 02/12/2020

ORIGINAL ASSINADA

Avaliação da Banca Examinadora:

O aluno apresentou seu trabalho na primeira parte da reunião, foi arguido e respondeu satisfato-

riamente às questões e sugestões dos examinadores. A banca reuniu-se em separado e deliberou

por unanimidade pela aprovação do candidato.

São Bernardo do Campo, / / .

MEMBROS DA BA	NCA EXAMINADORA
Prof. Dr. Renato Camargo Giacomini	Ass.:
Prof. Dr. Carlos Eduardo Thomaz	Ass.:
Prof. Dr. Vitor Ângelo Paulino de Aguiar	Ass.:
A Banca Julgadora acima-assinada atribuiu ao alun APROVADO	o o seguinte resultado:
VERSÃO FINAL DA DISSERTAÇÃO APROVO A VERSÃO FINAL DA DISSERTAÇÃO EM QUE FORAM INCLUÍDAS AS RECOMENDAÇÕES DA BANCA EXAMINADORA	Aprovação do Coordenador do Programa de Pós-graduação
 	Prof. Dr. Carlos Eduardo Thomaz

Versão 2016

Mestrado

PGE-10

Matrícula: 117309-5

A Deus pelo fôlego de vida, a família pelo apoio e amigos pelo companheirismo.

AGRADECIMENTOS

Agradeço a Deus pela capacidade, saúde e pela vida.

Agradeço ao meu orientador Prof. Dr. Renato C. Giacomini e ao meu co-orientador Prof. Dr. Marco Antônio A. de Melo, por acreditarem, apoiarem, e não desistirem desta empreitada que teve seus altos e baixos.

Agradeço a Prof. Dr. Marcilei Ap. Guazzelli pelos puxões de orelha e pelo apoio técnico.

Agradeço aos colegas do grupo de pesquisa que estão sobre tutela do Prof. Dr. Renato C. Giacomini, que sempre estiveram envolvidos me ajudando em questões técnicas e orientando no desenvolvimento dos próximos passos.

Agradeço aos colegas do LANF sobre a tutela do Prof. Dr. Nilberdo H. Medina, pelo apoio técnico e companherismo durante os experimentos em campo.

Agradeço a todo o time dos laboratórios da FEI pela disponibilidade e pronto atendimento em tudo que precisei.

Agradeço a FEI por toda a estrutura, suporte financeiro e oportunidades oferecidos durante todo o período de desenvolvimento deste trabalho.

Agradeço aos meus pais Eudilson A. Oliveira e Julinda A. Oliveira, e minha irmã Juliana A. Oliveira pelo apoio, e por tudo que fizeram para que eu chega-se até aqui.

Agradeço a minha esposa Ana Paula G. M. de Oliveira, pela paciência, suporte e carinho durante todo esse percurso.

Agradeço aos amigos Felipe G. H. Leite, Thales A. Ribeiro, Rafael Assalti, Diana M. Gustin, Max Pinheiro Jr. pelo apoio e ajuda em diversos momentos durante o desenvolvimento deste traqbalho.

"I still haven't found what I'm looking for"

RESUMO

Dispositivos eletrônicos são suscetíveis a defeitos causados por radiação ionizante, e o uso destes dispositivos é cada vez mais requisitado em aplicações embarcadas que operam em ambientes agressivos (presença de radiação) como o espaço, reatores nucleares e aceleradores de partículas. Entre os defeitos mais danosos estão os *Single Event Effects* (SEE). O efeito é causado por uma única partícula ionizada que, dependendo de diversos fatores, pode causar inversões lógicas em dispositivos eletrônicos digitais, ou até mesmo tornar o dispositivo inoperante. O estudo desses fenômenos é de grande importância na criação de tecnologia nacional, pois são requisitos básicos para gerar componentes resistentes à radiação. Através de experimentos inéditos no Brasil, envolvendo o Projeto CITAR (Circuitos Integrados Tolerantes à Radiação), criou-se o ambiente adequado para a realização destes estudos, pois para a reprodução destes fenômenos é necessário o uso de um acelerador de partículas que seja capaz de gerar feixes de íons pesados com baixo fluxo.

Neste trabalho são avaliados os resultados obtidos do experimento de radiação de partículas ionizantes em um transistor MOSFET tipo P, incluindo a criação de uma representação simulada do dispositivo real, através da ferramenta SENTAURUS. Foram simuladas emissões de íons pesados no componente com as mesmas características dos feixes utilizados em laboratório, com a expectativa de obter-se a mesma reposta gerada pelo dispositivo real.

Por fim, através de técnicas de aprendizado de máquina, foi criado um algoritmo capaz de classificar os diferentes eventos registrados durante os experimentos de campo, bem como avaliar sinais espúrios que compõe o dado obtido.

Como resultado das simulações, aproximamos a simulação do dispositivo 3N163 as características elétricas apresentadas pelos dispositivos reais, e através do treinamento de uma rede neural profunda utilizando os dados medidos em campo capaz de classificação de 97% de acurácia.

Palavras-chave: Single Event Effect, 3N163, MOSFET, Deep Learning

ABSTRACT

Electronic devices are susceptible to defects caused by ionizing radiation, and the use of these devices is increasingly required in embedded applications that operate in aggressive environments (presence of radiation) such as space, nuclear reactors and particle accelerators. Among the most damaging defects are *Single Event Effects* (SEE). The effect is caused by a single ionized particle that, depending on several factors, can cause logical inversions in digital electronic devices, or even render the device inoperative. The study of these phenomena is of great importance in the creation of national technology, as they are basic requirements for generating resistant components radiation. Through unprecedented experiments in Brazil, involving the CITAR Project (Integrated Circuits Tolerant to Radiation), the appropriate environment was created to carry out these studies, since for the reproduction of these phenomena it is necessary to use a particle accelerator that is capable of generate heavy ion beams with low flux.

In this work, the results obtained from the ionizing particle radiation experiment in a MOSFET p-type transistor are evaluated, including the creation of a simulated representation of the real device, using the SENTAURUS tool. Emissions of heavy ions in the component were simulated with the same characteristics as the ion beams used in the laboratory, with the expectation of obtaining the same response generated by the real device.

Finally, through machine learning techniques, an algorithm was created capable of classifying the different events recorded during field experiments, as well as evaluating spurious signals that make up the data obtained.

As a result of the simulations, we approximate the simulation of the 3N163 device to the electrical characteristics presented by the real devices, and through the training of a deep neural network using the data measured in the field capable of classifying 97% of accuracy.

Keywords: Single Event Effect, 3N163, MOSFET, Deep Learning

LISTA DE ILUSTRAÇÕES

Figura 1	_	Níveis de energia permitidos para um átomo, uma molécula hipotética e	
		um cristal de lítio.	24
Figura 2	_	Bandgap entre bandas de condução e valência.	25
Figura 3	_	Esquemático com os diagramas de bandas de energia de um material iso-	
		lante, semicondutor, semi-metal e metal.	26
Figura 4	_	Esquemático com os diagramas de bandas de energia de um material iso-	
		lante, semicondutor, semi-metal e metal, incluindo o valor no nível de fermi	
		e suas relações, para cada caso.	27
Figura 5	_	Promoção do elétron da camada de valência para a camada de condução.	28
Figura 6	_	Inserção de um átomo de Fosforo na estrutura cristalina do Silício. O efeito	
		desta inserção é a "doação"de um elétron na estrutura	29
Figura 7	_	Inserção de um átomo de Boro na estrutura cristalina do Silício. O efeito	
		desta inserção é a coleta de um elétron da estrutura, criando uma lacuna na	
		estrutura.	30
Figura 8	_	Migração de um elétron do nível de energia E_d para a banda de condução	31
Figura 9	_	Representação de um MOSFET canal N ilustrando as cargas aprisionadas	
		no óxido após radiação. Do lado esquerdo o dispositivo operando em modo	
		normal, e do lado direito após a radiação.	32
Figura 10	_	Diversos defeitos criados pelo deslocamento de átomos	33
Figura 11	_	Simples representação dos cintos de radiação em volta da Terra	34
Figura 12	_	Geração de cargas devido a colisão de uma partícula ionizante em uma	
		junção P-N	35
Figura 13	_	Geração de cargas devido a colisão de uma partícula de alta energia (efeitos	
		de segunda ordem)	35
Figura 14	_	Coleta de cargas através de <i>funneling</i> e difusão	36
Figura 15	_	Seção de choque do SEE em função do Linear Energy Transfer (LET)	37
Figura 16	_	Corrente em função do tempo de coleta de cargas (t_c)	40
Figura 17	_	A esquerda: Dados lineares aproximados por uma função linear. A di-	
		reita: Uma função linear aproximando dados quadráticos	45
Figura 18	_	Seleção de partições para treinamento e testes, validação cruzada k -partições	
		$(k=5) \dots \dots \dots \dots \dots \dots \dots \dots \dots $	47

Figura 19	_	Gráfico de pares mostrando a dispersão de cada variável do conjunto de	
		dados Iris para a espécie Setosa vs as demais	48
Figura 20	_	Árvores de decisão iniciada com uma variável de menor entropia	51
Figura 21	_	Árvores de decisão iniciada com uma variável de maior entropia	51
Figura 22	_	Um conjunto de dados em 3D projetado em um sub-espaço 2D	54
Figura 23	_	Novo conjunto de dados projetado em 2D	54
Figura 24	_	Representação de duas componentes de uma PCA para dados de uma gaus-	
		siana bivariada centrada em (0,0)	55
Figura 25	_	Representação matemática de um neurônio artificial	59
Figura 26	_	Rede neural <i>feedfoward</i>	60
Figura 27	_	Mapeamento da saída da função XOR para duas variáveis de entrada X_1 e	
		X_2	62
Figura 28	_	Erro de treinamento e validação ao longo do tempo (epochs) e ponto ótimo	
		de parada	66
Figura 29	_	Um exemplo de convolução 2D	68
Figura 30	_	Exemplo de filtro max pooling e average pooling para um tamanho $2x2$	70
Figura 31	_	Representação gráfica da equação de estados de um sistema dinâmico des-	
		dobrado. Cada nó representa um estado, em um dado tempo t e a função f	
		que mapea o estado $t \text{ em } t + 1$	72
Figura 32	_	Representação gráfica de uma rede neural recorrente sem saída. A caixa	
		preta representa um atraso de $t + 1$ no estado atual $\ldots \ldots \ldots \ldots \ldots$	72
Figura 33	_	Representação de uma rede neural completa e sua representação desdo-	
		brada no tempo.	73
Figura 34	_	Representação de uma rede neural completa e sua representação desdo-	
		brada no tempo.	76
Figura 35	_	Visão geral do acelerador Pelletron.	78
Figura 36	_	Esquemático da configuração experimental usadas nas radiações	79
Figura 37	_	Íons pesados utilizados nos experimentos. LET em função da profundidade	
		no silício	79
Figura 38	_	Visão geral da nova linha SAFIIRA	80
Figura 39	_	Esquemático do sistema de polarização do transistor 3N163	82
Figura 40	_	Dispositivo sob teste na câmara de vácuo.	82

Figura 41	_	Visualização esquemática do arranjo utilizado para a realização dos testes	
		de SEE	83
Figura 42	_	Disposição do Dispositive Under Test (DUT) na câmera utilizado no se-	
		gundo experimento.	84
Figura 43	_	Detalhamento do dispositivo desencapsulado	85
Figura 44	_	Simulação da incidência de diversas partículas ionizadas em uma estrutura	
		metal/óxido de silício/silício	86
Figura 45	_	Gráfico demostrando a perda de energia de íons ^{16}O com energia de $63 MeV$	
		em um corpo composto por metal/óxido de silício/silício	87
Figura 46	_	Visualização esquemática do arranjo utilizado para a realização dos testes	
		de SEE	89
Figura 47	_	Visualização esquemática do arranjo utilizado para a realização dos testes	
		de SEE	90
Figura 48	_	Visualização da primeira estrutura do dispositivo 3N163 gerada	91
Figura 49	_	Corte ao transversal detalhando a estrutura de grade adota para as simulações.	92
Figura 50	_	Uma amostra de cada íon medido durante os experimentos	93
Figura 51	_	Proporção de cada elemento no total de 123000 amostras. Cada quadrado	
		representa aproximadamente 1500 amostras	94
Figura 52	_	Média das amostras para o Íon de prata e o desvio padrão de cada ponto	95
Figura 53	_	Dados gerados a partir de um amostra real	96
Figura 54	_	Rede neural final da arquitetura DeepConvLSTM	99
Figura 55	_	Gráfico de diversos $I_D x V_G$ para um valor de V_D =-0,1 V	02
Figura 56	_	Gráfico de diversos $I_D x V_G$ para um valor de V_D =-1,0 V	02
Figura 57	_	Gráfico da média de $I_D \mathbf{x} V_G$ para um valor de V_D =-0,1V	03
Figura 58	_	Gráfico $I_D x V_G$ mostrando uma série de ajustes realizados nos parâmetros	
		do modelo físico para um valor de V_D =-0,1V	05
Figura 59	_	Gráfico comparando os valores $I_D x V_G$ de referência com o real para um	
		valor de V_D =-1V	05
Figura 60	_	Valores da corrente I_D ao longo do tempo para a emissão de um íon de	
		carbono nas regiões de porta e dreno do dispositivo 3N163	07
Figura 61	_	Valores da corrente I_D ao longo do tempo para a emissão de um íon de	
		oxigênio nas regiões de porta e dreno do dispositivo 3N163	07

_	Valores da corrente I_D ao longo do tempo para a emissão de um íon de
	silício nas regiões de porta e dreno do dispositivo 3N163
_	Valores da corrente I_D ao longo do tempo para a emissão de um íon de
	cloro nas regiões de porta e dreno do dispositivo 3N163
_	Resumo das correntes I_D 's ao longo do tempo para a um conjunto de íons
	na região de porta do dispositivo 3N163
_	Resumo das correntes I_D 's ao longo do tempo para a um conjunto de íons
	na região de dreno do dispositivo 3N163 109
_	Matriz de confusão do resultado da classificação no conjunto de dados de
	teste para o modelo SVM 110
_	Matriz de confusão do resultado da classificação no conjunto de dados de
	teste para o modelo DeepConvLSTM 111
_	Dados simulados usados para validação com os modelos de redes neurais . 112
_	Tempo de treinamento para 10 rodadas diferentes com 300 epochs em cada
	rodada para cada modelo
_	Tempo de treinamento de cada modelo
_	Influência de cada ponto para uma correta classificação de uma amostra de
	Silício utilizando a arquitetura DeepConvLSTM 117
_	Influência de cada ponto para uma correta classificação de uma amostra de
	Silício utilizando a arquitetura DeepConvGRU 117
_	Influência de cada ponto para uma correta classificação de uma amostra de
	Silício utilizando a arquitetura DeepConvBiLSTM 118
_	Influência de cada ponto para uma correta classificação de uma amostra de
	Silício utilizando a arquitetura DeepConvBiGRU
	Influência de code ponte pono uma cometa classificação de sincl sinculado
-	influencia de cada ponto para uma correta classificação do sinai simulado

LISTA DE TABELAS

Tabela 1	-	Energia do feixe, profundidade de penetração e LET superficial	80
Tabela 2	_	Energia do feixe, profundidade de penetração e LET no silício	81
Tabela 3	_	Energia do feixe, profundidade de penetração e LET no silício para dife-	
		rentes regiões do dispositivo	87
Tabela 4	_	Energia do feixe, profundidade de penetração e LET no silício utilizados	
		em simulação	106
Tabela 5	_	Resumo dos resultados de cada modelo em diferentes métricas	111
Tabela 6	_	Predição dos dados simulados utilizando-se as quatro arquiteturas propostas	113
Tabela 7	_	Teste ANOVA das acurácias dos modelos DeepConvGRU, DeepConvBiLSTM	ĺ
		e DeepConvBiGRU em relação a arquitetura DeepConvLSTM	115

LISTA DE ALGORITMOS

Algoritmo 1 -	- backpropagation		• •	• •	•	•••	•	•••	•	• •	•••	•	•	•	• •	•	•	•••	•	•	•	64
Algoritmo 2 -	- Janelas deslizante	s TS			•		•					•	•				•			•	•	100

LISTA DE ABREVIATURAS

CMOS	Complementary Metal Oxide Semiconductor
CNN	Convolutional Neural networks
DD	Displacement Damage
DUT	Dispositive Under Test
FCN	Fully Convolutional Network
FEI	Centro Universitário da FEI
FPGA	Field Programmable Gate Array
GRU	Gated Recurrent Unit
LAFN	Laboratório Aberto de Física Nuclear
LDA	Linear Discriminant Analysis
Leaky ReLU	Leaky Rectified Linear Units
LET	Linear Energy Transfer
LET_{TH}	Linear Energy Transfer threshold
LSI	Large Scale Integration
LSTM	Long Short Term Memory
MBU	Multiple Bit Upset
MLP	Multilayer Perceptron
MOS	Metal Oxide Semiconductor
MSV	Mean Square Voltage
NMOS	Type N Metal Oxide Semiconductor
PCA	Principal Component Analysis
PhuMob	Philips unified mobility model
PMOS	Type P Metal Oxide Semiconductor
ReLU	Rectified Linear Units
SDE	Sentaurus Structure Editor
SEB	Single Event Burnout
SEE	Single Event Effects
SEFI	Single Event Functional Interrupts
SEGR	Single Event Gate Rupture
SEL	Single Event Latch-up
SET	Single Event Transients

SEUSingle Event UpsetsSHESingle Event Hard ErrorsSRHShockley–Read–HallSVMSupport Vector MachineTCADTechnology Computer-Aided DesignTIDTotal Ionizing DoseTRIMTransport of Ions in Matter

LISTA DE SÍMBOLOS

E_c	Máxima energia da banda de condução, eV
E_d	Energia do nível permitido dentro do bandgap, eV
E_f	Energia do nível de Fermi, eV
E_g	Band gap ou lacuna de energia, eV
E_i	Energia do nível intrínse co de um semicondutor, eV
Δ_E	Perda de energia da partícula dentro do material, $\frac{eV}{a}$
E_v	Máxima energia da banda de valência, eV
I_D	Corrente de dreno, A
N_a	Concentração de dopantes do tipo P, cm^{-3}
N_d	Concentração de dopantes do tipo N, cm^{-3}
n_i	Concentração intrínseca de portadores, cm^{-3}
ϕ_F	Potencial de Fermi, V
ϕ_{FN}	Potencial de Fermi no material tipo N, V
ϕ_{FP}	Potencial de Fermi no material tipo P, V
Φ	Fluência do feixe acumulado, $\frac{partículas}{cm^2}$
Q_c	Carga crítica, pC
σ	Seção de choque, cm^2
t_c	Tempo de coleta de cargas, s
V_D	Tensão de dreno, V
VDS	Tensão entre dreno e fonte, V
V_G	Tensão de Porta,V
VGS	Tensão entre porta e fonte, V
V_{th}	Tensão de limiar, V
X_{OX}	Espessura da camada de óxido, cm

SUMÁRIO

1	ΙΝΤΡΟΝΙΟÃΟ	$\gamma\gamma$
1		22
2		23
2.1	DISPOSITIVOS SEMICONDUTORES	23
2.1.1	Bandas de energia, banda de valência e de condução	23
2.1.2	Semicondutor intrínseco	27
2.1.3	Semicondutor extrínseco	28
2.2	RADIAÇÃO	31
2.2.1	Efeitos de radiação ionizante em dispositivos eletrônicos	31
2.2.1.1	TID (Total Ionizing Dose) - Dose ionizante total	31
2.2.1.2	Displacement Damage (DD) - Dano por deslocamento	33
2.2.1.3	SEE (Single Event Effects) - Efeitos por Eventos Isolados	34
2.3	DETECÇÃO DE PARTÍCULA	39
2.4	ALGORITMOS DE APRENDIZADO	42
2.4.1	O que é aprendizado de máquina?	42
2.4.1.1	Tarefas T de algoritmos de aprendizado	42
2.4.1.2	Medidas de performance P de algoritmos de aprendizado $\ldots \ldots \ldots \ldots$	43
2.4.1.3	Experiência E de algoritmos de aprendizado	44
2.4.2	Capacidade, overfitting e underfitting	44
2.4.3	Hiperparâmetros e dados de validação	45
2.4.4	Validação cruzada	46
2.4.5	Algoritmos supervisionados	47
2.4.5.1	Árvores de decisão	47
2.4.6	Algoritmos não-supervisionados	52
2.4.7	Principal Component Analysis (PCA)	52
2.4.8	Motivação aprendizado profundo	55
2.4.8.1	Maldição da alta dimensionalidade	56
3	APRENDIZADO PROFUNDO	58
3.1	REDES NEURAIS	58
3.1.1	Funções de ativação	60
3.1.2	Algoritmo backpropagation e aprendizado da rede neural	61
3.2	REGULARIZAÇÃO PARA APRENDIZADO PROFUNDO	64

3.2.1	Aumento do conjunto de dados (Dataset Augmentation)	65
3.2.2	Parada precoce (<i>Early stopping</i>)	65
3.2.3	Métodos Ensemble	66
3.2.4	Dropout	67
3.3	Redes Neurais Convolucionais	67
3.3.1	Pooling	69
3.4	REDES NEURAIS RECORRENTES	70
3.4.1	Redes neurais recorrentes e o Desdobramento	71
3.4.2	O problema do aprendizado de longo prazo	74
3.4.3	Long Short Term Memory (LSTM)	74
3.4.4	Gated Recurrent Unit (GRU)	76
3.4.5	Proposta	77
4	MATERIAIS E MÉTODOS	78
4.1	ACELERADOR PELLETRON	78
4.1.1	Configuração experimental	78
4.1.2	Dispositivos sob teste	81
4.2	SIMULADORES	85
4.2.1	SRIM/TRIM	85
4.2.2	Sentaurus	88
4.2.3	Simulação	89
4.2.3.1	Dispositivo	90
4.2.3.2	Grade	91
4.3	ANÁLISE DE DADOS	92
4.4	TREINAMENTO REDE NEURAL	96
4.4.1	Motivação	96
4.4.2	Arquitetura da rede neural	97
4.4.3	Arquiteturas propostas	98
4.4.4	Treinamento do Modelo	98
4.5	INTERPRETABILIDADE	99
5	RESULTADOS	101
5.1	SIMULAÇÕES	101
5.1.1	Ajuste da estrutura	101
5.1.2	Simulação <i>Mixed mode</i> do circuito de polarização.	106

5.1.3	Simulação da emissão de partícula ionizante
5.2	MODELOS DE CLASSIFICAÇÃO
5.2.1	Modelo de classificação no cenário dos dados simulados
5.3	COMPARAÇÃO DEEPCONVLSTM, DEEPCONVGRU, DEEPCONVBILSTM
	E DEEPCONVBIGRU
5.4	INTERPRETABILIDADE DAS REDES NEURAIS
5.4.1	Interpretabilidade dos resultados para os sinais simulados
6	CONCLUSÕES
	REFERÊNCIAS
	APÊNDICE A – Artigo - XIII Encontro Acadêmico de Modelagem Com-
	putacional
	APÊNDICE B – Artigo - 2018 IEEE 19th Latin-American Test Symposium
	(LATS)

1 INTRODUÇÃO

A eletrônica hoje em dia é vital em diversas aplicações e áreas, por conta disso, os dispositivos eletrônicos devem ser robustos e capazes de operar em condições extremas de calor, frio e até mesmo em ambientes mais agressivos que apresentam radiação. Dispositivos eletrônicos que operam nessas condições (aplicações aeroespaciais, reatores nucleares, aceleradores de partículas entre outros) podem sofrer uma série de danos causados pelas partículas ionizantes presentes nestes ambientes, sendo o objeto de estudo deste trabalho, o fenômeno conhecido como *Single Event Effects* (SEE) Duzellier (2005). Este tipo de evento é causado por um única partícula ionizante que é capaz de gerar desde uma inversão lógica em dispositivos digitais até tornar o componente inoperante (MASSENGILL; TUINENGA, 2008). Por conta dos grandes avanços alcançados continuamente no campo da eletrônica, que dia após dia, alcança dimensões cada vez menores nos dispositivos Salahuddin, Ni e Datta (2018), e como consequência desta evolução, o componente se torna mais suscetível as influências da radiação Zheng et al. (2017), é de grande importância criar estudos sistemáticos sobre estes fenômenos, pois conhecendo melhor os mecanismos e efeitos decorrentes do evento, é possível criar estrategias para contornar estes problemas em aplicações onde este tipo de condição de operação é presente.

A proposta deste trabalho inclui utilizar um transistor pMOS 3N163. Este dispositivo, em especial, já se mostrou ser um bom dosímetro de raios-X de baixo custo (OLIVEIRA et al., 2014), apresentou potencial para a detecção de SEE (MEDINA, N. H. et al., 2015).Com o auxílio de ferramentas de simulação, será replicado o ambiente utilizado nos experimentos em campo, com o mesmo dispositivo e com as mesmas condições dos feixes de radiação, porém em ambiente simulado. Com os resultados da simulação é possível entender melhor as características dos sinais obtidos. Também são utilizadas técnicas de análise de dados, visando criar modelos de aprendizado de máquina com a finalidade de classificar os dados existentes, extraindo parâmetros e características de cada curva, e confrontar os resultados obtidos através da simulação com os modelos criados a partir dos dados reais.

Este trabalho está dividido em mais 4 capítulos. O capítulo 2, introduz todos os conceitos e referências necessárias para a elaboração destes estudos. No capítulo 3, são descritos os procedimentos experimentais e o materiais utilizados para a coleta dos dados em campo, o detalhamento dos experimentos realizados e as configurações utilizadas nos simuladores. No capítulo 4, são mostrados os resultados obtidos e as discussões e por fim as conclusões.

2 REVISÃO BIBLIOGRÁFICA

2.1 DISPOSITIVOS SEMICONDUTORES

Nesta seção serão introduzidos conceitos básicos sobre a física de materiais semicondutores e os mecanismos que cercam o funcionamento de dispositivos eletrônicos.

2.1.1 Bandas de energia, banda de valência e de condução

Em um átomo, os elétrons apresentam uma força de atração eletrostática do núcleo. Os elétrons que ocupam as camadas interiores experimentam uma força maior que aqueles que ocupam níveis superiores. Estas camadas apresentam patamares de energia discretos, portanto existem regiões definidas que um elétron pode ocupar. Os portadores das camadas mais externas são aqueles que participam da condução elétrica e nas reações químicas e são conhecidos como elétrons de condução. Quando dois ou mais átomos, se unem os seus os seus orbitais eletrônicos mais externos começam a se sobrepor, compartilhando a ocupação nos níveis de energia e dando origem a condução, por exemplo, o átomo de lítio tem configuração eletrônica $1s^2 2s^1$, numa molécula contendo dois átomos de Li, o nível $1s^2$ estará totalmente preenchido segundo o principio de exclusão de Pauli Pauli (1925), isto é com dois elétrons de subníveis conhecidos como bandas de energias. Num cristal, esta banda de energia é superposta permitindo o deslocamento dos portadores através da estrutura(COLINGE; COLINGE, 2003; SZE.; NG, 2007). A Figura 1 mostra um esquema dos níveis de energias permitidos para diferentes situações de uma molécula de Lítio.

Figura 1 – Níveis de energia permitidos para um átomo, uma molécula hipotética e um cristal de lítio.



Fonte: Autor "adaptado de"Colinge e Colinge, 2003, p.7

Como dito anteriormente, as camadas externas de um átomo são as que participam na condução elétrica e são as que dão origem as reações químicas. Os níveis de energia internos não participam por conta da alta atração eletrostática com o núcleo. A primeira banda de energia é fortemente ligada ao seu átomo, já as últimas camadas são responsáveis pela interligação entre os átomos de um cristal. Sendo assim, no exemplo do átomo de Li, o nível $1s^2$ não participa nas iterações atômicas, enquanto que no último nível, o 2, cada subnível tera um papel crucial no cristal, o 2s eh chamado banda de valência E_v , e a próxima banda de energia, a 2p eh conhecida como banda de condução E_c . Em baixas temperaturas (T = 0K), um semicondutor não apresenta nenhum elétron em sua banda de condução, porém com a acréscimo de temperatura alguns elétrons da camada de valência podem adquirir energia térmica o suficiente para migrarem entre essas duas camadas, deixando de participar da sua função inicial, que era fazer ligações entre os átomos, e passa a se mover livremente pela camada de condução. Esta diferença de energia, que vai do topo da camada de valência à base da de condução, é conhecida como "banda proibida"ou *bandgap* e é simbolizada por E_g conforme fig.2 (COLINGE; COLINGE, 2003; SZE.; NG, 2007)).

Figura 2 – Bandgap entre bandas de condução e valência.



Fonte: Autor "adaptado de"Colinge e Colinge, 2003, p.18

As diferenças entre um material semicondutor, isolante e condutor, são puramente quantitativas em termos de E_g (fig. 3). Para um material semicondutor o valor de E_g é menor que 2eVbandgap, assim a energia térmica gerada pela temperatura ambiente ou a excitação pelos fótons na frequência da luz visível podem conferir energia suficiente para fazer um elétron na camada de valência migrar para a de condução. O silício possui $E_g = 1,12eV$ e o germânio $E_g = 0,67eV$ (STREETMAN; BANERJEE et al., 1995), os dois elementos mais comuns em dispositivos eletrônicos. Materiais isolantes como o óxido de silício (SiO_2), diamante e nitreto de silício(Si_3N_4) apresentam valores de E_g , respectivamente, 9,0eV, 5,47eV e 5,0eV (VELLA et al., 2011; STREETMAN; BANERJEE et al., 1995), valores superiores aos presentes em materiais semicondutores, nestes materiais a energia térmica e excitação por luz não conseguem excitar o suficiente um elétron para migrar da camada de valência para a camada de condução. Por fim, temos o materiais condutores, que apresentam valores de E_g menores ou iguais a zero, ou seja, a banda de condução e valência estão sobrepostas ou uma esta imediatamente acima da outra (COLINGE; COLINGE, 2003; SZE.; NG, 2007).



Figura 3 – Esquemático com os diagramas de bandas de energia de um material isolante, semicondutor, semi-metal e metal.

Fonte: Autor "adaptado de"Owens, 2012, p.6

Um outro conceito que deve ser introduzido é o do Nível de Fermi E_f ((DIRAC, 1926)) que representa a máxima energia de um elétron no material em T = 0K (Figura 4). Nesta temperatura todos os níveis permitidos que estão abaixo de E_f são ocupados por elétrons e todos aqueles que estão acima não possuem elétrons. Uma definição alternativa diz que E_f é um nível de energia que tem probabilidade de 50% de conter elétrons mesmo que este nível esteja localizado entre as bandas de valência e condução (COLINGE; COLINGE, 2003; SZE.; NG, 2007). Figura 4 – Esquemático com os diagramas de bandas de energia de um material isolante, semicondutor, semi-metal e metal, incluindo o valor no nível de fermi e suas relações, para cada caso.



Fonte: Autor "adaptado de"Owens, 2012, p.6

2.1.2 Semicondutor intrínseco

Semicondutores considerados intrínsecos são aqueles que apresentam, em sua grande maioria, portadores livres (elétrons e lacunas¹) originados pelos átomos do próprio semicondutor. Caso um elétron adquira energia suficiente para vencer o *bandgap* e ser promovido da banda de valência para a de condução, gera um equivalente na banda de valência, ou seja, deixa um "buraco"ou lacuna na estrutura (figura 5), desta forma sempre que houver uma promoção de um elétron para a banda de condução haverá uma lacuna correspondente na banda de valência. Em semicondutores intrínsecos o nível de Fermi é representado por E_i , que pode ser calculado pela equação 1. A concentração de portadores em um material semicondutor intrínseco n_i , que no silício tem valor de $1,45x10^{10}cm^{-3}$ na temperatura de T = 300K, e uma função da temperatura e do material através de E_g (COLINGE; COLINGE, 2003; SZE.; NG, 2007).

¹As lacunas são os espaços criados pela ausência dos elétrons na estrutura de bandas. É o equivalente ao elétron em termos de condução, porém apresenta mobilidade reduzida em comparação ao elétron.

$$E_i = \frac{E_c - E_v}{2} \tag{1}$$



Figura 5 – Promoção do elétron da camada de valência para a camada de condução.

Fonte: Autor "adaptado de"Colinge e Colinge, 2003, p.33

2.1.3 Semicondutor extrínseco

Um semicondutor é chamando extrínseco a partir do momento que o material intrínseco é dopado de forma controlada com impurezas. Esta mudança na estrutura do cristal devida a introdução de outros materiais conhecidos como dopantes, impurezas dopantes ou átomos dopantes, permite controlar as características elétricas do material e direciona-las para exercer uma determinada função. Um exemplo claro disto é o silício, amplamente usado na indústria de semicondutores, e que apresenta um grande nível de pureza, maior que 99,99%, tem uma condutividade maior quando dopado (GROSSO; PARRAVICINI, 2000). As impurezas são divididas em duas classes, impurezas doadoras e impurezas aceitadoras, na tabela periódica, os materiais utilizados para este processo estão localizadas nos grupos III (boro) e V (fósforo e arsênio), a introdução desses novos materiais na estrutura do cristal de silício, altera suas características elétricas. As impurezas doadoras, grupo V da tabela periódica, apresentam cinco elétrons na sua camada de valência. Dentro da estrutura do cristal de silício, o novo átomo introduzido realiza quatro ligações compartilhando quatro elétrons com os átomos de silício nas vizinhanças. A energia térmica gerada pela temperatura ambiente é grande o suficiente para promover o quinto elétron do átomo dopante para a banda de condução, desta forma este portador está livre para se mover no cristal e contribuir para a condução elétrica. O átomo dopante, neste caso, faz uma "doação"para a estrutura cristalina, introduzindo um novo elétron por átomo de impureza inserido, por conta disso recebe o nome de dopante doador (Figura 6). A concentração de dopantes doadores é chamado de N_d , e os semicondutores que são dopados com este tipo de impurezas é dado o nome de semicondutores *tipo N*, esta classificação se dá por conta da predominância do tipo de portador, que neste caso é o elétron que contem carga negativa (COLINGE; COLINGE, 2003; SZE.; NG, 2007).

Figura 6 – Inserção de um átomo de Fosforo na estrutura cristalina do Silício. O efeito desta inserção é a "doação"de um elétron na estrutura.



Fonte: Autor "adaptado de"Owens, 2012, p.38

De forma similar, a inserção das impurezas aceitadoras, grupo III da tabela periódica, resultará no aumento de lacunas nos cristal, pois a impureza inserida apresenta apenas três elétrons em sua camada de valência e consegue capturas facilmente um elétron, para formar uma quarta ligação mais estável, criando um átomo de impureza eletricamente negativo e imóvel. A captura do elétron pelo átomo de impureza, forma a lacuna na camada de valência, que pode se mover livremente no cristal e participar da condução elétrica. O átomo dopante, neste caso, cria o buraco que pode ser preenchido por um elétron, por conta disso recebe o nome de dopante aceitador (fig.7). A concentração de dopantes aceitadores é chamado de N_a . Cristais que são dopados com este tipo de impurezas aceitadoras são conhecido como semicondutores *tipo P*, esta classificação se dá por conta da predominância do tipo de portador, que neste caso são as lacunas que apresentam cargas positivas ((COLINGE; COLINGE, 2003),(SZE.; NG, 2007)).

Figura 7 – Inserção de um átomo de Boro na estrutura cristalina do Silício. O efeito desta inserção é a coleta de um elétron da estrutura, criando uma lacuna na estrutura.



Fonte: Autor "adaptado de"Owens, 2012, p.38

A inserção de impurezas dopantes, doadoras ou aceitadoras, dão origem a um novo nível de energia permitido no *bandgap*, o nível E_d figura 8 . No caso de materiais do grupo V (arsênio ou fósforo), o nível E_d que surge fica a alguns meV abaixo da banda de condução Golin (1965) e a baixas temperaturas contém os elétrons que são inseridos pelas as impurezas dopantes, porém a energia térmica gerada em temperatura ambiente, é suficiente para romper sua ligação com o átomo de impureza, em outras palavras, o átomo extra é promovido de E_d para E_c . O mesmo ocorre para os dopantes aceitadores, um novo nível energético é criado dentro do *bandgap*, porém este está localizado acima de E_v , neste caso os elétrons no topo de E_v apresentam energia térmica suficiente para migrarem para o nível E_d gerado, deixando em seu lugar as lacunas, que ficam livres para se mover dentro do cristal (COLINGE; COLINGE, 2003; SZE.; NG, 2007).



Figura 8 – Migração de um elétron do nível de energia E_d para a banda de condução.

Fonte: Autor "adaptado de"Colinge e Colinge, 2003, p.33

2.2 RADIAÇÃO

Nesta seção serão abordados os aspectos relacionados aos fenômenos que ocorrem em dispositivos semicondutores decorrente a interação de partículas ionizantes com as estrutura dos materiais em estudo, seus mecanismos e classificação dos fenômenos.

2.2.1 Efeitos de radiação ionizante em dispositivos eletrônicos

Para o estudo dos efeitos da radiação ionizante em dispositivos eletrônicos, primeiro é necessário definirmos um termo que é usualmente utilizado para comparar a efeito da trilha de ionização de várias partículas nestes dispositivos. O *Linear Energy Transfer (LET)*, representa a perda de energia da partícula, e sua unidade de medida é $MeV * \mu m^{-1}$ ou $MeV * cm^2/mg$ quando está normalizado para a massa específica do material de absorção.

2.2.1.1 TID (Total Ionizing Dose) - Dose ionizante total

A interação mais comum para partículas carregadas no material semicondutor é com os elétrons, pois a maior parte do volume atômico é ocupado pelos elétrons (JOHNSTON, A., 2010). Neste processo, uma pequena quantidade de energia é transferida para um elétron via partícula irradiada, a energia é absorvida pelo elétron na banda de valência, causando sua

promoção para a banda de condução, gerando uma lacuna correspondente na banda de valência. A passagem desta radiação ionizante, seja ela um fóton, elétron, nêutron, próton ou íon pesado, cria um par elétron-lacuna no material. A ionização ocorre, predominantemente, em regiões de isolantes, como os óxidos usados em dispositivos semicondutores, mas também ocorrem nas regiões semicondutoras. Apesar de serem péssimos condutores, as cargas provenientes da ionização conseguem se deslocar depois de sua geração. A mobilidade dos elétrons é grande o suficiente para, na presença de um fraco campo elétrico, se deslocarem dentro do isolante, e em contrapartida, temos as lacunas com baixa mobilidade, que apresentam mais dificuldade de locomoção, tendendo a ficar no mesmo local (em temperaturas menores do que 100k para dióxido de silício) ou gradualmente migrar para fora do isolante em temperatura ambiente, sendo o processo de migração dependente da propriedade do material, magnitude de qualquer campo elétrico na região e do processo de recombinação de elétron-lacuna. Após a migração, próximo da região de interface² ($S_i - S_iO_2$) as armadilhas aprisionam as lacunas na região. O acumulo destas cargas armadilhadas dentro do material isolante (S_iO_2), próxima a região de interface pode ser visto em fig.9 (OLDHAM; MCLEAN, 2003; JOHNSTON, A., 2010)).

Figura 9 – Representação de um MOSFET canal N ilustrando as cargas aprisionadas no óxido após radiação. Do lado esquerdo o dispositivo operando em modo normal, e do lado direito após a radiação.



Fonte: Autor "adaptado de"Oldham e McLean, 2003, p.484

Os semicondutores são sensíveis a este tipo de armadilhamento de cargas e, em particular, os transistores MOS são fortemente afetados pelos danos por ionização (FACCIO et al., 2009). Alguns dos problemas causadas pelo TID em transistores MOS são: mudança na ten-

²Essa região é a compreendida entre o isolante da porta, geralmente óxido de silício, e a região de canal do dispositivo.

são de limiar³, devido às cargas estarem armadilhas próximo a interface, alterando a tensão necessária para ligar o dispositivo, e dependendo do valor da carga armadilhada, é possível que o dispositivo não possa mais ser desligado (OLDHAM; MCLEAN, 2003), diminuição da transcondutância e aumento da corrente de fuga (FACCIO et al., 2009).

2.2.1.2 Displacement Damage (DD) - Dano por deslocamento

Os danos por deslocamento ocorrem quando uma partícula incidente transfere energia suficiente para mover um átomo da sua posição normal na rede cristalina para outra, criando um "buraco"na grade. Para casos onde a energia transferida é muito alta, regiões de danos microscópicas são criadas e apresentam dimensões na ordem de $60\mu m$ (JOHNSTON, a., 2001). Energias transferidas para um átomo da estrutura cristalina podem gerar o mecanismo do DD. Para o caso onde uma pequena energia é absorvida, um par de vaga-intersticial é gerado, conhecido como par de Frenkel (Figura 10), onde a vaga gerada e o átomo estão próximos um do outro (SROUR; MARSHALL; MARSHALL, 2003).

Figura 10 – Diversos defeitos criados pelo deslocamento de átomos.



Fonte: Autor "adaptado de" Makowski, 2006, p.19

Este efeito é mais comum em transistores de junção bipolar (ARUTT et al., 2015) e optoeletrônicas (JOHNSTON, a., 2001), pois se tem perdas nos ganhos destes dispositivos.

³Tensão mínima necessária para a formação de um canal de condução no dispositivo.

2.2.1.3 SEE (Single Event Effects) - Efeitos por Eventos Isolados

O *Single Event Effects* (SEE) (Efeitos por Eventos Isolados) são efeitos de curta duração produzidas por uma única partícula ionizante que incide sobre os dispositivos semicondutores. A corrente é causada pelo rastro ionizado criado pela partícula carregado ao longo do seu caminho pelo semicondutor ou isolante. Se a carga gerada pela partícula for grande o suficiente, causará uma mudança no funcionamento normal do circuito, produzindo efeitos transientes que podem alterar o dispositivo (JOHNSTON, A., 2010). As fontes mais comuns do SEE são, raios galáticos cósmicos, partículas solares cósmicas e prótons aprisionados em cintos de radiação (fig.11)(JOHNSTON, A., 2010; STURESSON TEC-QEC, 2009).

Figura 11 – Simples representação dos cintos de radiação em volta da Terra.



Fonte: Autor "adaptado de" Allan Johnston, 2010, p.213

Uma partícula carregada ao passar por uma estrutura de um semicondutor gera um intenso rastro de pares elétron-lacuna. A densidade de cargas é muita alta, o que acaba causando uma distorção no campo elétrico ao longo e próximo do traço de ionização. A figura 12 ilustra o efeito causado por uma partícula ao incidir sobre uma junção N-P reversamente polarizada, a linha pontilhada representa a depleção da junção, que é alterada pela alta densidade de cargas geradas no processo.



Figura 12 - Geração de cargas devido a colisão de uma partícula ionizante em uma junção P-N

Fonte: Autor "adaptado de"Allan Johnston, 2010, p.334

Um segundo mecanismo pode gerar o SEE, este processo envolve colisões nucleares ao invés de ionização direta (Figura 13), onde um próton ou um nêutron atinge uma átomo na estrutura cristalina, causando o seu deslocamento. A movimentação deste átomo de recuo pode interagir com os elétrons do material alvo causando ionização pela sua trajetória.





Fonte: Fonte: Autor "adaptado de" Allan Johnston, 2010, p.335

Apenas uma porção da carga depositada por uma partícula ionizada é coletada, e existem diversas razões para isso. A primeira delas é, a recombinação de uma parte dos pares elétronlacuna geradas pela partícula ionizante antes da sua coleta pelo dispositivo. Segundo, para
dispositivos que apresentam uma geometria muito pequena (dimensões menores que $1\mu m$), o tamanho do nó de coleta dos pares elétron-lacuna é menor do que o diâmetro da traço de carga, que atingi um valor na ordem de 1μ m na condição de equilíbrio, um valor considerado grande para dispositivos com alta densidade de integração. As cargas geradas serão recombinadas mas não coletadas pelo dispositivo que sofreu a incidência da partícula ionizada, podendo outra junção ativa do circuito fazer esta coleta. Por último, a coleta de cargas é afetada pela dopagem do dispositivo, quanto maior for o nível de materiais dopantes no alvo menor será a coleta de cargas (JOHNSTON, A., 2010). O restante dos pares elétron-lacuna são colhidos através de um efeito denominado de funil (funnelling), a sua ocorrência se deve ao mecanismo gerado pela passagem da partícula ionizante, que ao passar pela estrutura do semicondutor gera uma quantidade de pares elétron-lacuna superior ao que são gerados pelos dopantes. Este plasma de elétron-lacuna se comporta como um condutor independente, que esta ligado desde o eletrodo onde a partícula insidio até o substrato do dispositivo, que é responsável pela coleta destas cargas através de deriva no campo elétrico. O resultado da coleta destas cargas é traduzida em um pulso de corrente momentâneo no dispositivo (fig.14) (MELOROSE; PERROY; CAREAS, 2015; NAKAJIMA et al., 2013; SCHWANK; SHANEYFELT; DODD, 2008; OLDHAM, 2003; STURESSON TEC-QEC, 2009).





Fonte: Autor "adaptado de"Oldham, 2003, p.24

Devido ao pequeno tempo envolvido no processo do SEE (na ordem de 1ns), a sua avaliação é feita usando o conceito de carga crítica. A carga crítica(Q_c) é definida como, a

miníma carga coletada por um circuito especifico necessária para alterar o seu estado. O mínimo valor de LET capaz de gerar uma Q_c responsável por um SEE é denominada de LET de limiar(LET_{TH}).

A seção de choque de um SEE é uma medida indireta da probabilidade de seu acontecimento. É calculada pela Equação 2:

$$\sigma = \frac{eventos}{\Phi} \tag{2}$$

Na fig.15, é ilustrado a seção de choque em função do LET, indicando o LET_{TH} onde a coleta de cargas não é o suficiente para gerar um SEE. Na parte superior é indicado o valor de , onde mesmo havendo um acréscimo do LET a probabilidade de ocorrências do SEE não aumenta (JOHNSTON, A., 2010; STURESSON TEC-QEC, 2009). O LET_{TH} e a são as chaves para mediar um SEE.

Figura 15 – Seção de choque do SEE em função do Linear Energy Transfer (LET).



Fonte: Autor "adaptado de"Sturesson Tec-Qec, 2009, p.11

SEE não destrutivo

 Single Event Upsets (SEU): O SEU é uma das classificações para o SEE. Este efeito afeta basicamente circuitos digitais, como células de memória e registradores. A geração de cargas durante o evento acaba mudando o estado de operação, de "0"para "1", por exemplo. Estas mudanças de estados de um *bit* é conhecido como *bit flip*. Caso o efeito se reflita em diversos bits no mesmo circuito, o efeito ganha a classificação de MBU (*Multiple Bit Upset (MBU*)) (NORMAND, s.d.; STURESSON TEC-QEC, 2009).

- Single Event Functional Interrupts (SEFI): O SEFI (Single Event Functional Interrupts (SEFI)), ocorre em dispositivos complexos, como os FPGAs, memórias avançadas, memórias flash e microprocessadores. A carga que é gerada pelo evento corrompe o sistema lógico do dispositivo fazendo o parar de funcionar corretamente. Frequentemente é desencadeado por um SEU nos registradores dos dispositivos que torna a funcionar corretamente após uma reinicialização do circuito (RIEMER et al., 2015; STURESSON TEC-QEC, 2009).
- Single Event Transients (SET): Basicamente, um íon que passa por um circuito, que causa um transiente de tensão em uma junção, é um SET. Os efeitos de um SET pode afetar circuitos subsequentes se não for bem filtrada. O SET pode gerar diversos SEU sendo estes, fortemente dependentes das condições de polarização. Este tipo de evento se torna cada vez mais comum devido a redução das geometrias dos dispositivos pois diminui o valor da carga crítica necessária para a sua ocorrência (OLDHAM, 2003).

SEE destrutivo

Single Event Latch-up (SEL): É um problema particular que ocorre em dispositivos CMOS com alta escala de integração(Large Scale Integration (LSI)). Pelo fato de termos os transistores NMOS e PMOS fabricados no mesmo wafer⁴ de silício, junções do tipo PNPN são formadas gerando tiristores⁵ parasitários. Se a tensão de entrada excede o da fonte de alimentação momentaneamente devido ao impacto de partículas de alta energia, o tiristor ficará ligado e o excesso de corrente continua a fluir podendo destruir o dispositivo devido ao sobreaquecimento. Este evento é fortemente dependente da temperatura. O limiar para o acontecimento de um *latchup* diminui em altas temperaturas. O efeito pode ser revertido removendo a alimentação do circuito (NAKAJIMA et al., 2013).

⁴O *wafer* é um disco fino, geralmente de silício, com orientação cristalográfica controlada, que é usado na fabricação de dispositivos eletrônicos.

⁵Tiristores são dispositivos semicondutores com quatro camadas de materiais tipo N e tipo P intercaladas de forma alternada, que atuam exclusivamente como uma chave biestável.

- Single Event Burnout (SEB): O SEB é tipicamente observado em transistores de potência. Uma partícula ionizada passa pelo dispositivo desligado, com uma alta tensão de bloqueio. A passagem do íon gera um FET de potência com capacidade de entrar no modo de segundo *breakdown*⁶. Se o efeito não for extinguido rapidamente, a alta corrente gerada faz com que o dispositivo sobreaqueça resultando em uma falha destrutiva (SEXTON; MEMBER, 2003).
- Single Event Gate Rupture (SEGR): O SEGR ocorre quando a interação de uma partícula ionizada com a óxido de porta, resulta na destruição da rigidez dielétrica do óxido usado nessa região, causando problemas quando se eleva o campo elétrico da porta, pois a corrente resultante no dispositivo é alta e capaz de danificar o componente (SEXTON; MEMBER, 2003).

2.3 DETECÇÃO DE PARTÍCULA

Os primeiros detectores de radiação usados no começo do século 20 eram baseados em emulsões fotográficas e telas fosforescentes. Esta limitação foi superada por Rutherford e Geiger (RUTHERFORD; GEIGERG, 1908), que desenvolveram um detector de radiação a gás que conseguia registrar eventos individuais nos experimentos, e aprimorada posteriormente por Geiger e Muller, gerando o tubo Geiger-Muller. A principal limitação que esse tipo de detector apresenta é medir energia depositada. O avanço nesta arte foi alcançado na década de 30, com a criação do tubo de Kubestsky (KUBETSKY, 1937) que posteriormente evoluiu para o fotomultiplicador (ZWORYKIN; MORTON; MALTER, 1936). Este detector, quando acoplado com um material cintilante, registra a energia que o evento deposita.

Na década de 60 o interesse por contadores de partícula baseados em cristais cresceu devido a dois grandes marcos que ocorreram na época. O primeiro deles foi o desenvolvimento de detectores de radiação baseados em germânio e silício, como consequência do rápido crescimento da industria de semicondutores (MILLER; GIDSON; DONOVAN, 1962). O segundo grande evento foi a síntese, com sucesso, de grandes cristais de Cádmio-Telúrio por Nobel (DE NOBEL, 1959), o que levou ao crescimento de um grande número de outros compostos semicondutores, principalmente na gama de compostos da coluna II-VI da tabela periódica.

⁶A região de *breakdown*, é uma região de operação onde o dispositivo recebe um alto valor de tensão fazendo a corrente que flui por ele crescer drasticamente.

Atualmente, o desenvolvimento de detectores são limitados, basicamente, pela qualidade dos materiais, pois eles apresentam defeitos, densidades irregulares de impurezas e desequilíbrios estequiométricos. Muitos esforços estão focados nas características que degradam a performance dos detectores, sendo os principais trabalhos conduzidos por (LUKE, 1995) entre outros. O campo de pesquisa, que ainda se encontra na sua infância, é a substituição do materiais padrões que são utilizados atualmente (Si e Ge) por semicondutores compostos (OWENS; PEACOCK, 2004), (MCGREGOR; HERMON, 1997), (OWENS, 2012).

Pensando em um modelo simplista de um detector de partículas, o primeiro ponto a ser considerado é a interação de uma partícula com o detector. Estas interações ocorrem rapidamente (tipicamente alguns nanosegundos em meio gasoso ou pico-segundos nos sólidos), sendo em muitos casos considerado que a deposição da energia da radiação no material é instantânea. O resultado desta interação, em uma ampla categoria de detectores, é a aparição de uma quantidade de carga elétrica dentro do volume ativo do detector. Assumindo que, no tempo t = 0 uma carga Q apareça no dispositivo, resultado de uma única partícula incidindo sobre o detector, teremos a coleta das cargas geradas no dispositivo pelos métodos discutidos anteriormente neste capítulo, resultando no aparecimento de um sinal elétrico. Os tempos de coleta variam muito com o meio onde as cargas são geradas e as mobilidades destas cargas. A figura 16 representa a resposta da coleta de cargas em função do tempo de coleta de cargas t_c :

Figura 16 – Corrente em função do tempo de coleta de cargas (t_c) .



Fonte: Autor "adaptado de"Glenn F. Knoll, 2014, p.106

Para se obter o valor da carga Q gerada por essa interação, é necessário realizar a integração da corrente no intervalo de t_c , conforme equação 3:

$$\int_0^{t_c} i(t)dt = Q \tag{3}$$

Os detectores de partícula geralmente são enquadrados em um dos seguintes modos: Modo de pulso, modo de corrento e modo e *Mean Square Voltage* (MSV).

MODO DE CORRENTE: Operando neste modo, o detector apresenta um valor de corrente média constante. Levando em consideração que o medidor tenha um tempo fixo de resposta T, o valor registrado pelo medidor por conta de uma sequência de eventos será dependente do tempo, calculado pela equação 4:

$$I(t) = \frac{1}{T} \int_{t-T}^{T} i(t') dt'$$
(4)

Este modo é largamente utilizado quando existe uma taxa de eventos muito alta, ou quando a aplicação é do tipo dosímetro. No modo de corrente, o valor mínimo de corrente detectável representa a média de interações com partículas ionizantes que ocorrem no detector. Neste modo basicamente, o valor de resposte captado pelo detector é proporcional a quantidade de cargas que dada partícula gera no volume sensível do dispositivo.

- Mean Square Voltage (MSV): O modo MSV, e mais utilizado quando se realiza medições em ambientes onde existem diversas fontes de radiação, quando uma carga gerada por uma partícula ionizante é muito diferente da outra. Neste caso, se o modo de corrente fosse selecionado, o valor de resposta do detector refletiria linearmente a contribuição de geração de cada carga para cada partícula, porém no modo MSV, o resultado é proporcional ao quadrado da derivada do sinal de carga gerada por cada evento, o que acaba gerando um "peso"para cada tipo de radiação emitida, facilitando assim a detecção e classificação do íon usado no momento da radiação. O modo de operação MSV, encontra grande aplicação na instrumentação de reatores (GLENN F. KNOLL, 2014).
- MODO DE PULSO: Na operação em modo de pulso, o detector é projetado para registrar cada irradiação individualmente. A operação de modo de pulso é o modo mais comumente utilizado para a maioria de detectores de radiação, devido as suas vantagens sobre o modo de corrente. A primeira vantagem está relacionada a sua sensibilidade que é muitos vezes maior em comparação com o modo MSV ou modo de corrente, devido a sua capacidade de identificar cada emissão de partícula

individualmente. A segunda vantagem e mais importante é que cada amplitude de pulso carrega informações sobre a partícula. Tanto no modo de corrente e MSV as informações sobre o pulso são perdidas. Por conta destas características o modo de pulso é largamente utilizado em instrumentação nuclear, circuitos de pulsos e em técnicas de processamento de pulsos.

2.4 ALGORITMOS DE APRENDIZADO

Nesta seção serão abordadas as técnicas utilizadas para o pré-processamento, processamento e classificação dos dados obtidos em campo, nos experimentos relacionados a detecção dos efeitos de radiação ionizante em disposivitivos semicondutores.

2.4.1 O que é aprendizado de máquina?

Um algoritmo de aprendizado de máquina é um algoritmo que, sem interferências externas, é capaz de aprender através das observações dos dados. Mas qual é a definição de aprendizado? Em seu livro, Mitchell (1997) expõe uma definição breve e simples: "O aprendizado de um programa de computador é realizado através da experiência E em relação a uma classe de tarefas T e medida de performance P, caso a performance nas tarefas T, medidas por P, são melhorados através da experiência E"

2.4.1.1 Tarefas T de algoritmos de aprendizado

O primeiro ponto a ser levado em consideração em algoritmos de aprendizado é, que tipo de tarefa a máquina se propõe a fazer. Em face a um problema existente, algumas abordagens para a sua solução podem ser utilizadas. A primeira delas é a utilização do conhecimento de um especialista para a modelagem de uma solução baseada em sua experiência, geralmente este tipo de tarefa já é bem mapeada e a maioria das variações são conhecidas, porém caso haja uma nova variável ou procedimento, a solução inicial deverá passar por uma manutenção, adicionandose o novo comportamento não mapeado anteriormente. No caso anterior, podemos dizer que existe uma solução ótima, que cobre todo o espectro de possibilidades, que seria factível gerar uma solução de forma programática, contudo, um conjunto muito grande de caminhos a serem explorados, ou uma explosão combinatorial de possíveis situações, acaba tornando essa abor-

dagem insustentável ou até mesmo inviável, nesse tipo de situação onde uma solução ótima não existe ou é desconhecida, a abordagem via aprendizado de máquina ganha seu espaço, trazendo abordagens baseada em dados passados e soluções sub-ótimas.

Os tipos de tarefas para algoritmos de aprendizado de máquina podem ser agrupadas em duas grandes vertentes: supervisionadas, aquelas que contém uma reposta esperada em seu conjunto de dados tendo um caráter preditivo, e tarefas não-supervisionadas, aquelas que não apresentam variável resposta e tem natureza descritiva. Dentro das tarefas supervisionadas podemos destacar as tarefas de classificação e regressão, que tem por objetivo traçar superfícies de separação ou identificar linhas de tendência respectivamente. O outro grande grupo de tarefas, as não-supervisionadas, são aquelas voltadas a identificar estruturas nos dados de forma a descrevê-los, são principalmente, mas não se limitam, a tarefas de clusterização de dados, que visam encontrar grupos de dados semelhantes e agrupá-los de tal forma que é possível encontrar as regras e características por trás de cada grupo.

As tarefas citadas anteriormente são algumas das aplicações em aprendizado de máquina e nem de longe se limitam a estes casos, a intenção é apenas mostrar alguns exemplos da capacidade destes algoritmos e suas vastas aplicações.

2.4.1.2 Medidas de performance P de algoritmos de aprendizado

A medida de performance de modelos de aprendizado de máquina estão intimamente relacionados com a tarefa a ser executada, seja classificação ou regressão, pois cada uma tem sua natureza e particularidades. Modelos de classificação apresentam diversas métricas de performance, das quais precisão e acurácia são as mais conhecidas, e a sua escolha é feita de acordo com o objetivo final. Por exemplo, se o objetivo é otimizar ao máximo os casos de acerto, mesmo que alguns casos certos fiquem de fora, a melhor métrica para ser usada é a precisão. Isto fica mais claro no seguinte caso de aplicação: classificar se um cliente terá ou não problemas de crédito, o acionamento de toda uma cadeia de cobranças e provisionamentos de um banco será efetuado unicamente quando a entidade estiver confiante. Por outro lado, uma métrica como a acurácia privilegia a seleção de todos os *possíveis* casos positivos, mesmo que entre eles tenham alguns casos negativos, uma aplicação desta medida de performance em tarefas de classificação, é em filtros de conteúdo para vídeos indicados para crianças, é muito interessante que nestes casos qualquer suspeita de vídeos inapropriados seja filtrado, mesmo que algum deles seja um vídeo apto.

Como podemos ver, as medidas de performance devem ser muito bem alinhadas com o a tarefa a ser desempenhada, pois não se avalia um peixe pela sua capacidade de escalar árvores.

2.4.1.3 Experiência E de algoritmos de aprendizado

Por fim, temos a experiência que o algoritmo irá experimentar durante o processo de aprendizado, seja supervisionado ou não-supervisionado. Na experiência supervisionada, o algoritmo de aprendizado conta com um conjunto de dados com as características do fenômeno a ser aprendido e uma variável resposta para cada instância desse conjunto de dados. Já nos casos de experiências não-supervisionadas o algoritmo de aprendizado conta com um conjunto de dados sem uma variável resposta definida, o que torna a tarefa de aprendizada focada em extrair padrões dos dados e aprender suas distribuições, e dependendo da tarefa, utilizar esse aprendizado para realizar seu objetivo.

2.4.2 Capacidade, overfitting e underfitting

Os algoritmos ou modelos de aprendizado de máquina tem como principal objetivo ser bons generalizadores, ou seja, dado um conjunto de dados nunca antes vistos, os modelos de aprendizado de máquina devem classificá-los ou regredi-los, dependendo de sua tarefa, da melhor forma possível, o que nos remete a diferença de um problema de otimização, que visa minimizar o erro apenas em um conjunto de dados conhecido. A esse conjunto de dados nunca vistos antes pelo modelo se dá o nome de *dados de teste*, e quanto menor for o erro nesse conjunto de dados melhor é a capacidade de generalização do algoritmo. Porém existe um compromisso entre o quão acurado um modelo é na tarefa de treinamento, com os dados de treino, e na fase de validação com os dados de teste. Quando a diferença entre os erros obtidos durante a etapa de treinamento e de validação são muito grandes se da o nome de *overfitting*. Quando o modelo, obtém uma taxa de erro muito grande durante a fase de treinamento se dá o nome de *underfitting*.

De maneira geral, a capacidade de um modelo é a habilidade de se ajustar, ou aprender, uma grande variedade de funções. Modelos com pouca capacidade apresentam um fraco ajuste aos dados de treinamento, pois possuem poucas funções capazes de desdobrar as representações e limitando a flexibilidade no aprendizado (*underfitting*). Em contrapartida, modelos com muita capacidade para capturar as não linearidades dos dados acabam se ajustando muito próximas ao determinado conjunto de dados de treinamento, tendo uma representação fiel unicamente destes dados, o que pode gerar uma fraca capacidade de generalização, com pouca capacidade de generalizar em face do conjunto de dados de teste (*overfitting*).

A forma que se encontra de controlar a capacidade de um modelo de aprendizado de máquina é através da escolha do espaço de hipótese do modelo, o que restringirá ou expandirá a capacidade do modelo de aprender as estruturas e distribuições por trás de um conjunto de dados. Para exemplificar o espaço de hipótese, iremos usar uma regressão linear simples, dada pela eq. 5, este modelo tem capacidade de se ajustar a dados onde a variável y tem uma relação linear com seu preditor X (fig. 17 esquerda), neste cenário temos uma capacidade ajustada ao problema pois o espaço de hipóteses do modelo contemplam todas funções lineares. Em contra partida, se temos o cenário representado em fig.17 a direita, a capacidade de um regressor linear não atende a necessidade da tarefa, nesse caso o espaço de hipóteses esta desajustado, e é necessária a revisão do espaço de hipótese expandindo o polinômio para um grau quadrático.

$$y = wx + b \tag{5}$$

Figura 17 – A esquerda: Dados lineares aproximados por uma função linear. A direita: Uma função linear aproximando dados quadráticos.



2.4.3 Hiperparâmetros e dados de validação

A maioria dos algoritmos de aprendizado de máquina contam com hiperparâmetros que permitem controlar o seu comportamento. Estes parâmetros, na maioria dos casos, não são otimizados pelo algoritmo em si, necessitando a utilização de técnicas externas para realizar a busca dos valores ótimos de cada parâmetro em seu espaço amostral. Durante o processo de busca dos melhores parâmetros de um modelo, é necessário certificar-se que não contenha o viés dos dados de treinamento, pois acabamos caindo no problema de *overfiting* como anteriormente dito, para isso utiliza-se a boa prática de dividir os dados de treinamento em duas partes, a primeira dela é maior (geralmente algo em torno de 80% a 90%) e destinada ao treinamento do modelo, sendo o restante chamados de dados de validação, e a sua importância durante a etapa de treinamento é muito grande pelo fato de se tratar de dados que o modelo nunca vê. Os ajustes dos parâmetros são baseados nos resultados obtidos nos dados de validação, guiando o processo de treinamento sem enviesar essa seleção e realizar um sobre ajuste do modelo nos dados de treinamento.

2.4.4 Validação cruzada

A utilização de dados de validação é uma boa prática para evitar problemas de *overfitting* durante a etapa de treinamento do modelo, porém existem outros problemas no processo de treinamento ligadas a natureza dos dados escolhidos. Um dos problemas que podem ocorrer, é que, quando se trata de um conjunto de dados pequeno, a forma como este é dividido pode interferir no resultado final do modelo, pois pode-se selecionar uma porção de dados para o treinamento que contenha uma distribuição particular e que difere daquela vista nos dados de teste, sendo assim, o modelo perderá sua capacidade de generalizar e terá uma taxa de erro elevada nesse cenário.

Para evitar o problema de viés na seleção das partições dos dados realiza-se a validação cruzada, ou (*k-fold cross-validation*), esquematizada na fig.18. Esta técnica consiste em dividir os dados em k partições e realizar diversas rodadas utilizando-se diferentes configurações de partições para realizar o treino e teste do modelo. Por exemplo, dado um conjunto de dados dividi-se em 5 partições de tamanhos iguais não sobrepostos, a primeira rodada de treinamento consistirá em utilizar as partições de 1 a 4 para treino e a partição 5 para teste, a seguinte rodada de treinamento sera efetuada nas partições 1-3 e 5 e o teste sera feito na partição 4, e assim por diante, permutando a configuração de seleção destas partições até que todos os conjuntos tenham sido utilizados tanto na fase de treino quanto na de teste. O erro de teste será a média dos erros obtidos em cada rodada de seleção de partições.





Fonte: Autor

2.4.5 Algoritmos supervisionados

Como mencionado anteriormente, algoritmos de aprendizado de máquina supervisionados podem ser definidos, de forma geral, como algoritmos que, dado um conjunto de variáveis independentes X de entrada, aprendem uma função que mapeia a variável dependente y, f(X) = y. Na grande maioria dos casos, a função f(x) é uma aproximação estocástica, tornando a tarefa de aprendizado supervisionado numa aproximação de uma distribuição de probabilidade condicional p(y|X). De forma geral, é possível avaliar o desempenho de algoritmo supervisionado observando-se o resultado obtido para dados nunca vistos antes, sendo um bom modelo aquele que aproxima da melhor forma os resultados esperados para esses dados inéditos. Para valores de y discretos, separados em classes, atribuímos o nome de classificação. Para valores de y contínuos, aproximamos um algoritmo de regressão do valor médio para cada par (X,y) do nosso conjunto de dados, pois a probabilidade de se obter o valor exato em regressões é quase 0.

2.4.5.1 Árvores de decisão

Para entender melhor os algoritmos supervisionados, será descrita de maneira intuitiva um algoritmo que se baseia em regras simples que quebram as variáveis independentes de forma que possam ser tomadas as decisões para se realizar a devida classificação ou regressão. Para



Figura 19 – Gráfico de pares mostrando a dispersão de cada variável do conjunto de dados Iris para a espécie Setosa vs as demais.

Fonte:Autor

isto, vamos utilizar um conjunto de dados amplamente conhecido chamado Flor Iris introduzido por Fisher (1936) no seu trabalho intitulado *The use of multiple measurements in taxonomic problems*(em português, O uso de medições múltiplas em problemas taxonômicos), esse conjunto de dados consiste em 50 amostras de cada uma das três espécies de iris (iris setosa, iris virgínica e iris versicolor), contendo 4 parâmetros de cada amostra, comprimento e largura das pétalas e sépalas, em centímetros. Na fig.19, temos a dispersão de cada amostra plotados em gráficos, e nesse cenário, estão dividas em duas categorias, distinguindo se é ou não da espécie iris setosa, tornando a tarefa de classificação binária.

Inspecionando a fig.19 pode se ver que criando uma regra simples para os parâmetros relacionados as pétalas conseguimos classificar corretamente todos os casos. Isto é, considerando Largura pétala $\leq 0.75cm$ ou Comprimento pétala $\leq 2.5cm$, conseguimos distinguir a classe *is_setosa*. O algoritmo de árvore de decisão performa a mesma tarefa, cada nó de uma árvore de decisão realiza um teste simples que se desdobra entre outras decisões até chegarem as folhas, que representam os valores de saída do algoritmo.

O algoritmo de aprendizado de árvore de decisão tem uma abordagem gulosa⁷ que busca minimizar o tamanho final da árvore, tornando sua interpretação mais simples. A ideia é selecionar aquela variável que consegue discriminar da melhor forma possível a variável resposta.

É necessário definir uma métrica para a seleção dessas variáveis que melhor discriminam a variável alvo, para isso utiliza-se a noção de ganho de informação, conhecida como entropia, a unidade fundamental na teoria de informação (SHANNON, 1948).

A entropia pode ser calculada utilizando a eq.6 e é definida como a medida da incerteza de uma variável randômica.

$$H(V) = \sum_{k} P(v_k) \log_2 \frac{1}{P(v_k)} = -\sum_{k} P(v_k) \log_2 P(v_k)$$
(6)

Quando temos aquisição de informação o valor de entropia diminui. Qualquer evento cujo resultado sempre é o mesmo, um dado viciado por exemplo, não apresenta incerteza e o seu valor de entropia é 0, desta forma não há ganho de informação observando este valor. Uma moeda honesta, que apresenta probabilidades iguais para cara e coroa, terá "1 bit"de entropia. Por outro lado, caso tenhamos uma moeda desonesta onde 99% dos lances dão cara e apenas 1% coroa, se sempre apostarmos em cara, erraremos apenas em 1% dos casos. De forma intuitiva, podemos inferir que essa moeda é menos incerta que a moeda honesta e isto sera demonstrado a seguir. Usando a eq.6 nos dois cenários, temos:

$$H(honesta) = -(0.5\log_2 0.5 + 0.5\log_2 0.5) = 1 \text{ bit}$$

Aplicando o caso a uma moeda honesta, vemos que a entropia é igual a 1 *bit*, que é necessário para representar os estados possíveis.

 $H(honesta) = -(0.99 \log_2 0.99 + 0.01 \log_2 0.01) \approx 0.08 \ bit$

⁷Em ciência da computação, se refere a estratégia de busca da solução ótima de um problema.

Definindo a entropia para variáveis binárias randômicas temos a eq.7:

$$B(q) = -(q \log_2 q + (1-q) \log_2 (1-q)) \tag{7}$$

Voltando ao caso do conjunto de dados Iris, temos 50 casos positivos de setosa, logo p = 50, e 100 casos negativos n = 100, para calcular a entropia temos eq.8:

$$H(objetivo) = B(\frac{p}{p+n})$$
(8)

Sendo assim, temos que $B(\frac{50}{50+100}) \approx 0.92 \, bit$. Observando individualmente cada variável, é possível medir exatamente o quanto de entropia do total ela representa.

Uma variável V com valores distintos d, consegue dividir o conjunto de dados E em subconjuntos E_1, \ldots, E_d , e cada subconjunto E_k , apresenta um número p_k de amostras positivas e um número n_k de amostras negativas, sendo assim, necessitamos de $B(p_k/(p_k + n_k))$ bits de informação adicionais para responder a questão. Uma amostra randomicamente selecionada, do conjunto de dados apresenta o k-ésimo valor para o atributo com a probabilidade p_k + n_k/p + n_k , desta forma o valor restante de entropia pode ser calculado pela eq.9:

$$Resto(V) = \sum_{k=1}^{d} \frac{p_k + n_k}{p + n} B(\frac{p_k}{p_k + n_k})$$
(9)

Então de forma geral, o ganho de informação que cada atributo presente em um conjunto de dados é dado por:

$$Ganho(V) = B(\frac{p}{p+n}) - Resto(A)$$
⁽¹⁰⁾

Usando eq.10, podemos selecionar de forma eficiente, quais variáveis independentes (X) desempenham de forma melhor na quebra da variável dependente (y), de acordo com o ganho que cada variável apresenta. Uma analogia que pode ser feita em relação a entropia, é o conceito de "pureza"em cada nó, ou seja, a medição da homogeneidade dos itens selecionados em cada nó, quanto mais homogêneo melhor. Por fim na fig.20 temos uma árvore gerada iniciando com uma variável com baixa entropia, o que acaba gerando uma árvore com muitas quebras até conseguir realizar a correta classificação.





Fonte: Autor

Por outro lado na fig.21 a árvore é iniciada por uma variável com maior entropia, resultando em uma árvore mais rasa e que discrimina de forma correta a classe *is_setosa*, tornando a interpretação dos resultados simples.





Fonte: Autor

2.4.6 Algoritmos não-supervisionados

Outra grande classe de algoritmos de aprendizado caem sobre a tarefas não-supervisionadas. Essas contam apenas com um conjunto de variáveis independentes X, e não contam com uma variável dependente y, sendo assim, não temos um objetivo bem definido como no casos de algoritmos supervisionados.

Algoritmos não supervisionados, tem como característica em comum a busca por uma estrutura dos dados, seja em uma tarefa de clusterização com o *K-means*, ou reduzir a dimensão dos dados, como o *PCA*, tarefas essas muito utilizadas em etapas de exploração de dados.

2.4.7 Principal Component Analysis (PCA)

Em paralelo com o que foi feito na seção de algoritmos supervisionados, será introduzida uma técnica amplamente conhecida no cenário de algoritmos não supervisionados o PCA. Em diversas áreas do conhecimento, os dados estão crescendo de forma exponencial, sendo assim, a necessidade da utilização de técnicas para interpretar estes dados é altamente requerida.

Em um cenário onde é necessário realizar uma análise exploratória dos dados, tomandose de duas em duas variáveis, o número de combinações cresce rapidamente, para um conjunto de dados com 10 variáveis, por exemplo, teríamos 45 pares de variáveis, o que tornaria a analise trabalhosa e impraticável. Desta forma, é necessário diminuir drasticamente a dimensão deste conjunto de dados, para que seja possível extrair informação relevante, porém mantendo um compromisso com a manutenção da maior parte da informação presente no conjunto de dados para que não se perca valor. Uma das técnicas mais amplamente utilizadas é a Principal Component Analysis (PCA). Esta técnica é aplicada em diversas áreas do conhecimento, como por exemplo na análise de sistemas lineares e suas pertubações (MOORE, 1981), em análises geológicas para estimar a concentração de metais pesados (LOSKA; WIECHUŁA, 2003) e até mesmo na classificação facial (KIM; JUNG; KIM, 2002). A ideia por trás do algoritmo é simples, cada n observação pertence a um espaço p-dimensional, porém nem todas essas dimensões são igualmente interessantes, portanto, reduzir a dimensionalidade dos dados preservando a maior variabilidade presente neste conjunto é a chave por trás deste método. O número de componentes gerados pelo procedimento é menor ou igual ao número de variáveis presentes nos dados originais, sendo a primeira componente responsável por representar a maior parte da variância presente nos dados, por conta disso esta componente é conhecida como "componente principal", as próximas componentes seguem a ordem em máximas variâncias, levando-se em consideração, a ortogonalidade em relação a componente anterior. A primeira componente principal de um conjunto de variáveis X_1, X_2, \ldots, X_p é a combinação linear normalizada de suas variáveis:

$$Z_1 = \phi_{11}X_1 + \phi_{21}X_2 + \dots + \phi_{p1}X_p \tag{11}$$

que contém a maior variância presente neste conjunto de dados. O termo normalizado se refere a $\sum_{j=1}^{p} \phi_{j1}^2 = 1$, onde $\phi_{11}, \ldots, \phi_p 1$ são os pesos da primeira componente principal. Estes pesos são condicionados de tal forma que a soma dos quadrados destes valores é igual a um, evitando possíveis arbitrariedades na variância. Assim, dado um conjunto de dados $n \times p X$, computamos a primeira componente principal da seguinte maneira:

$$z_{i1} = \phi_{11}x_{i1} + \phi_{21}x_{i2} + \dots + \phi_{p1}x_{ip}$$
(12)

que apresenta a maior variância atendendo a restrição $\sum_{j=1}^{p} \phi_{j1}^2 = 1$. No fundo temos que a obtenção dos pesos para a primeira componente principal resolve o problema de otimização:

$$maximize(\phi_{11}, \dots, \phi_p 1) \left\{ \frac{1}{n} \sum_{i=1}^n \left(\sum_{j=1}^p \phi_{j1} x_{ij} \right)^2 \right\} para \sum_{j=1}^p \phi_{j1}^2 = 1.$$
(13)

Uma interpretação geométrica para a primeira componente principal é que o vetor de pesos ϕ_1 com elementos $\phi_{11}, \phi_{21}, \ldots, \phi_{p1}$ define a direção no espaço das variáveis onde esta presente a maior variação dos dados. Ao projetarmos os n dados x_1, \ldots, x_n nesta direção, estes dados projetados são a componente principal z_{11}, \ldots, z_{n1} por si só. A figura 22 ilustra a projeção de dados em três dimensões para um plano em duas dimensões e a figura 23 mostra o resultado final da projeção.



Figura 22 – Um conjunto de dados em 3D projetado em um sub-espaço 2D.

Fonte: Autor "adaptado de"Géron, 2017, p.318

Figura 23 – Novo conjunto de dados projetado em 2D.



Fonte: Autor "adaptado de"Géron, 2017, p.319

Por fim, a figura 24 ilustra duas componentes, sendo a componente em vermelho, a principal, pois através dela a maior parte da variância daquele dados é explicada. De forma análoga tem-se a segunda componente (seta amarela) que é ortogonal a primeira, e representa a segunda porção com maior variância para esse caso.





Gaussiana bivariada centrada em (0,0) com duas componentes de PCA.

Fonte:Autor

2.4.8 Motivação aprendizado profundo

Quando se trata de modelos de aprendizado profundo estamos entrando em um novo ramo do aprendizado de máquina, que supera os modelos clássicos como árvores de decisão e SVM, mas este mundo de possibilidades vem carregado de novos desafios a serem superados, como o poder computacional exigido ou a interpretação dos resultados obtidos. Como regra geral, quando se trata de um problema de aprendizado de máquina, adota-se a abordagem mais simplista do problema, supondo que o espaço de busca do problema pode ser mapeado com modelos lineares, e com o aprofundamento da complexidade do problema outras abordagens mais sofisticadas podem ser requeridas até chegarmos nas abordagem do aprendizado profundo. Existem motivações que levam a adoção de uma abordagem de aprendizado profundo, como a não necessidade de criação de caraterísticas do problema por um especialista, a capacidade de melhoria dos resultados conforme o volume de dados cresce, capacidade de transferência de conhecimento, capacidade de generalização e a capacidade de lidar com a maldição da alta dimensionalidade (HINNEBURG; KEIM, 1999), este tópico será explorado adiante.

Levando em consideração os pontos anteriormente citados, o presente trabalho, propõe uma abordagem para a classificação de sinais medidos de SEE, dada a sua natureza temporal, variabilidade e volume de informação, uma abordagem de aprendizado profundo casa perfeitamente com a necessidade da tarefa de classificação dos sinais medidos. Nas próximas sessões serão descritas as estruturas básicas utilizadas na construção do algoritmo proposto.

2.4.8.1 Maldição da alta dimensionalidade

O problema surge por conta do comportamento das distâncias em espaços com muitas dimensões. É possível observar esse comportamento comparando-se um espaço com dimensão p = 2 e p = 10000. No primeiro caso, se adotarmos um quadrado de lado um, e aleatoriamente pegar-se um ponto, este terá por volta de 0.4% de chance de estar localizado a menos de 0.001 da borda deste quadrado, por outro lado, se o mesmo experimento for realizado para um hipercubo 10000-dimensional de lado um, a mesma probabilidade de um ponto adquirido aleatoriamente é de 99,999999%, pois em um espaço com alta dimensão muitos pontos estão próximos de uma o borda. O problema é maior quando se pega dois pontos aleatórios e mede-se a distância entre eles. Em um espaço com dimensão p = 2 a distância, em média, é de 0,52 em um quadrado de lado um, em um espaço com dimensão p = 3 essa mesma distância vai para 0,66, porém em um hipercubo 1000000-dimensional a distância entre dois pontos adquiridos aleatoriamente é de 408,25, o que é uma distância contra intuitiva, pois se considerar que cada lado deste hipercubo mede uma unidade. Esta característica implica que, conjunto de dados com alta dimensão correm o risco de serem muito esparsos, e em algoritmos que se baseiam em medidas de distância entre cada instância, a predição e a regressão para novos valores que são apresentados para o algoritmo é uma tarefa difícil e menos confiável se comparados para um conjunto de dados com dimensão menor (DONOHO, 2000; GÉRON, 2017).

Outro ponto agravante é que, conforme o número de parâmetros aumenta, o número de combinações possíveis para esse conjunto de dados cresce de forma exponencial, o que torna o número de combinações muitas vezes maior que o número de exemplos presentes no conjunto

de dados, desta forma, modelos clássicos de aprendizado de máquina falham por não terem formas de solucionar os problemas em casos nunca vistos de um conjunto de dados.

3 APRENDIZADO PROFUNDO

A humanidade sempre se inspirou na natureza para desenvolver suas criações, seja para medir a capacidade de um motor com a unidade cavalos (cv) ou criar células de energia solar mais eficientes, inspirado no movimento do girassol, mas existe uma coisa que ainda buscamos que é emular um ser humano, ou pelo menos a sua forma de pensar, aprender e agir. Nas próximas sessões serão explorados as técnicas que estão na vanguarda do aprendizado de máquina, as redes neurais profundas, que são capazes de performar diversas tarefas, desde classificação de imagens a geração de dados sintéticos (imagens, vídeo, som) de forma realística, muita vezes imperceptível para olhares mais desatentos.

3.1 REDES NEURAIS

Em seu trabalho clássico McCulloch e Pitts (1943), apresentaram o conceito de "neurônio artificial", que séria a unidade básica de uma rede neural, baseado nas recentes descobertas da neurociência da época, sobre as características "tudo ou nada"de um neurônio, que somente disparam um pulso quando os valores em sua entrada alcançam um certo limiar, sendo assim, os autores usaram essa características e equacionaram os primeiros modelos matemáticos de um neurônio artificial. De modo simplista, o neurônio é ativado quando uma combinação linear dos valores de suas entradas x_i e seus respectivos pesos $w_{i,j}$ ultrapassam um valor limiar, definido pela função de ativação g(z) resultando no valor de ativação a_j , sendo assim temos eq.14:

$$z_j = \sum_{i=0}^n w_{i,j} x_i$$

$$a_j = g(z_j)$$
(14)

No esquema representado na fig.25, é possível observar a estrutura de um neurônio artificial. O conjunto destes elementos formam as redes neurais.



Figura 25 - Representação matemática de um neurônio artificial

Fonte: Autor

De modo geral, as redes neurais, são aproximadores globais de uma função f(x) utilizando a combinação de diversas funções, $f^{(1)}$, $f^{(2)}$, $f^{(3)}$, $f^{(n)}$, formando $f(x) = f^{(n)}(f^{(3)}(f^{(2)}(f^{(1)}(x))))$. Essa estrutura em cadeia é a mais conhecida e comumente utilizada no cenário de redes neurais, sendo nesse caso, a função $f^{(1)}$ nossa primeira camada (camada de entrada), a nomeação continua até $f^{(n)}$, que representa a última camada, nomeada de camada de saída. O conjunto dessas camadas definem a profundidade do modelo ou rede neural, sendo o nome aprendizado profundo, relacionado a esse conceito. As camadas que se localizam entre a primeira e ultima camada de uma rede neural recebem o nome de camadas ocultas, isso se dá por conta da característica de aprendizado de uma rede neural, pois o dado de entrada somente tem informação para a primeira camada, as demais camadas são alimentadas com as informações processadas pelas camadas passadas, e cabe ao modelo de aprendizado, utilizar da melhor forma possível as demais camadas para desempenhar a função de aproximar a função desejada, esse processo é conhecido como propagação *feedfoward*. Por fim, o nome rede neural vem da sua representação gráfica acíclica conectando as unidades de cada camada através das sinapses em analogia ao neurônio biológico. Sua representação pode ser visto na fig.26.





Fonte: Autor

A escolha do número de camadas ocultas que uma rede neural e sua largura, que representa o número de unidades que cada camada terá, é uma etapa muito sensível e importante do processo de criação de uma rede neural, determinando a sua eficiência na tarefa de aprendizado, contudo não existem regras claras e consenso em relação a este aspecto, o que torna essa etapa do projeto um processo de tentativa e erro em busca dos parâmetros adequados para cada problema (STATHAKIS, 2009).

3.1.1 Funções de ativação

Outro parâmetro concernente a redes neurais é a função de ativação que estará presente em cada unidade da rede neural. Dentre as diversas opções serão descritas as mais proeminentes e largamente utilizadas, são elas: *Rectified Linear Units* (ReLU), *Leaky Rectified Linear Units* (Leaky ReLU), função Logística ou sigmoid e tangente hiperbólica.

A primeira delas a ReLU é fácil de otimizar pois se assemelha a unidades com a função linear, e se mostrou mais eficiente em treinamentos de redes neurais profundas (GLOROT; BORDES; BENGIO, 2011). A equação que descreve o comportamento da ReLU é demonstrada em 15.

$$g(z) = max(0,z) \tag{15}$$

A variação da ReLU é a Leaky ReLU, que foi introduzida no contexto em que o problema a ser solucionado continham matrizes de dados muito esparsas e com predominância de 0 nas ativações, sua adoção visa reduzir os efeitos dessas ativações (MAAS; HANNUN; NG, 2013). Sua equação é descrita em 16.

$$g(z) = \begin{cases} z & \text{se } z > 0, \\ 0.01z & \text{caso contrário.} \end{cases}$$
(16)

A função de ativação logística ou sigmoid é recomendada para as camadas de saída em situações onde é necessário retornar um valor de probabilidade, pois a função só existe entre os valores 0 e 1. É desencorajada para ser usada em camadas ocultas pois a função rapidamente satura para valores altos ou baixos de z, tornando a tarefa de otimização nesses cenários dificultosa. Sua equação é descrita em 17.

$$g(z) = \frac{1}{1 + e^{-z}} \tag{17}$$

Por fim, temos as tangentes hiperbólicas, que apresentam o mesmo formato da função sigmoid em formato de "s", porém esta compreendida entre -1 e 1. Sua vantagem esta pelo fato de valores negativos de z são fortemente mapeados para um valor negativo e vice versa. Essa função de ativação também é aplicada em situações de classificação entre duas classe. Sua função é descrita em 18.

$$g(z) = \tanh(z) \tag{18}$$

3.1.2 Algoritmo *backpropagation* e aprendizado da rede neural

A arquitetura Perceptron introduzido nas décadas de 1950 e 1960 por Rosenblatt (1960), era considerada uma revolução pois poderia resolver diversos problemas computacionais com apenas uma camada de profundidade, contudo em 1969, um problema que, a princípio, tem uma baixa complexidade, trouxe um grande desafio para a área das redes neurais, o problema da função *XOR* (MINSKY; PAPERT, 1969). A função XOR, é uma função binária, que retorna 1 caso algum dos valores de entrada sejam exclusivamente 1 e 0 caso contrário. Por ser um problema não linearmente separável, a resolução não era possível para a arquitetura do perceptron, que conta com apenas com uma camada de unidades, podendo realizar a separação dos pontos com uma única linha, como visto na fig.27.

Figura 27 – Mapeamento da saída da função XOR para duas variáveis de entrada X_1 e X_2



Fonte: Autor

A solução para o problema da função *XOR* seria adicionar novas camadas à arquitetura original, gerando a arquitetura conhecida como *Multilayer Perceptron* (MLP), expandindo assim a capacidade de representação da rede neural. O grande problema desta abordagem é encontrar um conjunto de pesos $(w_{i,j})$ suficientemente bons para que a rede neural realize a tarefa de forma satisfatória. Neste cenário é que surge o algoritmo de *backpropagation*, e a primeira demonstração de sua eficiência no problema XOR foi realizado em 1985 (RUMELHART; HINTON; WILLIAMS, 1985).

O algoritmo de *backpropagation* permitiu a adoção das MLP, por permitir a atualização dos pesos de cada conexão em cada camada. O método conhecido como regra Delta Widrow e Hoff (1960), utilizada para atualização dos pesos e treinamento de redes neurais de uma única camada que já era um avanço do método de aprendizado do perceptron, não se aplica em redes neurais com múltiplas camadas, mas pode ser considerado um caso especifico do *backpropaga-tion* já que em sua primeira divulgação, era conhecido como regra Delta generalizada.

Observando a estrutura de uma rede neural, observarmos que existe algumas formas de controlar o resultado esperado na saída, que seria alterando os valores de a, w ou b dada pela

eq.14, sendo o candidato mais apropriado o $w_{i,j}$ que representa o peso que cada sinal tem na representação de cada dado de entrada.

O objetivo do algoritmo de *backpropagation* portanto é, otimizar a procura dos conjuntos de W para que se obtenha o menor erro possível na saída do modelo, para isso, definimos uma função de custo J, que deve ser minimizada, indicando os avanço no processo de treinamento na medida em que seu valor diminui. Portanto temos eq.19:

$$J(\Theta) = -\frac{1}{m} \left[\sum_{i=1}^{m} \sum_{k=1}^{K} y_{k}^{(i)} \log(h_{\Theta}(x^{(i)}))_{k} + (1 - y_{k}^{(i)}) \log(1 - h_{\Theta}(x^{(i)})_{k})\right] + \frac{\lambda}{2m} \sum_{l=1}^{L-1} \sum_{i=1}^{sl} \sum_{j=1}^{sl+1} (\theta_{jl}^{l})^{2}$$
(19)

sendo:

K =Número de classes do problema;

m = Número de amostras no conjunto de dados de treinamento;

sl = Número de unidades da camada;

L = Número de camadas;

 λ = Fator de regularização;

 $y_k^{(i)}$ = Variável resposta da amostra selecionada referente a posição i e classe k;

 $x^{(i)}$ = Conjunto de variáveis independentes referente a posição *i*;

 h_{Θ} = Função hipótese do modelo;

A eq.19 é uma generalização da função de custo do algoritmo de regressão logística, e o seu resultado direciona o processo de aprendizado da rede neural. Com a função de custo definida, podemos definir o algoritmo de *backpropagation*, que irá calcular as derivadas parciais referentes a cada camada, começando pela camada de saída e partindo para as demais camadas, excluindo a primeira camada que é a de entrada, propagando assim o erro obtido pelo modelo para cada estágio da rede neural, possibilitando a atualização dos pesos W conforme sua influência na resposta do modelo. Um pseudo código do *backpropagation* é mostrado em algoritmo 1:

O resultado obtido pelo algoritmo $D_{ij}^{(l)}$ é numericamente igual a $\frac{\partial}{\partial \Theta_{ij}^{(l)}} J(\Theta)$, essas derivadas parciais correspondem a otimização dos valores de w_{ij} em cada camada, convergindo para a redução de $J(\Theta)$. Para maiores detalhes e um tutorial passo a passo sobre o treinamento de uma rede neural consulte (WYTHOFF, 1993).

3.2 REGULARIZAÇÃO PARA APRENDIZADO PROFUNDO

Um dos problemas enfrentados no treinamento de redes neurais profundas é equilibrar a capacidade do modelo em realizar da melhor forma possível sua tarefa no conjunto de treinamento e conseguir generalizar para novos casos vistos, por exemplo, no conjunto de dados de teste. Existem diversas técnicas que pretendem diminuir o erro de teste, sem prejudicar o erro de treino, para esse conjunto de técnicas às quais se dá o nome de regularização.

A regularização no contexto de máquina pode ser definida como modificações realizadas no algoritmo de aprendizado com a intenção de reduzir o erro de generalização do modelo mas não seu erro de treino. Esse conceito está diretamente ligado a questão da capacidade de um modelo em realizar uma tarefa, *overfitting* e *underfitting*, tópicos já mencionados anteriormente.

Existem diversas classes de regularização. A primeira delas esta relacionada a penalização de parâmetros conhecida como regularização L^2 (regressão ridge)(HOERL; KENNARD, 1970) e L^1 (regressão LASSO)(TIBSHIRANI, 1996), que penaliza os pesos do modelo minimizando a sua influência, porém melhorando o quesito de generalização do modelo.

Outros métodos mais sofisticados, contam com um treinamento contraditório da rede neural. Esta técnica consiste em introduzir pequenas pertubações nos exemplos de treino, aumentando a capacidade de generalização do modelo (GOODFELLOW; SHLENS; SZEGEDY, 2014).

Nas próximas sessões desse capítulo serão abordados as técnicas de regularização adotadas nesse trabalho, sendo que as demais técnicas, principalmente *ridge* e LASSO, são implementadas nos algoritmos de otimização.

3.2.1 Aumento do conjunto de dados (*Dataset Augmentation*)

Uma das formas de um modelo de aprendizado profundo melhorar sua capacidade de generalização é aumentando a diversidade e quantidade de dados. Porém, o número de dados disponíveis pode ser limitada ou de difícil aquisição. Uma das formas de lidar com a escassez de dados é gerar dados falsos para alimentar o modelo durante a etapa de treinamento, contudo, existem dados mais fáceis que outros de serem gerados. Um exemplo de dado falso simples de ser gerado são imagens, com algumas transformações simples como rotação, dilatação, retração e outros, é possível gerar outras informações para treinamento do modelo, sendo que esta abordagem já foi utilizada com sucesso em (JAITLY; HINTON, 2013).

No presente trabalho utilizamos uma abordagem de *data augmentation* para aumentar a diversidade de dados e balancear os dados disponíveis. Mais detalhes serão abordados na sessão 4.3.

3.2.2 Parada precoce (*Early stopping*)

Os modelos de aprendizado profundo contam com uma grande capacidade de aprendizado, por conta do grande número de camadas e unidades em sua arquitetura, sendo possível que modelos de aprendizado profundo, em suas primeiras iterações (epochs¹) reduza o erro de treino e de validação. Conforme o número de *epochs* crescem, o erro de treinamento tende a continuar diminuindo, porém pode haver um descolamento entre o erro de treino e de validação, esse formato de U no erro de validação indica que o modelo esta sobre-ajustando os dados de treino perdendo assim sua capacidade de generalização. Podemos observar na figura 28 o comportamento dos erros de treinamento e validação ao longo do tempo.

¹Uma epoch representa uma rodada de treinamento em todo o conjunto de dados disponível.



Figura 28 – Erro de treinamento e validação ao longo do tempo (*epochs*) e ponto ótimo de parada

Fonte: Autor "adaptado de"Goodfellow et al., 2016, p.243

A estrategia da parada precoce monitora os dois valores de erro, e quando se detecta uma regressão na melhoria do erro de validação, o treinamento naquele ponto é parado, muitas vezes antes do número de *epochs* definidos previamente ao treino, preservando assim, os parâmetros obtidos naquele momento, sendo adotados como os melhores para a tarefa em questão. Apesar de simples, essa técnica é efetiva e uma das mais aplicadas durante o treinamento de redes neurais profundas(CARUANA; LAWRENCE; GILES, 2001).

3.2.3 Métodos Ensemble

As técnicas de *Ensemble* visam diminuir o erro de validação através da união de modelos treinados em diferentes cenários, seguindo duas estratégias gerais. A primeira delas é conhecida como *bagging* (BREIMAN, 1994). Essa técnica, utiliza o mesmo modelo treinando em k diferentes conjunto de dados, todos com o mesmo tamanho do conjunto de dados original, porém são amostrados com reposição. Como são conjunto de dados diferentes, os erros de validação de cada modelo serão diferentes, sendo a resposta média entre os modelos a mais plausível. Outra abordagem é a *boosting*(DRUCKER et al., 1994), os modelos são construídos de forma aditiva, sempre a entrada do próximo modelo será o conjunto de dados onde o modelo anterior desempenhou pior, desta forma o resultado final é um modelo composto por diversos modelos especialistas em uma região do espaço do conjunto de dados.

3.2.4 Dropout

A técnica de *Dropout*(SRIVASTAVA et al., 2014) é outro método simples porém efetivo em busca do equilíbrio entre o erro de treino e de validação. O método consiste em apagar, de forma aleatória, um conjunto de unidades da rede neural durante o processo de treinamento, desde que essas unidades não façam parte da saída. Ao realizar essa especie de poda da rede, o modelo diminui sua capacidade de aprendizado diminuindo os efeitos de sobre-ajuste e aumentando a capacidade de generalização. Por ter um baixo custo computacional, pode ser utilizado no lugar de técnicas de *bagging* em redes neurais muito grandes e complexas.

3.3 Redes Neurais Convolucionais

As redes neurais convolucionais (*Convolutional Neural networks* (CNN)), são redes neurais especializadas em processar dados dispostos em formato de grade, como series temporais (1D) e imagens (2D). O conceito de CNN foi introduzido em (LECUN et al., 1989) e desde então vem apresentando resultados relevantes na área de aprendizado de máquina (LA-WRENCE et al., 1997; KALCHBRENNER; GREFENSTETTE; BLUNSOM, 2014). Essa rede neural tem em seu nome convolucional pois aplica uma operação matemática chamada convolução. Essa operação matemática expressa o quanto o formato de uma função interfere na outra, sua definição eq.20.

$$(f * g)(t) \stackrel{\Delta}{=} \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau$$
(20)

Sendo assim, nosso primeiro argumento $f(\tau)$ é conhecido como função de entrada, $g(t-\tau)$ é conhecido como *kernel* da convolução. Na fig.29 temos o esquema de uma convolução em duas dimensões representada.



Fonte: Autor "adaptado de "Goodfellow et al., 2016, p.330

O primeiro ponto é referente à conectividade esparsa que as redes CNN, representando um número menor de parâmetros em relação à abordagens totalmente conectadas. Em um problema de classificação de imagem, por exemplo, uma abordagem de redes neurais totalmente conectadas, teríamos uma imagem com muitos pixels, e cada pixel dessa matriz, teria uma unidade representando na camada de entrada, e todas unidades de entrada estariam conectados com os itens da próxima camada, e assim sucessivamente. Com a adoção de uma camada convolucional, tem-se uma queda no número de parâmetros de treinamento, pois a adoção do tamanho do *kernel* que irá realiza a operação de convolução é muitas vezes menor que a dimensão da matriz de entrada, diminuindo o número de parâmetros de treinamento e reduzindo o tempo total de execução da rede(GOODFELLOW et al., 2016). Um segundo ponto muito importante sobre redes neurais convolucionais é o compartilhamento de parâmetros. Em uma abordagem totalmente conectada, temos que cada peso se refere a uma região dos dados de entrada e esse peso é utilizado uma única vez, por outro lado, em redes convolucionais temos um *kernel* que é menor que o dado de entrada, sendo assim, um mesmo filtro é usado em várias regiões do dado de entrada, e ao invés de se ter um aprendizado localizado em uma específica parte dos seus dados, o *kernel* visita todas as regiões do seu dado aprendendo características globais ao invés de características pontuais. Esta característica otimiza o processo de treinamento e armazenamento da rede neural, pois os mesmos pesos são compartilhados com diversos pontos da rede neural reduzindo o número de parâmetros a serem treinados, otimizados e armazena-dos(GOODFELLOW et al., 2016). Por fim, um terceiro ponto que torna a adoção de redes neurais convolucionais interessantes em alguns cenários é a propriedade de equivariância. Uma função é dita como equivariante, quando o valor de sua saída muda com a mesma intensidade a mudança aplicada na entrada. Essa propriedade é muito importante em tarefas de detecção de objetos em imagens, pois sem isso, uma rede teria que ter um conjunto de dados de treinamento onde o objeto a ser detectado estivesse em todas posições e rotações possíveis. Com essa propriedade, quando uma característica é detectada, uma representação dessa característica é refletida na saída, permitindo assim seu correto uso independente de posição. Essa característica também se aplica no processamento de séries temporais, uma característica do sinal de entrada que muda no tempo, tem sua representação equivalente representada no resultado do processamento da rede.

3.3.1 Pooling

Uma camada convolução em si, onde os filtros (*kernel*), de forma paralela, procuram no conjunto de dados de entrada as principais características e as evidenciam para que façam parte das informações importantes para o restante da rede. A próxima etapa é a de detecção, onde os sinais gerados pela camada anterior passam por funções não lineares como a ReLU. Por fim, o terceiro e ultimo estágio é responsável por modificar os resultados do estágio anterior. Esta etapa é muito importante, pois com ela é possível amenizar as pequenas variações que possam ocorrer de uma mesma característica de entrada sem que isso se reflita como uma nova informação a ser adotada. As duas operações de *pooling* são a *max pooling* (ZHOU; CHELLAPPA, 1988) e *average pooling*. A primeira delas a *max pooling*, preserva a máxima ativação de cada região do filtro enquanto a *average pooling* realiza a média dois valores de cada região. A fig.30 exemplifica cada uma dessas operações.



Figura 30 – Exemplo de filtro max pooling e average pooling para um tamanho 2x2

Average pooling

Fonte: Autor

3.4 REDES NEURAIS RECORRENTES

Assim como as redes neurais convolucionais são utilizados para dados que apresentam estrutura em formato de grade, como imagens por exemplo, temos as redes neurais recorrentes que são especializadas em processar um sequência de dados, como textos, voz, vídeo e séries temporais. Uma das principais vantagens desse tipo de rede neural é a capacidade de processar informações de diferentes tamanhos, o que não é possível em redes neurais convencionais. A capacidade de compartilhamento de parâmetros presentes também nas redes neurais recorrentes as torna interessante no cenário em que uma mesma informação pode se repetir em sequências diferentes do conjunto de dados de entrada, por exemplo nas frases: "Paula nasceu na França e fala francês."e "Paula fala francês e nasceu na França". Desta duas frases a informação aparece em lugares diferentes em cada frase. Uma rede neural tradicional, treinada em tamanhos fixos de frases, teria que aprender para cada posição e de forma individual, toda as regras da língua e ainda teria a limitação do tamanho de texto, já as redes neurais recorrentes apresentam a capacidade de compartilhar seus parâmetros em todas as etapas tornando o processo de aprendizado desse contexto mais eficiente. Como citado na sessão referente às redes neurais convolucionais,

estas apresentam a capacidade de compartilhar parâmetros também, e em um primeiro momento podem figurar como possíveis candidatas na utilizados de dados com estrutura temporal, porém sua aplicação é limitada, pois o compartilhamento de parâmetros se limitam a poucas unidades, tornando sua capacidade temporal pequena tornando o seu uso pouco interessante para grande cadeias de informação (LANG; WAIBEL; HINTON, 1990).

3.4.1 Redes neurais recorrentes e o Desdobramento

As redes neurais recorrentes apresentam características de sistemas dinâmicos, então uma forma de se entender o seu funcionamento é olhando para esses sistemas com comportamento dinâmicos, que já foram representados através de redes neurais recorrentes (TSOI; BACK, A. D., 1994; TSOI; BACK, A., 1997). O estado de um sistema dinâmico é formado por um conjunto de valores, necessários e suficientes para se caracterizar a situação deste sistema, ou seja, dado o conjunto de variáveis de um sistema temos a sua situação naquele instante de tempo dado o conjunto de informações de entradas que podem afetar esse sistema, o que pode se caracterizar como uma memória daquilo que aconteceu em um instante de tempo t, sendo assim temos a eq.21. Esta característica é importante em ser definida, pois ela se repete quando se trata de redes neurais recorrentes.

$$s^{(t)} = f(s^{(t-1)}; \theta)$$
(21)

O termo recorrente vem da expressão presente em eq.21, que faz referencia ao próprio estado em (t-1). Para um número finito de passos no tempo τ , podemos reescrever eq.21 de tal forma que ela possa ser representada de uma forma acíclica, como uma rede neural tradicional. Um exemplo para $\tau = 3$ é demonstrado em eq..

$$s^{(3)} = f(s^{(2)}; \theta) = f(f(s^{(1)}; \theta); \theta)$$
(22)

Com esse desdobramento é fácil visualizar um relação temporal, em que um estado depende diretamente do seu estado anterior. A representação gráfica das eq.21 e 24 pode ser visto em fig.31.
Figura 31 – Representação gráfica da equação de estados de um sistema dinâmico desdobrado. Cada nó representa um estado, em um dado tempo t e a função f que mapea o estado t em t + 1.



Fonte: Autor "adaptado de"Goodfellow et al., 2016, p.369

É importante notar na eq.21 o termo θ , que representa os parâmetros da função f, que é o mesmo presente em todos os instantes de tempo, esse compartilhamento de parâmetros permite aprender ao longo do tempo toda a estrutura de dados de forma eficiente. Para um sistema que é influenciado por um sinal externo $x^{(t)}$, temos eq.23.

$$s^{(t)} = f(s^{(t-1)}, x^{(t)}; \theta)$$
(23)

Muitas redes neurais recorrentes, utilizam a eq.23, como a equação de estado de suas camadas ocultas, porém substituindo s para h (do inglês *hidden*, oculto). A nova representação gráfica da eq.23 pode ser visto em fig.32.

Figura 32 – Representação gráfica de uma rede neural recorrente sem saída. A caixa preta representa um atraso de t + 1 no estado atual



Fonte: Autor "adaptado de"Goodfellow et al., 2016, p.370

A representação de uma rede neural recorrente desdobrada é vista na fig.32 do lado direito da figura. É possível notar que o número de unidades que essa versão desdobrada esta ligada ao tamanho do vetor de entrada X, e o processo de obtenção desse grafo estendido é chamada de desdobramento e é realizada pela função f. O gráfico desdobrado após t passos pode ser representado por $g^{(t)}$, descrito em eq.24.

$$h^{(t)} = g^{(t)}(x^{(t)}, x^{(t-1)}, x^{(t-2)}, \dots, x^{(2)}, x^{(1)}) = f(h^{(t-1)}, x^{(t)}; \theta)$$
(24)

A função $g^{(t)}$, da conta de toda a sequência passada de X e calcula o valor do estado para o valor de entrada atual, essa abordagem permite que o modelo de aprendizado, independendo do tamanho da sequência, sempre tenha o mesmo tamanho de entrada e permite que a mesma função f seja utilizada em todos os intervalos de tempo. Após a definição da estrutura básica de uma rede neural recorrente baseada no modelo de sistemas dinâmicos, conseguimos expandir essa estrutura e finalmente definir uma estrutura completa com entrada e saída, valendo-se dos princípios da capacidade de desdobramento da rede base. A fig. 33 mostra a estrutura completa de uma rede neural convolucional.





Fonte: Autor "adaptado de"Goodfellow et al., 2016, p.373

A estrutura representada na fig.33 mapeia x para uma sequência de valores o. A unidade L computa a diferença entre a saída mapeada o com a respostas esperada y. A conexão entre a camada de entrada e as camadas ocultas é feita através da matriz de pesos U. A conexão entre as camadas ocultas e definida pela matriz de pesos W, e finalmente a conexão das camadas ocultas com a saída e feita pela matriz de pesos V. Com essa estrutura básica é possível desenvolver as equações de propagação *feedfoward* para as redes neurais convolucionais. O processo de propagação começa com o esta inicial $h^{(0)}$, e para cada passo de tempo de t = 1 a $t = \tau$, temos o seguinte conjunto de equações:

$$a^{(t)} = b + Wh^{(t+1)} + Ux^{(t)}$$
⁽²⁵⁾

$$h^{(t)} = g(a^{(t)}) \tag{26}$$

$$o^{(t)} = c + V h^{(t)} (27)$$

$$\hat{y} = softmax(o^{(t)}) \tag{28}$$

Onde os vetores b e c são *bias*. Esse é o funcionamento básico de uma rede neural recorrente, para aprofundamento no assunto e maiores detalhes sobre o processo de treinamento e de otimização pode ser encontrado em (GOODFELLOW et al., 2016).

3.4.2 O problema do aprendizado de longo prazo

Um dos grandes problemas com o processo de aprendizado com dependência de longo tempo é a propagação dos gradientes em diversos estágios tendem a desaparecer, em sua maioria, ou tende a explodir os valores, são menos prováveis porém mais danosos para o processo de otimização. Mesmo com parâmetros controlados e estáveis, as iterações a longo prazo tendem a tornar os valores dos parâmetros pequenos tendendo a zero (HOCHREITER, 1998; BENGIO; FRASCONI; SIMARD, 1993). Existem algumas propostas que visam diminuir esse problema presente em redes neurais recorrentes, que são a adoção das *Long Short Term Memory* (LSTM) e *Gated Recurrent Unit* (GRU).

3.4.3 Long Short Term Memory (LSTM)

As células LSTM foi proposta em (HOCHREITER; SCHMIDHUBER, 1997) para lidar com o problema dos gradientes que somem (assumem valores nulos) ou os gradientes que explodem (que assumem valores infinitos), dificultando o procedimento de treinamento da rede neural recorrente. Umas das principais contribuições dessa arquitetura é o circuito interno onde os valores dos gradientes conseguem circular por um longo período de tempo. Esse recurso foi aprimorado com a adição de um peso para esse circuito que é condicionado com o estado atual ao invés de ser um valor fixo (GERS; SCHMIDHUBER; CUMMINS, 1999). As LSTM se mostraram eficientes em diversos cenários como reconhecimento de fala (GRAVES; JAITLY; MOHAMED, 2013a), geração de escrita manual (GRAVES, 2013) e até mesmo em geração de legendas para imagens (KIROS; SALAKHUTDINOV; ZEMEL, 2014). O estado da célula LSTM é representado por $s_i^{(t)}$, que circula pelo caminho interno da célula. Este caminho é controlado por $f_i^{(t)}$ (para o passo t e a célula i), essa função é nomeada de portão do esquecimento (*forget gate*). Este portão tem a função de controlar o peso que o sinal $s_i^{(t)}$, variando o valor desse peso entre 0 e 1, através de uma função sigmoid, conforme eq.29.

$$f_i^{(t)} = \sigma(b_i^f + \sum_j U_{i,j}^f x_j^{(t)} + \sum_j W_{i,j}^f h_j^{(t-1)})$$
(29)

onde,

 $x^{(t)}$: Valor de entrada atual;

 $h^{(t)}$: Vetor da camada oculta atual;

b^f: Vetor de Viés (*biases*);

U^f: Pesos de entrada do portão de esquecimento;

W^f: Pesos recorrentes do portão de esquecimento;

A equação de atualização do estado interno $s_i^{(t)}$ é dado pela eq.30.

$$s_i^{(t)} = f_i^{(t)} s_i^{(t-1)} + g_i^{(t)} \sigma(b_i + \sum_j U_{i,j} x_j^{(t)} + \sum_j W_{i,j} h_j^{(t-1)})$$
(30)

onde,

b: Vetor de Viés (biases);

U: Pesos de entrada da célula LSTM;

W: Pesos recorrentes da célula LSTM;

O portão de entrada $g_i^{(t)}$, também é uma função sigmoide, similar ao portão de esquecimento, porém é obtida com o seus próprios parâmetros, como visto em eq.31.

$$g_i^{(t)} = \sigma(b_i^g + \sum_j U_{i,j}^g x_j^{(t)} + \sum_j W_{i,j}^g h_j^{(t-1)})$$
(31)

b^g: Vetor de Viés (*biases*);

U^g: Pesos de entrada do portão de esquecimento;

W^g: Pesos recorrentes do portão de esquecimento;

O último portão é o de saída $q_i^{(t)}$, e é calculado de acordo com as eq.32 e 33.

$$h_i^{(t)} = \tanh s_i^{(t)} q_i^{(t)}$$
(32)

$$q_i^{(t)} = \sigma(b_i^o + \sum_j U_{i,j}^o x_j^{(t)} + \sum_j W_{i,j}^o h_j^{(t-1)})$$
(33)

onde,

b°: Vetor de Viés (biases);

Uº: Pesos de entrada do portão de saída;

Wº: Pesos recorrentes do portão de saida;

O conjunto destes três portões entrada, esquecimento e saída, formam a célula LSTM, e permitem o aprendizado das características mais importantes ao longo do tempo. A fig.34 mostra a representação gráfica de uma célula LSTM.

Figura 34 – Representação de uma rede neural completa e sua representação desdobrada no tempo.



Fonte: Autor "adaptado de"Goodfellow et al., 2016, p.405

3.4.4 Gated Recurrent Unit (GRU)

Uma alternativa mais enxuta a LSTM é a célula GRU, porém com desempenho nas mesmas tarefas semelhantes (CHO et al., 2014). A principal diferença entre as duas células é o

controle simultâneo do sinal de esquecimento e o sinal de atualização do estado da unidade. As equações são descritas de eq.34 a 36.

$$h_i^{(t)} = u_i^{(t-1)} h_i^{(t-1)} + (1 - u_i^{(t-1)}) \sigma(b_i + \sum_j U_{i,j} x_j^{(t-1)} + \sum_j W_{i,j} r_j^{(t-1)} h_j^{(t-1)})$$
(34)

$$u_i^{(t)} = \sigma(b_i^{(u)} + \sum_j U_{i,j}^u x_j^{(t)} + \sum_j W_{i,j}^u h_j^{(t)})$$
(35)

$$r_i^{(t)} = \sigma(b_i^{(r)} + \sum_j U_{i,j}^r x_j^{(t)} + \sum_j W_{i,j}^r h_j^{(t)})$$
(36)

onde,

u: Portão de atualização;

r: Portão de reinicio;

Por ter um número reduzido de parâmetros a serem otimizados, o tempo de treinamento de um mesmo modelo contendo células GRU ao invés de células LSTM é relativamente inferior, bem como o tamanho de armazenamento de um modelo treinado.

3.4.5 Proposta

O presente trabalho tem como objetivo a criação de métodos para estudo de fenômenos SEE, através da simulação da estrutura de um dispositivo p-mos 3N163 e a incidência de partículas ionizantes nesse dispositivo simulado, e a criação de um modelo de redes neurais que seja capaz de classificar corretamente os sinais obtidos em experimentos realizados em campo, utilizando uma abordagem de *Data augmentation* extrapolado para um sinal unidimensional. A união destas duas frentes será realizada através da avaliação dos sinais simulados pelo modelo de redes neurais com a finalidade de encontrar pontos em comum entre o sinal simulado e os dados reais.

4 MATERIAIS E MÉTODOS

Neste capítulo serão apresentados os materiais e métodos utilizados para o desenvolvimento deste estudo. Desde o acelerador de partículas, utilizado para a geração das partículas ionizantes ao aparato desenvolvido para aquisição dos dados para posterior analise.

4.1 ACELERADOR PELLETRON

O acelerador Pelletron 8UD (Figura 35) é um acelerador eletrostático da categoria *Tandem*, que pode alcançar energias superiores a 100 MeV para experimentos que necessitam a utilização de íons mais pesados.

Figura 35 – Visão geral do acelerador Pelletron.



Fonte:(LANF-USP, 2016)

4.1.1 Configuração experimental

As configurações possíveis permitem, além de experimentos dentro da câmara de espalhamento no vácuo, irradiação em dispositivos fora deste ambiente, nos casos em que os alvos sejam difíceis de serem utilizados no vácuo. Usando uma janela com uma folha fina de alumínio ou mylar, é possível encaminhar os íons de dentro da câmara para o dispositivo em questão. Nesta configuração, a janela pode ser posicionada em ângulos de espalhamento de 15 ou 45 graus (Figura 36), porém limitada a feixes externos de prótons de até ${}^{28}Si$, devido a alta perda de energia no ar e nas janelas de saída.

Figura 36 – Esquemático da configuração experimental usadas nas radiações.



Fonte: Autor "adaptado de"V. A P Aguiar et al., 2014, p.2

Na data de realização dos experimento os íons disponíveis no LAFN são: ${}^{1}H$, ${}^{7}Li$, ${}^{10}B$, ${}^{12}C$, ${}^{16}O$, ${}^{19}F$, ${}^{28}Si$, ${}^{35}Cl$, ${}^{48}Ti$, ${}^{63}Cu$ e ${}^{107}Ag$. Na figura 37 é possível consultar os valores de LET utilizados nos experimentos em função da profundidade de penetração no silício.

Figura 37 – Íons pesados utilizados nos experimentos. LET em função da profundidade no silício.



Fonte: Autor "adaptado de"N. H. Medina et al., 2015, p.3

A tabela 1 discrimina os íons utilizados no estudo, que incluem ${}^{12}C$, ${}^{16}O$, ${}^{19}F$, ${}^{28}Si$, ${}^{35}Cl$, ${}^{63}Cu$ e ${}^{107}Ag$. Estes íons foram espalhados a 15 graus através de uma folha de ouro de 275 μ g.cm⁻², o que disponibiliza LETs de 2 a 45 MeV/mg/cm². O espelhamento pela folha de

ouro em um ângulo conhecido é feita com o objetivo de diminuir o fluxo de partículas e prover uma distribuição uniforme no dispositivo.

Íon	Energia (MeV)	Penetração no Si(μ m)	LET no Si(MeV/mg/cm ²)
^{12}C	45,0	55,4	2,6
^{16}O	52,5	38,9	4,6
^{19}F	42,0	23,8	6,8
^{28}Si	66,0	21,6	12,8
^{35}Cl	75,0	19,5	17,1
^{63}Cu	82,0	14,1	30,7
107Ag	97,5	13,1	39,8

Tabela 1 – Energia do feixe, profundidade de penetração e LET superficial.

Fonte: Autor "adaptada de"N. H. Medina et al., 2015, p.3

Um segundo experimento foi realizado para obtenção de novos dados para avaliar efeitos previstos em simulação. Nesta segunda rodada foi utilizada a nova linha disponível para emissão das partículas, presente no LAFN nomeada de SAFIIRA (figura 38).

Figura 38 – Visão geral da nova linha SAFIIRA.



Fonte: Autor

Esta nova linha foi construída com o objetivo de atender a padrões internacionais que regem os estudos de radiações ionizantes, segundo alguns destes requerimentos, como por exemplo, baixo fluxo de íons, área de cobertura elevada, alta uniformidade, manipulação da amostra e diversas combinações de íons e energias (AGUIAR, V. et al., 2017). Nesta segunda rodada do experimento foi utilizado um único feixe de íons de ${}^{16}O$ utilizando uma energia de 63MeV. A tabela 2 contém informações sobre o feixe utilizado na segunda rodada.

Tabela 2 – Energia do feixe, profundidade de penetração e LET no silício.

Íon	Energia (MeV)	Penetração no Si(μ m)	LET no Si(MeV/mg/cm ²)
^{16}O	63	50,4	4,6

Fonte: Autor

4.1.2 Dispositivos sob teste

Para a realização deste estudo foi utilizado o transistor PMOS 3N163 (3N163..., 2016).

A primeira configuração utilizada nos testes, usou uma polarização com 0,13V entre a dreno e fonte (VDS) e -4,5V entre porta e fonte (VGS). A corrente de dreno foi amplificada através de um amplificador operacional LM324, e a medição desta corrente realizada através de um conversor A/D¹ de 12 bits de aproximação sucessiva precedida por um circuito de *Sample-and-Hold*. Devido a baixa taxa de amostragem em relação a duração de um evento SEE, apenas 0,02% dos eventos possíveis foram registradas (MEDINA, N H et al., 2013), surgindo a necessidade da criação de um novo sistema para realização da coleta de dados.

A segunda configuração, implementou uma série de melhorias no experimento. A primeira delas foi o circuito de polarização (Figura 39), que conferiu ao experimento a capacidade de polarizar o dispositivo sob teste em diversas regiões de operação. A taxa de amostragem teve um aumento significativo com a utilização de um osciloscópio da fabricante Rohde & Schwarz modelo 1-GHz RTO1012, que fornece a capacidade de amostragem de 10Gsamples/s, equipado com um recurso de *trigger*² digital de tempo real. Este sistema de monitoramento, se mostrou de grande valor por conta da economia de memória, pois apenas registrava os eventos nos quais os valores de tensão excediam um determinado limiar.

¹Conversor analógico-digital.

 $^{^{2}}$ O *trigger* é um recurso disponível em osciloscópios, que permite o registro de eventos que ocorrem a partir de valor de tensão, por exemplo, específico.



Figura 39 – Esquemático do sistema de polarização do transistor 3N163.

Fonte: Autor "adaptado de"N. H. Medina et al., 2015, p.2

O dispositivo sob teste foi posicionado dentro da câmara de vácuo (Figura 40), utilizandose de uma placa onde era localizado o circuito de polarização, conectores e suportes para o transistor. O dispositivo foi irradiado a 15 graus, conforme esquema representado na figura 41 (SILVEIRA et al., 2015).



Figura 40 – Dispositivo sob teste na câmara de vácuo.

Fonte: Autor "adaptado de"Silveira et al., 2015, p.5



Figura 41 – Visualização esquemática do arranjo utilizado para a realização dos testes de SEE.

Fonte: Autor "adaptado de" (MEDINA, N. H. et al., 2015), p.1

Durante as simulações de radiação foram detectadas outras formas de onda resultantes da ação das partículas ionizantes na região de porta do dispositivo 3N163 simulado, além do tradicional sinal de pico esperado. Desta forma novas rodadas de experimentos se fizeram necessárias para a obtenção de mais dados para serem confrontados com os resultados simulados. Para esse novo experimento, usou-se a mesma configuração de circuito polarizador descrito anteriormente, pois mostrou-se um bom aparato para a obtenção dos pulsos decorrentes da radiação, porém a disposição da placa na câmera se deu de forma diferente da anterior, nesta nova linha o dispositivo se encontrava diretamente posicionado na direção do feixe, não houve espalhamento do feixe feito por uma lâmina de ouro, desta forma foi possível registrar um numero maior de eventos durante o experimento. A figura 42 contempla o posicionamento do dispositivo a ser testado e toda o aparato de polarização necessário para seu funcionamento.

Figura 42 – Disposição do *Dispositive Under Test* (DUT) na câmera utilizado no segundo experimento.



Fonte: Autor

Em todas as configurações o encapsulamento do dispositivo é removido, com a finalidade de expor as regiões sensíveis diretamente aos feixes irradiados. A eliminação do encapsulamento, nestes casos, ajuda nos cálculos posteriores da seção de choque, pois é conhecida a fluência dos feixes sem a interferência impostas pelo encapsulamento do dispositivo. Na figura 43 é possível observar o componente desencapsulado.



Figura 43 – Detalhamento do dispositivo desencapsulado.

Fonte: Autor

4.2 SIMULADORES

Nesta seção serão explorados os recursos de software utilizados para os cálculos de *Linear Energy Transfer* (LET), dados que são necessários para a inserção na simulação da radiação do dispositivo, bem como o simulador Sentaurus.

4.2.1 SRIM/TRIM

O *Transport of Ions in Matter* (TRIM) é formado por um grupo de programas utilizados para calcular o poder de freamento e alcance dos íons na matéria, usando o tratamento da mecânica quântica para as colisões entre íon e átomo. Com esse conjunto de ferramentas é possível calcular distribuição de íons, danos no material, dano total de cascata, passos de colisão monocamada, pulverização catódica de superfície, cascatas de nêutrons, elétrons, energia do íon, ângulo e posição (ZIEGLER; ZIEGLER; BIERSACK, 2010), (ZIEGLER; BIERSACK; ZIEGLER, 2008).





Fonte:Autor

Neste trabalho o TRIM foi utilizado para calcular os valores de LET e suas respectivas penetrações nos materiais para cada íon utilizado no experimento real. Com os valores obtidos através do TRIM foi possível obter o valor de ionização que cada partícula ou seja a Δ_E (energia perdida na trajetória do feixe), com esta informação é possível obter o LET em cada caso, a figura 45 demonstra a ionização ao longo da trajetória do dispositivo.

Figura 45 – Gráfico demostrando a perda de energia de íons ${}^{16}O$ com energia de 63MeV em um corpo composto por metal/óxido de silício/silício.



Fonte:Autor

Através do resultado demostrado na figura 45 é possível calcular o valor do LET utilizando a equação 37 demonstrada abaixo:

$$LET = \frac{\bar{E_L}}{23,21} \tag{37}$$

A tabela 3 mostra os valores obtidos através do TRIM e os dados relacionados a cada entrada, estes valores são usados como entrada para a simulação da emissão de uma partícula ionizada no dispositivo.

Tabela 3 – Energia do feixe, profundidade de penetração e LET no silício para diferentes regiões do dispositivo.

Íon	Energia (MeV)	Penetra	ção por região(µm)	LET po	or região(MeV/mg/cm ²)
		Dreno	Porta	Dreno	Porta
^{12}C	45,0	51,5	51,34	3,55	3,55
^{16}O	52,5	39,8	36,6	5,75	5,65
^{28}Si	66,0	15,12	14,35	13,54	13,47
^{35}Cl	75,0	11,94	12,15	17,2	17,12

Fonte: Autor

4.2.2 Sentaurus

A simulação do dispositivo e da partícula ionizante são feitos através da ferramenta Sentaurus da empresa Synopsys. São usadas duas ferramentas de TCAD da Synopsys, *Sentaurus Structure Editor* (SDE) e Sentaurus Device. O SDE é responsável pela criação de uma estrutura contendo as grades de pontos, os perfis de dopagem do dispositivo e seus respectivos contatos elétricos. Já a ferramenta Sentaurus Device é um estimulador de dispositivo, 2D e 3D, que fornece a possibilidade de simular uma grande gama de dispositivos. Provê simulações multidimensionais, eletro-térmicas, dispositivos de modo-misto³ e circuitos. Incorpora modelos físicos avançados e métodos numéricos robustos para as simulações (SYNOPSYS, 2009).

Para a construção e simulação do dispositivo utilizado neste trabalho foram utilizados os seguintes modelos:

- A) SRH: Modelo *Shockley–Read–Hall* (SRH) responsável pela contabilização da recombinação através de níveis de defeitos profundos.
- B) Auger: Modelo responsável pela taxa de recombinação entre bandas.
- C) DopingDependence: Utilizando este modelo é ativada a degradação da mobilidade devido a dispersão de impurezas. Este modelo apresenta três modos Masetti, Arora e UniBo, escolhidos de acordo com material do dispositivo. Para o silício, o modelo Masetti(MASETTI; SEVERI; SOLMI, 1983) é o indicado.
- D) Enormal: Modelo utilizado para calcular o campo elétrico que se situa de forma perpendicular ao canal e ao óxido de porta.
- E) PhuMob: O modelo *Philips unified mobility model* (PhuMob), proposto por Klaassen (KLAASSEN, 1992), unifica a descrição da mobilidade dos portadores majoritários e minoritários, além de descrever a dependência da mobilidade em função da temperatura levando em consideração o espalhamento dos pares elétron-lacuna, impurezas ionizadas por portadores de carga e agrupamento de impurezas.
- F) HighFieldSaturation: Na presença de altos campos elétricos, a velocidade de deriva dos portadores não é proporcional ao longo do campo elétrico. Utilizando este modelo o efeito é considerado, e para isso leva em conta três submodelos: o modelo de mobilidade atual, o modelo de saturação de velocidade e o modelo de força motriz.

³Simulação de sistemas que unem sinais analógicos e digitais.

- G) IncompleteIonization: No silício os dopantes podem ser considerados completamente ionizados à temperatura ambiente. Porém, quando os níveis de impureza são relativamente profundos em comparação a energia térmica a ionização incompleta deve ser considerada, para isso é utilizado o modelo IncompleteIonization.
- H) CarrierCarrierScattering: Este modelo contabiliza a degradação devido ao espalhamento entre portador-portador. Dois modelos são suportados para descreverem esse espelhamento, o primeiro modelo é baseado pelo trabalho de Choo (CHOO, 1972) e o segundo por Fletcher (NEVILLE H. FLETCHER, 1957).

4.2.3 Simulação

Para avaliar e extrair informações dos resultados obtidos nos experimentos de radiação foi criada uma simulação do dispositivo real. Para a criação do modelo de simulação, foram avaliadas imagens do dispositivo real sem o encapsulamento, para estimar as dimensões de construção do dispositivo(figura 46).

Figura 46 – Visualização esquemática do arranjo utilizado para a realização dos testes de SEE.



Fonte: Autor "adaptado de"Alves et al., 2011, p.2356

Este dispositivo em particular é uma combinação de sub-sistemas construídos de tal maneira que as regiões de dreno, porta e fonte fossem intercalados sucessivamente. A figura 47 mostra o resultado final obtido na construção do dispositivo.

Figura 47 – Visualização esquemática do arranjo utilizado para a realização dos testes de SEE.

Y Z X



Fonte: Autor

4.2.3.1 Dispositivo

Os parâmetros de tecnologia, como X_{OX} e os perfis de dopagem entre outros parâmetros que foram adotados inicialmente são descritos em (CARVAJAL et al., 2009). Com esses valores foi gerada a primeira estrutura do dispositivo (figura 48).



Figura 48 – Visualização da primeira estrutura do dispositivo 3N163 gerada.

Fonte: Autor

4.2.3.2 Grade

Um dos fatores cruciais de uma boa simulação é a geração da grade, pois através dela que os algoritmos efetuam os cálculos. Por se tratar de um dispositivo relativamente grande (com dimensões que apresentam valores na ordem de centenas de μ m) e tridimensional, os critérios de geração da grade devem ser vistos com grande atenção, pois o tempo de simulação está ligado diretamente com o número de pontos gerados.

A primeira versão de grade gerada foi feita sem refinamentos, contando com dois trechos de definição. O primeiro trecho contemplava desde os contatos elétricos até a profundidade de 100 μ m no corpo do silício, o restante foi gerado utilizando-se um parâmetro na definição da grade que possibilita a geração de um número de vértices proporcional a dopagem do dispositivo.

Posteriormente, a grade sofreu alterações para otimizar o tempo de simulação. Foram definidas regiões especificas para cada canal, restringindo a grande densidade de pontos nessas regiões de interesse, diminuindo inclusive a profundidade da grade para contemplar apenas a formação do canal. Para a simulação da radiação considerou-se a profundidade máxima que seria atingida pelos íons selecionados. A grade gerada pode ser observada na figura 49, nesta

imagem é mostrado um corte realizado na região escolhida para a incidência da partícula, mostrando a grade geral e seus detalhamentos.

Figura 49 - Corte ao transversal detalhando a estrutura de grade adota para as simulações.



Fonte: Autor

É importante salientar que grande parte da região de silício não contribui com o valor final de corrente, por conta disso apenas as regiões próximas as interface óxido/silício recebem um refinamento maior, pois a maior parte da corrente circula nessa região. Outro ponto de destaque é o caminho percorrido pela partícula, que gera pares elétron-lacuna pelo seu percurso e contribuem com a corrente que circula pelo dispositivo.

4.3 ANÁLISE DE DADOS

Os dados obtidos durante os experimentos no Pelletron apresentam um comportamento transitório, são eventos de curta duração registradas via osciloscópio. O dado registrado é resultado da coleta dos pares elétron-lacuna, provenientes do traço deixado pela passagem da partícula ionizante na estrutura do dispositivo, pelo campo elétrico presente no dispositivo, esses pares coletados são absorvidos pela corrente principal do dispositivo, sendo registrada pela

geração de um pico no valor da corrente atual do dispositivo. Cada sinal registrado apresenta 2000 pontos, que são amostrados a cada $10\eta s$, totalizando uma janela de captura de $20\mu s$ do evento. Os experimentos contaram com sete íons diferentes, e foram registrados por volta de 123000 amostras totais. Uma amostra de cada sinal pode ser observado na fig.50.



Figura 50 – Uma amostra de cada íon medido durante os experimentos.

Observando a fig.50, nota-se que cada íon tem um valor de pico especifico e uma largura de base diferente, esses parâmetros estão ligados a profundidade de penetração de cada íon, tornando essa região uma forte candidata a maior importância na classificação.

Por conta das características de cada íon, foram registradas uma quantidade diferente de eventos para cada íon da mesma categoria. Íons com maior fluência apresentavam mais registros em relação aos que apresentavam um valor menor. A figura 51 mostra a proporção de cada íon registrado durante os experimentos.

Fonte: Autor



Figura 51 – Proporção de cada elemento no total de 123000 amostras. Cada quadrado representa aproximadamente 1500 amostras.

Analisando a fig.51 é possível observar um grande desbalanceamento entre as classes. Enquanto o íon de flúor representa 33% dos casos disponíveis no conjunto de dados, o íon de prata representa apenas 1% do total, e este tipo de assimetria entre as classes pode prejudicar o processo de treinamento de um algoritmo de aprendizagem, pois pode enviesar as decisões do modelo caso esse fator não seja levado em consideração. Outro fator importante observado nos dados é a diversidade de características que o sinal medido de um mesmo íon pode apresentar, conforme observado na fig.52. Devido ao fato de cada íon atingir o dispositivo de formas diferentes como energia, profundidade de penetração e outros, cada resposta medida apresenta sua própria forma, principalmente o valor de pico do sinal, o que adiciona mais complexidade na tarefa de classificação desse sinal.

Fonte: Autor



Figura 52 – Média das amostras para o Íon de prata e o desvio padrão de cada ponto

Fonte: Autor

Esses dois fatores, desbalanceamento da base e diferentes características do sinal para um mesmo íon, nos obriga a ter uma abordagem cautelosa antes de iniciarmos o treinamento de qualquer modelo, evitando problemas durante essa fase. Como queremos que o modelo tenha um bom desempenho durante a etapa de treinamento e que seu valor de erro de validação seja o menor possível, foi desenvolvida uma técnica para aumentar a diversidade dos dados tanto para a questão do desbalanceamento, quanto para a questão da diversidade de sinais de um mesmo íon. A técnica foi inspirada pelos métodos de aumento do conjunto de dados realizados em imagens (PEREZ; WANG, 2017), que consiste em rotacionar, mover, expandir entre outras transformações, as imagens com a finalidade de criar novos dados de treinamento de forma simples e rápida. Afim de não alterar as características da serie temporal em si, a nossa abordagem foi criar novos dados realizando um deslocamento horizontal baseada na posição do valor máximo do sinal em um intervalo entre 10% a 90% da janela total do sinal. Cada sinal é amostrado de forma aleatória, com reposição, um deslocamento é realizado e armazenado em uma nova base de dados. Outro fator importante é a remoção do viés de centralidade presente no sinal original, como essa técnica desloca o sinal ao longo do tempo esse tipo de viés é removido. A fig.53 mostra um exemplo de sinal gerado pelo método. Para cada íon foram reservados entre sinais originais e sinais gerados um total de 41000 amostras, sempre adquiridos de forma aleatória.





4.4 TREINAMENTO REDE NEURAL

Nesta sessão será apresentada a arquitetura base utilizada neste trabalho bem como suas variações propostas.

4.4.1 Motivação

Antes de abordarmos as arquiteturas de redes neurais propostas, é importante elencar as motivações para uma abordagem de redes neurais profundas na tarefa de classificação de eventos de partículas ionizantes. A primeira delas é a extração de parâmetros dos dados de forma automática, através do aprendizado das informações que alimentam o modelo de rede neural (MAO; JAIN, 1995). Outro ponto é a capacidade de encontrar e relacionar as características do dado em relação ao tempo.Por conta da natureza transitória dos dados coletados no experimento, é importante ter a capacidade de identificar quando certa característica se manifesta

Fonte: Autor

ao longo do sinal, e utilizar esse tipo de informação para enriquecer as informações que são levadas em consideração no momento de classificar o sinal, sendo assim, a adoção de células LSTM, trará ao modelo final a capacidade de avaliar a informação considerando sua natureza temporal. Um terceiro ponto a ser avaliado é a característica que redes neurais apresentam em aperfeiçoar a sua capacidade de aprendizado e generalização com o aumento do dado de treino disponível, o que torna essa abordagem promissora ao passo que o volume de informação dessa natureza tende a aumentar conforme outros experimentos como esses são desenvolvidos. Por fim, a flexibilidade e capacidade de transferência de conhecimento presentes em redes neurais, permite que o trabalho aqui desenvolvido seja aprimorado e expandido, utilizando os modelos de redes neurais aqui gerados paracriar outras abordagens além da classificação, como filtros de sinais (ZHU; MOUSAVI; BEROZA, 2019) e modelos de geração e simulação de novos dados como o SEE(KUO; LIN; HU, 2020).

4.4.2 Arquitetura da rede neural

A aplicação de redes neurais profundas tem papel em diversas áreas e em diferentes tarefas. No nosso cenário, será utilizado no contexto de classificação de series temporais. A primeira abordagem adotada foi utilizar a arquitetura proposta em (KARIM et al., 2017) que propõe uma extensão da arquitetura conhecida como FCN (WANG; YAN; OATES, 2017), adicionando camadas LSTM, formando assim a arquitetura LSTM-FCN, e sua variação com o recurso de *attention*, formando assim ALSTM-FCN. A adoção destas duas novas arquiteturas aumentou a performance de classificação no conjunto de dados UCR(DAU et al., 2019), que são utilizados como referência para medir a qualidade da classificação de series temporais, em relação a marcos anteriores para o mesmo problema. A adoção desta técnica para o nosso problema não foi satisfatório, devido problemas encontrados durante as etapas de treinamento da rede neural proposta e pelos resultados parciais obtidos após a execução de muitas *epochs* não se aproximaram do modelo de base proposto para comparação. Por esses motivos a sua implementação foi abortada precocemente.

A segunda abordagem, e a adotada nesse trabalho, leva em consideração a arquitetura proposta pelo grupo de pesquisa Netherlands eScience center (NLeSC), e utiliza uma ferramenta que ajuda na busca dos melhores hiperparâmetros para essas arquiteturas nomeado de Mcfly(KUPPEVELT et al., 2020). Esta ferramenta permite a busca sistemática de uma rede neural, baseada em duas arquiteturas CNN e DeepConvLSTM, através da geração de diversas redes neurais com diferentes topologias a fim de encontrar a melhor combinação dos parâmetros da rede. O modelos gerados são treinados em uma pequena porção dos dados (aproximadamente 10% do conjunto de treino disponível)é amostrada de forma aleatória e sua performance é avaliada nesse cenário, escolhendo-se aquele que obter o melhor resultado.

A arquitetura DeepConvLSMT foi a que obteve melhores resultados no conjunto de dados preliminares, sendo assim a arquitetura escolhida. Dada a arquitetura escolhida, uma nova rodada de testes foi realizada para otimizar hiperparâmetros específicos a essa abordagem. São eles: Número de camada convolucionais (N), número de camadas LSTM (M), número de filtros em cada camada convolucional (#ConvFiltros) e o número de células da camada LSTM (#CélulasLSTM).

4.4.3 Arquiteturas propostas

Neste trabalho iremos estender a utilização da arquitetura DeepConvLSTM, propondo algumas alterações. A primeira delas é a utilização de GRU ao invés de células LSTM. A adoção dessas células já mostraram ganho de eficiência nos mesmos cenários que a LSTM e menores tempos de treinamento pois realiza menos operações matemáticas (CHUNG et al., 2014). Essa nova proposta recebe o nome de DeepConvGRU.

A segunda derivação da arquitetura é a adoção de camadas bidirecionais tanto com células LSTM quanto com células GRU, dando origem assim a duas novas arquiteturas a DeepConv-BiLSTM e DeepConvBiGRU. A adoção de camadas bidirecionais obteve ganhos de resultado em cenários de processamento de linguagem natural (GRAVES; JAITLY; MOHAMED, 2013b).

4.4.4 Treinamento do Modelo

O processo de treinamento da arquitetura obtida foi realizada utilizando-se os dados originais em conjunto com os dados gerados de forma artificial. Foram amostradas 5000 exemplos de cada classe, de forma aleatória, sendo reservado 80% para treinamento (da porção de treinamento foram reservados 10% para validação) e 20% para testes. A primeira etapa de treinamento, realizada apenas com a arquitetura DeepConvLSTM, foi definida para serem executada em 100 *epochs*. O valor de *loss* do conjunto de validação era monitorado ao fim de cada *epoch* com a finalidade de evitar estagnação do processo de treinamento, avaliando se havia melhora nas métricas do modelo, caso não houvesse progresso a taxa de aprendizado do modelo

era reduzida para corrigir esse problema (BENGIO, 2012). A rede neural final obtida pode ser vista em fig.54.



Figura 54 - Rede neural final da arquitetura DeepConvLSTM

Para fins de comparação e linha de base, um modelo SVM foi treinado com a mesma base de dados, e os resultados obtidos foram comparados com os resultados da rede Deep-ConvLSTM. Para o processo de comparação das 4 arquiteturas presentes nesse trabalho, foram realizadas 10 rodadas de treinamento para cada arquitetura, com inicialização de pesos baseados no número da rodada de treinamento, evitando assim o viés de inicialização dos pesos, e foram executadas 300 *epochs* em cada rodada. As características de distribuição do conjunto de dados e os métodos de controle da performance do modelo são iguais as descritas anteriormente.

4.5 INTERPRETABILIDADE

A questão de interpretabilidade de modelos é um campo importante do aprendizado de máquina, principalmente em modelos considerados "caixa preta", como é o caso das redes neurais, pois é importante conhecer os motivos pelos quais o modelo está obtendo os resultados. Uma área proeminente na questão de interpretação dos dados é a de processamento de imagens, cria-se mapas de calor mostrando as regiões mais importantes para o modelo realizar a classificação correta de um objeto(CRUZ-ROA et al., 2013). Para entendermos os resultados obtidos, desenvolvemos um simples algoritmo que mede a influência de cada ponto da serie temporal no processo de classificação, baseado em métodos de interpretabilidade em imagens. Nomeamos esse algoritmo de janelas deslizantes TS (*Time Series*). O algoritmo está descrito em algoritmo 2.

O algoritmo usa filtros de diversos tamanhos de comprimento para omitir partes, de forma sequencial, do sinal original. Com o sinal omitido, é medido o ganho ou perda de certeza na classificação do sinal correto, caso a omissão de uma região promova o ganho na probabi-

- 1 Entrada: Amostra a ser avaliada: X
- 2 Entrada: Classe de X: y_{verdade}
- 3 Entrada: modelo M
- 4 Entrada: lista com tamanho dos kernels: K
- 5 Calcula o número de elementos de X: X_n ;
- 6 Calcula o número de kernels de K: K_n ;
- 7 Inicializa matriz: M_{K_n,X_n} ;
- 8 Inicializa lista: X_{inf} ;
- 9 Obtenha o valor de $y_{predito}$ utilizando X através de M;
- 10 se $y_{predito} \neq y_{verdade}$ então
- retorna Erro 11
- 12 fim

```
13 Calcula a probabilidade y_{prob} utilizando X através de M;
```

- 14 para cada variável $k \in K$ faça
- Calcula o número de partições de X: $P = \frac{X_n}{k}$; 15
- para cada variável par $\in P$ faça 16
- Obtenha X_{temp} anulando partição par de X; 17
- Calcula a probabilidade $y_{probTemp}$ utilizando X_{temp} através de M; 18
- Calcule $\Delta_{k,par} = y_{prob} y_{probTemp}$; 19
- Armazene o resultado em $M_{K_n,X_n} = \Delta_{k,p}$; 20
- fim 21
- 22 fim
- para cada *variável* $p \in X$ faça 23
- $inf_{temp} = \sum_{k=1}^{K} M_{k,p};$ $X_{inf}^{p} = inf_{temp};$ 24
- 25
- 26 fim

```
27 retorna X_{inf}
```

lidade de classificação indica que a região não é relevante para a classificação, caso ocorra o oposto a região é tida como mais importante. Por se tratar de vários filtros de tamanhos diferentes mede-se a influência temporal da região em diversos instantes de tempo do sinal. No final da excursão de todos os filtros ao longo de todo o sinal, as influências de cada ponto são ponderadas, e aquelas regiões com os maiores valores de influência são destacadas como pontos relevantes e os demais como pontos não relevantes.

5 RESULTADOS

Neste Capítulo serão expostos os resultados obtidos em simulação, bem como as analises referentes aos dados obtidos nos experimentos realizados em campo.

5.1 SIMULAÇÕES

Nas sessões seguintes serão demonstrados os resultados obtidos durante as fases de simulação, começando pela criação da estrutura passando pelo sistema de polarização e a simulação da partícula ionizante.

5.1.1 Ajuste da estrutura

A primeira fase deste trabalho teve como foco a criação de uma estrutura que simulasse as características elétricas do dispositivo 3N163. Para tanto, foram extraídas curvas $I_D x V_G$ para valores de $V_D = -0.1V$ (Figura 55) e $V_D = -1.0V V_D = -0.1V$ (Figura 56) de seis amostras do transistor em questão, para serem utilizados como referências para o dispositivo simulado. Estas curvas foram obtidas variando-se o valor de V_G em passos de 25mV até chegar ao valor de 5V. Figura 55 – Gráfico de diversos $I_D x V_G$ para um valor de V_D =-0,1 V.



Fonte: Autor

Figura 56 – Gráfico de diversos $I_D x V_G$ para um valor de V_D =-1,0 V.



Fonte: Autor

Com este conjunto de curvas, extraiu-se uma curva média e os valores de desvio padrão. Esta curva média (figura 57) foi usada como referência para os ajustes da estrutura de simulação, pois apesar de ser o mesmo modelo de transistor, cada um apresenta ligeiras diferenças de corrente para um mesmo valor de polarização, em grande parte por pertencerem a lotes diferentes de fabricação entre outras pequenas diferenças de processo que acabam refletindo no resultado final da produção.

Figura 57 – Gráfico da média de $I_D x V_G$ para um valor de V_D =-0,1V.



Fonte: Autor

A criação da estrutura levou em consideração informações extraídas de estimativas (CARVAJAL et al., 2009).

Com estes dados, as primeiras simulações foram realizadas, porém os primeiros resultados obtidos divergiam muito dos valores esperados, pois era necessário a realização de diversos ajustes na estrutura, alguns exemplos são:

 AJUSTE DA GRADE: A primeira grade adotada, era pouco detalhada na região de interesse (canais do transistor). Por se tratar de um dispositivo *multi finger*¹ o detalhamento desta região exigia um cuidado especial pois não se trata de uma região de canal contínua, necessitando a criação de grades individuais para cada canal.

¹Esta configuração utiliza da soma corrente de diversos transistores idênticos para a formação de um único dispositivo.

- CORRENTE DE FUGA: Um dos problemas que mais afetaram as simulações foi a corrente de fuga presente no dispositivo simulado. Após minuciosa inspeção no modelo, observou-se que havia uma região que apresentava uma alta taxa de recombinação de portadores. Devido ao layout dos terminais metálicos, notou-se que em determinados trechos dos contatos de dreno e fonte, que não participam da polarização do dispositivo, estavam influenciando diretamente no processo de condução, pois mesmo com tensão de porta igual a 0V havia condução de um valor considerável de corrente. A solução foi, assim como no dispositivo real, isolar estas regiões do restante do volume ativo de silício.
- TEMPO DE SIMULAÇÃO: Uma outra barreira que interferiu significantemente no processo de simulação, foi o tempo necessário para a realização de uma única simulação de uma curva IdxVg, por exemplo, que gastava algo em torno de 20 horas, o que tornava o processo de ajustes moroso e ineficiente. Diversas tentativas foram realizadas em busca de uma otimização deste processo, desde da criação de uma grade mais enxuta a utilização de um número maior de *cores* da CPU, porém sem resultados significativos. A solução que mostrou-se mais eficiente, foi alterar alguns parâmetros nos métodos de solução numérica do simulador, forçando as CPUS a realizarem mais operações para entregarem os resultados de convergência com menos interações possíveis. Esta mudança mostrou-se de grande valor pois o número de *cores* utilizados reduziu de 20 para 8 e o tempo de simulação de 20 para cerca de 3 horas.
- MODELO FÍSICO: Neste ponto, a determinação correta do modelo físico é fundamental para o sucesso da simulação. Após uma série de experimentações e testes o modelo eleito foi o *Phumob*, pois apresentou melhores resultados para o trabalho proposto e mostrou-se responsivo na questão de ajustes finos em seus parâmetros de configuração da simulação.

Com os ajustes dos parâmetros acima, iniciou-se uma rodada de refinamentos dos valores obtidos, com a finalidade de se aproximar os resultados decorrentes da estrutura simulada com os valores obtidos pela medição. Os resultados intermediários podem ser observados na figura 58 e o resultado final na figura 59.

Figura 58 – Gráfico $I_D x V_G$ mostrando uma série de ajustes realizados nos parâmetros do modelo físico para um valor de V_D =-0,1V.



Fonte: Autor

Figura 59 – Gráfico comparando os valores $I_D x V_G$ de referência com o real para um valor de V_D =-1V .



Fonte: Autor

5.1.2 Simulação Mixed mode do circuito de polarização.

Para a realização da simulação da radiação, primeiramente foi necessário emular as mesmas condições de polarização que se encontrava em campo, com a finalidade de ter o cenário mais próximo da realidade possível, para isso o simulador Sentaurus conta com um modo de simulação nomeado de *Mixed mode*². O esquemático do circuito descrito pelo SPICE utilizado pode ser observado na figura 40 da sessão 4.1.2.

O circuito de polarização gerado proveu o correto funcionamento da estrutura simulada, gerando o mesmo nível de corrente medido no experimento real, que nesto caso, é uma corrente de dreno no patamar de 100μ A.

5.1.3 Simulação da emissão de partícula ionizante

Neste ponto do trabalho, iniciou-se a simulação da emissão da partícula sobre a estrutura desenvolvida. Para isso, quatro dos setes feixes disponíveis no primeiro experimento realizado no Pelletron foram selecionados. O critério de seleção utilizado levou em consideração a profundidade que cada feixe poderia alcançar no dispositivo, para possibilitar os efeitos em diferentes profundidades. A tabela 4 lista os feixes selecionados.

Tabela 4 – Energia do feixe, profundidade de penetração e LET no silício utilizados em simulação .

Íon	Energia (MeV)	Penetração no Si(µm)	LET no Si(MeV/mg/cm ²)
^{12}C	45,0	55,4	2,6
^{16}O	52,5	38,9	4,6
^{28}Si	66,0	21,6	12,8
35Cl	75,0	19,5	17,1

Fonte: Autor

Cada simulação considerou uma janela para o evento de 15ns. Para definir o momento em que o evento inicia a execução da simulação é dividida em duas etapas. A primeira etapa, que tem tamanho de 2ns, apresentam um passo de resolução máxima de 50ps e a partir desse ponto o valor máximo passa para 1ns. Os resultados de simulação para os íons de carbono (figura 60), oxigênio (figura 61), silício (figura 62) e cloro (figura 63) emitidos nas regiões de dreno e porta, podem ser observados nas figuras a seguir:

²Neste modo é possível utilizar circuitos descritos em SPICE em conjunto com a estrutura simulada, possibilitando a utilização de modelos de dispositivos padrão durante a simulação.





Fonte: Autor

Figura 61 – Valores da corrente I_D ao longo do tempo para a emissão de um íon de oxigênio nas regiões de porta e dreno do dispositivo 3N163.



Fonte: Autor
Figura 62 – Valores da corrente I_D ao longo do tempo para a emissão de um íon de silício nas regiões de porta e dreno do dispositivo 3N163.



Fonte: Autor

Figura 63 – Valores da corrente I_D ao longo do tempo para a emissão de um íon de cloro nas regiões de porta e dreno do dispositivo 3N163.



Fonte: Autor

Os resultados das simulações para emissão das partículas nas regiões de porta e dreno estão sumarizados nas figuras 64 e 65 respectivamente.





Fonte: Autor

Figura 65 – Resumo das correntes I_D 's ao longo do tempo para a um conjunto de íons na região de dreno do dispositivo 3N163



Fonte: Autor

5.2 MODELOS DE CLASSIFICAÇÃO

Nesta sessão serão demonstrados os resultados obtidos pelo modelo DeepConvLSTM versus o resultado do modelo SVM utilizado como referência da classificação. Na fig.66, ob-

servarmos o desempenho do modelo SVM para cada íon disponível no conjunto de dados. O resultado do modelo de referência foi relativamente bom, principalmente nas classes referentes aos íons de oxigênio(O) e cloro(Cl), contudo, apresenta dificuldades na desambiguação dos íons de cobre(Cu) e titânio(Ti).



Figura 66 – Matriz de confusão do resultado da classificação no conjunto de dados de teste para o modelo SVM

Fonte: Autor "adaptado de"Oliveira et al., 2020, p.146

Partindo para os resultados obtidos pelo modelo proposto conseguimos ver na fig.67 uma melhora na classificação em todos as classes, melhorando a questão da desambiguação na hora de definir a classe correta.



Figura 67 – Matriz de confusão do resultado da classificação no conjunto de dados de teste para o modelo DeepConvLSTM

Fonte: Autor "adaptado de"Oliveira et al., 2020, p.146

Por fim, a superioridade do modelo proposto também aparece utilizando-se outras métricas como mostrado na tab.5. É possível observar ganhos de performance em todos os pontos avaliados, mostrando a relevância da informação temporal e a representação em altas dimensões trazidas pelo modelo DeepConvLSTM.

Tabela 5 - Resumo dos resultados de cada modelo em diferentes métricas

Classes	SVM		DeepConvLSTM-NN			Amostras	
Classes	Precisão	Recall	F1-score	Precisão	Recall	F1-score	Allosuas
Cl	0,97	0,99	0,98	0,99	1,00	1,00	1002
Cu	0,82	0,82	0,82	0,98	0,95	0,96	975
F	0,99	1,00	0,99	1,00	1,00	1,00	1006
0	1,00	1,00	1,00	1,00	1,00	1,00	1041
Ag	0,93	0,90	0,91	0,96	0,92	0,94	1012
Si	0,94	0,94	0,94	0,96	0,99	0,97	974
Ti	0,75	0,75	0,75	0,91	0,93	0,92	990
Resumo	0.91	0.91	0.91	0.97	0.97	0.97	7000

Fonte: Autor "adaptado de"Oliveira et al., 2020, p.146

5.2.1 Modelo de classificação no cenário dos dados simulados

Uma ponte para validarmos os resultados simulados do SEE e comparar com os dados reais obtidos através dos experimentos é a utilização dos modelos de redes neurais. Pela natureza de aprendizado através do dado que as redes neurais apresentam, temos a vantagem de contar com características do sinal aprendidos de forma automática pelo modelo, que vão além dos parâmetros que seriam levados em consideração por uma analise estatística feita manualmente, e não enviesa a comparação entre o dado simulado e o real pois os dados gerados no processo de tratamento do conjunto de dados aumenta a capacidade de generalização do modelo eliminando algum possível viés. Na fig.68, é mostrado 3 dos 4 sinais simulados sendo que o elemento carbono ficou de fora por não fazer parte dos dados de treinamento dos modelos de redes neurais.



Figura 68 - Dados simulados usados para validação com os modelos de redes neurais

Fonte: Autor

É notável que existem semelhanças entre os dados simulados com os dados reais, porém o dado simulado apresenta uma característica diferente após o pico do sinal, aparentemente a recuperação do dado tem uma descida mais lenta e suave, enquanto na maioria das amostras dos dados reais, existe uma descida acelerada após o pico e também um vale mais acentuado em relação ao que a simulação apresenta. A tabela 6 apresenta todos os modelos de redes neurais desempenhando a predição da possível classe de cada sinal simulado. É notória a predominância de todos modelos em classificar os sinais como cloro, mesmo o modelo sendo totalmente balanceado em relação a todas as classes. A única classe que é classificada corretamente em relação aos dados simulados é a cloro, possivelmente a silhueta do sinal é a mais próxima dentre

as simuladas que esta dos dados reais. Adiante será explorada as questões das regiões de maior importância para a classificação do sinal simulado.

Modelo	Classe verdadeira	Classe Predita	Prob. classe verdadeira	Prob. classe predita
DeepConvLSTM	Sicílio	Cloro	0,01702	0,93007
DeepConvLSTM	Cloro	Cloro	0,93589	0,93589
DeepConvLSTM	Oxigênio	Cloro	0,00250	0,89982
DeepConvGRU	Sicílio	Cloro	0,00168	0,96578
DeepConvGRU	Cloro	Cloro	0,98207	0,98207
DeepConvGRU	Oxigênio	Cloro	0,00094	0,95707
DeepConvBiLSTM	Sicílio	Cloro	0,00463	0,96076
DeepConvBiLSTM	Cloro	Cloro	0,96076	0,97742
DeepConvBiLSTM	Oxigênio	Cloro	0,00135	0,95849
DeepConvBiGRU	Sicílio	Cloro	0,00435	0,97832
DeepConvBiGRU	Cloro	Cloro	0,98622	0,98622
DeepConvBiGRU	Oxigênio	Cloro	0,00095	0,96026

Tabela 6 – Predição dos dados simulados utilizando-se as quatro arquiteturas propostas

Fonte: Autor

5.3 COMPARAÇÃO DEEPCONVLSTM, DEEPCONVGRU, DEEPCONVBILSTM E DE-EPCONVBIGRU

As variações propostas aqui, terão dois critérios de comparação. Vale ressaltar que foram executadas 10 rodadas de cada arquitetura, levando-se em consideração 300 *epochs* de treinamento, e cada rodada com sua própria semente de gerador aleatório, usado nesse cenário para evitar o viés de inicialização dos pesos. O primeiro deles é o tempo de treinamento, que é um fator muito importante que pode fazer projetos se tornarem inviáveis e custosos e com o aumento do volume de dados e complexidade do problema, esse tempo de treinamento só tende a crescer. Observando a fig.69, é possível verificar o resultado esperado. As arquiteturas que continham células GRU em substituição das células LSTM, obtiveram um tempo em torno de 18% menor sem as estruturas bidirecionais, e com a arquitetura bidirecional por volta de 16% de redução no tempo. Outro fato relevante é o acréscimo de tempo nas arquiteturas bidirecionais, como essa arquitetura basicamente é uma rede espelhada que considera os dados do fim para o começo, o tempo de treinamento será maior pois o número de parâmetros a serem otimizados aumentou.



Figura 69 – Tempo de treinamento para 10 rodadas diferentes com 300 *epochs* em cada rodada para cada modelo

Observando fig.69 também podemos notar a questão da variabilidade do tempo de treinamento de cada rede neural. As arquiteturas com células GRU tem um desvio padrão relativamente menor, o que vai de encontro com o número reduzido de parâmetros em relação a sua contrapartida com células LSTM, que nos trás indícios que o processo de otimização nesse cenário é mais determinístico entre diferentes inicializações de pesos. Algumas rodadas apresentaram um resultado considerado como fora do comum (*outlier*), o que pode ser explicado pela inicialização da rede que pode facilitar ou dificultar o processo de automatização.

O segundo critério de avaliação é a acurácia do modelo no conjunto de dados de teste. Podemos observar na fig.70, que as diferenças das medianas das acurácias são relativamente baixas sendo percebidas nas primeiras casas decimais.

Fonte: Autor



Figura 70 – Tempo de treinamento de cada modelo

Fonte: Autor

Afim de realizar um avaliação estatística das diferenças de performance de cada modelo, adotamos o teste ANOVA que avalia se há diferenças estatisticamente significantes entre as distribuições em questão. A hipótese nula (H_0), enuncia que não há diferença entre a média das distribuições, e caso p - valor > 0,05 considera-se que não existe evidências suficientes para se rejeitar H_0 . Caso p - valor < 0,05 existe evidências que pelo menos uma das médias das distribuições é diferente das outras, nesse caso rejeita-se H_0 e aceita-se a hipótese alternativa (H_a). A tabela 7 mostras os resultados obtidos pelo teste ANOVA para as acurácias de cada modelo em relação ao modelo DeepConvLSTM.

Tabela 7 – Teste ANOVA das acurácias dos modelos DeepConvGRU, DeepConvBiLSTM e DeepConvBiGRU em relação a arquitetura DeepConvLSTM

Modelo	Estatística F	p-valor
DeepConvGRU	0,4600	0,5062
DeepConvBiLSTM	3,5233	0,0768
DeepConvBiGRU	8,6162	0,0088

Fonte: Autor

Segundo os resultados dos testes estatísticos, a única arquitetura que apresenta um acréscimo com relevância estatística é a DeepConvBiGRU, provando que a adoção das células GRU com o recurso bidirecional, pode melhorar as métricas dessa abordagem de redes neurais para a classificação de sinais de SEE. Outro ponto a se observar é a variância nos resultados de acurácia, as arquiteturas com células GRU apresentam um maior desvio padrão em relação as versões com células LSTM. Como as células GRU são uma versão simplificada das LSTM, esta conta com menos recursos para lidar com a informação ao longo do tempo, o que pode trazer mais disparidade entre os resultados em cada rodada. O desvio padrão entre as rodadas de uma mesma arquitetura se explica pelo fato das inicializações com sua própria semente, o que torna a busca do mínimo global da função de erro diferente em cada cenário, mostrando como uma simples inicialização da rede neural pode interferir o resultado final.

5.4 INTERPRETABILIDADE DAS REDES NEURAIS

Por fim, temos os resultados obtidos pelo algoritmo de interpretação desenvolvido nesse trabalho, nomeado de Janela deslizante TS. O algoritmo utiliza o modelo treinado e usa as predições em uma das amostras para analisar a influencia que cada ponto do sinal apresenta para o modelo realizar a classificação de forma correta da amostra. As fig.71 a 74 mostram o resultado obtido após a execução do algoritmo de interpretabilidade.





Fonte: Autor





Fonte: Autor





Fonte: Autor





Fonte: Autor

Analisando os resultados obtidos, podemos observar que grande parte do transitório, que representa a colisão da partícula ionizante no dispositivo, é considerada como uma das principais regiões que contribuem com a classificação correta do sinal. A região que precede o pico do sinal, quase em sua totalidade é considerada irrelevante para a classificação do sinal, apenas uma pequena região na ponta é considerada importante, mas pode ser efeito to tamanho de kernel selecionado para a execução do algoritmo de interpretação. Logo após do pico do sinal, o primeiro vale é considerado irrelevante para a classificação do sinal, seguido do segundo pico que também é considerado importante. Observando os pares fig.71 e 72, fig.73 e 74, mostram que as arquiteturas que contam com as células GRU consideram as regiões ruidosas após o pico do sinal como relevantes enquanto as arquiteturas com LSTM é mais restritivo com essas regiões.

5.4.1 Interpretabilidade dos resultados para os sinais simulados

Como parte do trabalho, fizemos interface dos sinais simulados com os sinais obtidos em experimento através dos modelos de redes neurais. Como explorado na sessão 5.2.1, vimos que apenas o resultado da simulação com íon de cloro foi corretamente classificado. O resultado da interpretação pode ser visto em fig.75.







As regiões de importância estão concentradas majoritariamente próximo ao valor de pico, as demais regiões são pouco relevantes, o que mostra que os sinais obtidos no experimento compartilham poucas características entre si. É necessário refinar a simulação para aumentar a fidelidade do sinal simulado com o sinal real. Outro fator que pode dificultar a classificação é a presença do ruído inerente ao processo de aquisição do sinal, o que pode inserir informação de difícil simulação e classificação.

6 CONCLUSÕES

O primeiro ponto a ser avaliado é o ajuste da estrutura do modelo 3D simulado do dispositivo P-MOS 3N163. Após a avaliação dos resultados da comparação entre as curvas $I_D x V_g$ presentes na fig.59, podemos observar que os ajustes realizados na estrutura do modelo de simulação obteve um resultado satisfatório quando comparados com a curva média obtida de dispositivos reais, habilitando a estrutura para a próxima etapa referente à simulação da emissão de partículas ionizantes, emulando as condições presentes durante o experimento, afim de se observar o fenômeno SEE.

Os resultados das simulações das emissões de partículas, apresentaram respostas com o formato e valores de máxima próximos aos que se observa nos dados reais medidos, para as emissões realizadas na região de dreno do dispositivo, contudo, quando a emissão era realizada na região de porta do dispositivo, a resposta obtida em simulação apresentava um comportamento diferente do esperado, formando um pequeno vale no momento do evento tendo a corrente em regime do dispositivo estabilizada em pouco tempo, ao invés de um pico momentâneo produto da coleta dos pares elétron-lacuna gerados pelas passagem do íon pelo dispositivo. Até o momento de escrita deste texto, comportamento semelhante não foi medido em campo, deixando esse comportamento em aberto e passível de estudo mais aprofundado.

Em relação à fidelidade do evento simulado e medido, recorreu-se à utilização do modelo de aprendizado de máquina e ao algoritmo de interpretação dos resultados da rede neural, afim de avaliar a existência de características em comum entre o sinal simulado e os sinais reais atestando a precisão do sinal simulado. Observando os resultados da tabela 6, apenas o sinal simulado de cloro obteve uma classificação correta, porém o que não garante que o sinal foi realmente classificado de forma correta já que a classe predita de forma predominante é a cloro, mesmo o modelo contando com um conjunto de dados balanceado. Observando a fig.68 temos uma pequena região considerada como influente para a classificação do sinal simulado de cloro, o que mostra a baixa fidelidade do sinal simulado com o sinal real, pois a rede neural não foi capaz de encontrar características nesse sinal que indiquem a sua classe. Dado esse cenário é possível afirmar que, a etapa de polarização e as características elétricas do modelo do dispositivo 3N163 estão ajustadas de forma correta, cabendo um aprimoramento da sessão de simulação do evento SEE.

A proposta de se utilizar uma abordagem de aprendizado profundo para a classificação do sinais medidos em campo, mostrou-se acertada quando comparada com a abordagem de um modelo clássico de classificação (SVM), pois os resultados presentes na tabela 5, mostram que o modelo de rede neural obteve resultados superiores em todas as métricas avaliadas, mesmo o modelo clássico já apresentando um boa performance. A adoção desta técnica de aprendizado profundo, abre novos horizontes para o estudo dos efeitos de SEE, pois o parâmetros aprendidos pelas camadas das redes neurais trazem novas informações em relação a anatomia do evento, uma vez que se aprendeu quais são os pontos importantes para a classificação do sinal, é possível utilizar essas informações como ponto de atenção no desenvolvimento de técnicas para mitigar a ação desse efeito em dispositivos semicondutores e aplicações reais. Outro ponto de atenção e a possibilidade de transferência de conhecimento que pode realizar através da reutilização do modelo já treinado, mudando sua arquitetura pode-se expandir sua utilização para filtragem de sinais e até mesmo geração de sinais de SEE artificiais.

Em relação às arquiteturas derivadas da DeepConvLSTM aqui propostas, conseguimos observar através da fig.70 e do resultado do teste estatístico presente na tabela 7, que houve sim um ganho na questão da acurácia quando se utiliza a estrutura aqui batizada de DeepConvBi-Gru, elevando a capacidade de classificação dos sinais, contudo, vale observar que existe um acréscimo no tempo de treinamento, o que pode tornar o modelo impeditivo em alguns cenários.

Por fim, temos o algoritmo proposta para interpretabilidade dos resultados obtidos pelas redes neurais, útil para avaliar a influência que cada região do sinal apresenta no processo da correta classificação do sinal. É possível observar nas fig.72 a 74, a influência de cada região do sinal. A região próxima ao valor máximo da onde é mostrada como relevante, sendo esse resultado positivo pois é nessa região que o fenômeno se manifesta e mostrando o aprendizado dos modelos de redes neurais. Em contrapartida, a região logo após ao valor máximo do sinal tem menor influência ou até valores negativos, assim como a maior parte do sinal que apresenta ruídos, o que pode ser interpretado como características inseridas pelo sistema de medição e efeitos de segunda ordem, pois se são considerados pouco influentes há indícios então que são características pouco discriminantes entre uma classe e outra, sendo assim são características compartilhadas e presentes em todos os sinais, logo são efeitos inseridos pelo sistema, que é o ponto em comum entre um experimento e outro.

Em resumo, esta abordagem híbrida no estudo dos fenômenos SEE, se mostrou de grande proveito pois uma abordagem alimenta e aprimora a outra, nesse caso o modelo de classificação ajuda no aperfeiçoamento da frente de simulação pois é possível realizar uma interface entre os dados simulados e os dados reais via modelo de classificação. A utilização de técnicas de aprendizado de máquina abrem novas possibilidades no processo de investigação de

fenômenos causados por partícula ionizante, o que ajuda na criação de sistemas mais robustos ao SEE.

REFERÊNCIAS

AGUIAR, V. A P et al. Experimental setup for Single Event Effects at the S??o Paulo 8UD Pelletron Accelerator. **Nuclear Instruments and Methods in Physics Research, Section B: Beam Interactions with Materials and Atoms**, v. 332, p. 397–400, ago. 2014. ISSN 0168583X. DOI: 10.1016/j.nimb.2014.02.105. Disponível em: http://linkinghub.elsevier.com/retrieve/pii/S0168583X14003620>.

AGUIAR, V.A.P et al. New Setup for SEE Measurements in South America. **RADECS 2017**, 2017.

ALVES, A. D C et al. Mapping ion beam induced current changes in a commercial MOSFET. **Nuclear Instruments and Methods in Physics Research, Section B: Beam Interactions with Materials and Atoms**, Elsevier B.V., v. 269, n. 20, p. 2355–2359, 2011. ISSN 0168583X. DOI: 10.1016/j.nimb.2011.02.073. Disponível em: http://dx.doi.org/10.1016/j.nimb.2011.02.073.

ARUTT, Charles N et al. Proton Irradiation as a Screen for Displacement- Damage Sensitivity in Bipolar Junction Transistors. v. 62, n. 6, p. 2498–2504, 2015.

BENGIO, Yoshua. Practical recommendations for gradient-based training of deep architectures. In: NEURAL networks: Tricks of the trade. [S.l.]: Springer, 2012. P. 437–478.

BENGIO, Yoshua; FRASCONI, Paolo; SIMARD, Patrice. The problem of learning long-term dependencies in recurrent networks. In: IEEE. IEEE international conference on neural networks. [S.l.: s.n.], 1993. P. 1183–1188.

BREIMAN, Leo. Bagging predictors (Technical Report 421). University of California, Berkeley, 1994.

CARUANA, Rich; LAWRENCE, Steve; GILES, C Lee. Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. In: ADVANCES in neural information processing systems. [S.1.: s.n.], 2001. P. 402–408.

CARVAJAL, M. A. et al. Simulated and experimental angular response of a commercial MOSFET used as dosimeter. **Proceedings of the 2009 Spanish Conference on Electron Devices, CDE'09**, v. 00, n. 1, p. 184–187, 2009. DOI: 10.1109/SCED.2009.4800461.

CHO, Kyunghyun et al. On the properties of neural machine translation: Encoder-decoder approaches. **arXiv preprint arXiv:1409.1259**, 2014.

CHOO, SEOK CHEOW. Theory of a Forward-Biased Diffused-Junction P-L-N Rectifier- Part I: Exact Numerical Solutions. **IEEE Transactions on Electron Devices**, n. 8, p. 51–66, 1972.

CHUNG, Junyoung et al. Empirical evaluation of gated recurrent neural networks on sequence modeling. **arXiv preprint arXiv:1412.3555**, 2014.

COLINGE, Jean-Pierre; COLINGE, Cynthia A. **Physics of semiconductor devices**. 2nd. [S.l.]: Kluwer Academic Publisher, 2003. ISBN 1402070187.

CRUZ-ROA, Angel Alfonso et al. A deep learning architecture for image representation, visual interpretability and automated basal-cell carcinoma cancer detection. In: SPRINGER. INTERNATIONAL Conference on Medical Image Computing and Computer-Assisted Intervention. [S.l.: s.n.], 2013. P. 403–410.

DAU, Hoang Anh et al. The UCR time series archive. **IEEE/CAA Journal of Automatica Sinica**, IEEE, v. 6, n. 6, p. 1293–1305, 2019.

DE NOBEL, D. PHASE EQUILIBRIA AND SEMICONDUCTING PROPERTIES OF CADMIUM TELLURIDE *). **R 380 Reprinted from Philips Res. Repts**, v. 14, p. 361–399, 1959.

DIRAC, Paul Adrien Maurice. On the theory of quantum mechanics. **Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character**, The Royal Society London, v. 112, n. 762, p. 661–677, 1926.

DONOHO, David L. Aide-Memoire. High-Dimensional Data Analysis: The Curses and Blessings of Dimensionality. American Math. Society Lecture-Math Challenges of the 21st Century, 2000. DOI: 10.1.1.329.3392.

DRUCKER, Harris et al. Boosting and other ensemble methods. **Neural Computation**, MIT Press, v. 6, n. 6, p. 1289–1301, 1994.

DUZELLIER, Sophie. Radiation effects on electronic devices in space. Aerospace Science and Technology, Elsevier Masson, v. 9, n. 1, p. 93–99, 2005. ISSN 12709638. DOI: 10.1016/j.ast.2004.08.006.

FACCIO, Federico et al. TID and displacement damage effects in vertical and lateral power MOSFETs for integrated DC-DC converters. In: IEEE. 2009 European Conference on Radiation and Its Effects on Components and Systems. [S.l.: s.n.], 2009. P. 46–53.

FISHER, Ronald A. The use of multiple measurements in taxonomic problems. **Annals of eugenics**, Wiley Online Library, v. 7, n. 2, p. 179–188, 1936.

GÉRON, Aurélien. Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. first. [S.l.]: O'Reilly, 2017.

GERS, Felix A; SCHMIDHUBER, Jürgen; CUMMINS, Fred. Learning to forget: Continual prediction with LSTM. IET, 1999.

GLENN F. KNOLL. **Radiation Detection and Measurement**. [S.l.: s.n.], 2014. P. 1–5. ISBN 9780874216561. DOI: 10.1007/s13398-014-0173-7.2. arXiv: arXiv:1011.1669v3.

GLOROT, Xavier; BORDES, Antoine; BENGIO, Yoshua. Deep sparse rectifier neural networks. In: PROCEEDINGS of the fourteenth international conference on artificial intelligence and statistics. [S.l.: s.n.], 2011. P. 315–323.

GOLIN, Stuart. The electronic band structure of arsenic. II. Self-consistent approach. **Physical Review**, APS, v. 140, 3A, a993, 1965.

GOODFELLOW, Ian J; SHLENS, Jonathon; SZEGEDY, Christian. Explaining and harnessing adversarial examples. **arXiv preprint arXiv:1412.6572**, 2014.

GOODFELLOW, Ian et al. Deep learning. [S.l.]: MIT press Cambridge, 2016. v. 1.

GRAVES, Alex. Generating sequences with recurrent neural networks. **arXiv preprint arXiv:1308.0850**, 2013.

GRAVES, Alex; JAITLY, Navdeep; MOHAMED, Abdel-rahman. Hybrid speech recognition with deep bidirectional LSTM. In: IEEE. 2013 IEEE workshop on automatic speech recognition and understanding. [S.l.: s.n.], 2013. P. 273–278.

GRAVES, Alex; JAITLY, Navdeep; MOHAMED, Abdel-rahman. Hybrid speech recognition with deep bidirectional LSTM. In: IEEE. 2013 IEEE workshop on automatic speech recognition and understanding. [S.l.: s.n.], 2013. P. 273–278.

GROSSO, G.; PARRAVICINI, G.P. **Solid State Physics**. [S.l.]: Elsevier Science, 2000. ISBN 9780123044600. Disponível em: https://books.google.fr/books?id=-f7kenVZXlQC>.

HINNEBURG, Alexander; KEIM, Daniel A. Optimal grid-clustering: Towards breaking the curse of dimensionality in high-dimensional clustering. In:

HOCHREITER, Sepp. The vanishing gradient problem during learning recurrent neural nets and problem solutions. **International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems**, World Scientific, v. 6, n. 02, p. 107–116, 1998.

HOCHREITER, Sepp; SCHMIDHUBER, Jürgen. Long short-term memory. Neural computation, MIT Press, v. 9, n. 8, p. 1735–1780, 1997.

HOERL, Arthur E; KENNARD, Robert W. Ridge regression: Biased estimation for nonorthogonal problems. **Technometrics**, Taylor & Francis Group, v. 12, n. 1, p. 55–67, 1970.

JAITLY, Navdeep; HINTON, Geoffrey E. Vocal tract length perturbation (VTLP) improves speech recognition. In: PROC. ICML Workshop on Deep Learning for Audio, Speech and Language. [S.l.: s.n.], 2013. v. 117.

JOHNSTON, a. Radiation Damage of Electronic and Optoelectronic Devices in Space. **Presented at 4th International Workshop on Radiation Effects on Semiconductor Devices for Space Application**, p. 1–9, 2001. Disponível em: <message:%7B%5Ctextless%7DD3CD75E93F6C6Fhatsui@spring8.or.jp%7B% 5Ctextgreater%7D\$%5Cbackslash\$npapers2://publication/uuid/AE8AB6AA-5C97-49B8-BB1B-6E24CD088BD0>.

JOHNSTON, Allan. **Reliability and Radiation Effects in Compound Semiconductors**. [S.l.: s.n.], 2010. P. 376. ISBN 978-981-4467-65-0. DOI: 10.1142/9789814277112.

KALCHBRENNER, Nal; GREFENSTETTE, Edward; BLUNSOM, Phil. A convolutional neural network for modelling sentences. **arXiv preprint arXiv:1404.2188**, 2014.

KARIM, Fazle et al. LSTM fully convolutional networks for time series classification. **IEEE** access, IEEE, v. 6, p. 1662–1669, 2017.

KIM, Kwang In; JUNG, Keechul; KIM, Hang Joon. Face recognition using kernel principal component analysis. **IEEE Signal Processing Letters**, v. 9, n. 2, p. 40–42, fev. 2002. ISSN 1070-9908. DOI: 10.1109/97.991133.

KIROS, Ryan; SALAKHUTDINOV, Ruslan; ZEMEL, Richard S. Unifying visual-semantic embeddings with multimodal neural language models. **arXiv preprint arXiv:1411.2539**, 2014.

KLAASSEN, D B M. A unified mobility model for device simulation-I. Model equations and concentration dependence. **Solid State Electronics**, v. 35, n. 7, p. 953–959, 1992. ISSN 00381101. DOI: 10.1016/0038-1101(92)90325-7.

KUBETSKY, L. A. Multiple amplifier. **Proceedings of the Institute of Radio Engineers**, v. 25, n. 4, p. 421–433, 1937.

KUO, Ping-Huan; LIN, Ssu-Ting; HU, Jun. DNAE-GAN: Noise-free acoustic signal generator by integrating autoencoder and generative adversarial network. **International Journal of Distributed Sensor Networks**, SAGE Publications Sage UK: London, England, v. 16, n. 5, p. 1550147720923529, 2020.

KUPPEVELT, D van et al. Mcfly: Automated deep learning on time series. **SoftwareX**, Elsevier, v. 12, p. 100548, 2020.

LANF-USP. Acelerador Pelletron | Departamento de Física Nuclear. 2016. Disponível em: http://portal.if.usp.br/fnc/node/389>. Acesso em: 14 set. 2016.

LANG, Kevin J; WAIBEL, Alex H; HINTON, Geoffrey E. A time-delay neural network architecture for isolated word recognition. **Neural networks**, Elsevier, v. 3, n. 1, p. 23–43, 1990.

LAWRENCE, Steve et al. Face recognition: A convolutional neural-network approach. **IEEE** transactions on neural networks, IEEE, v. 8, n. 1, p. 98–113, 1997.

LECUN, Yann et al. Backpropagation applied to handwritten zip code recognition. **Neural computation**, MIT Press, v. 1, n. 4, p. 541–551, 1989.

LINEAR INTEGRATED SYSTEMS. 3n163, 3n164. [S.l.], 2016.

LOSKA, Krzysztof; WIECHUŁA, Danuta. Application of principal component analysis for the estimation of source of heavy metal contamination in surface sediments from the Rybnik Reservoir. **Chemosphere**, v. 51, n. 8, p. 723–733, 2003. ISSN 0045-6535. DOI: https://doi.org/10.1016/S0045-6535(03)00187-5. Disponível em: http://www.sciencedirect.com/science/article/pii/S0045653503001875>.

LUKE, P N. Unipolar Charge Sensing with Coplanar Electrodes- Application to Semiconductor Detectors. **IEEE TRANSACTIONS ON NUCLEAR SCIENCE**, v. 42, n. 4, 1995.

MAAS, Andrew L; HANNUN, Awni Y; NG, Andrew Y. Rectifier nonlinearities improve neural network acoustic models. In: 1. PROC. icml. [S.l.: s.n.], 2013. v. 30, p. 3.

MAKOWSKI, Dariusz. **the Impact of Radiation on Electronic Devices With the Special Consideration of Neutron and Gamma Radiation Monitoring Wpływ Promieniowania Na Pracę Pomiaru Promieniowania**. 2006. F. 151. Tese (Doutorado) – Technical University of Łódź Department.

MAO, Jianchang; JAIN, Anil K. Artificial neural networks for feature extraction and multivariate data projection. **IEEE transactions on neural networks**, IEEE, v. 6, n. 2, p. 296–317, 1995.

MASETTI, G; SEVERI, M; SOLMI, S. Modeling of Carrier Mobility Against Carrier Concentration in Arsenic-Doped, Phosphorus-Doped, and Boron-Doped Silicon. **IEEE Transactions on Electron Devices**, v. 30, n. 7, p. 764–769, 1983. ISSN 0018-9383. DOI: 10.1109/T-ED.1983.21207.

MASSENGILL, Lloyd W; TUINENGA, Paul W. Single-event transient pulse propagation in digital CMOS. **IEEE Transactions on nuclear science**, IEEE, v. 55, n. 6, p. 2861–2871, 2008.

MCCULLOCH, Warren S; PITTS, Walter. A logical calculus of the ideas immanent in nervous activity. **The bulletin of mathematical biophysics**, Springer, v. 5, n. 4, p. 115–133, 1943.

MCGREGOR, D.S.; HERMON, H. Room-temperature compound semiconductor radiation detectors. **Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment**, North-Holland, v. 395, n. 1, p. 101–124, ago. 1997. ISSN 01689002. DOI: 10.1016/S0168-9002(97)00620-7. Disponível em: http://linkinghub.elsevier.com/retrieve/pii/S0168900297006207.

MEDINA, N H et al. Brazilian facilities to study radiation effects in electronic devices. **2013 14th European Conference on Radiation and Its Effects on Components and Systems** (**RADECS**), p. 1–7, 2013. DOI: 10.1109/RADECS.2013.6937368. Disponível em: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6937368>.

MEDINA, N. H. et al. First successful SEE measurements with heavy ions in Brazil. **IEEE Radiation Effects Data Workshop**, 2015-Janua, January, p. 1–3, 2015. DOI: 10.1109/REDW.2014.7004571.

MELOROSE, J.; PERROY, R.; CAREAS, S. **RADIATION EFFECTS IN SEMICONDUCTORS**. [S.l.: s.n.], 2015. v. 1. ISBN 9788578110796. DOI: 10.1017/CBO9781107415324.004. arXiv: arXiv:1011.1669v3.

MILLER, G L; GIDSON, W M; DONOVAN, P F. Semiconductor PArticle Detectors. **Annual Review of Nuclear Science**, v. 12, n. 1, p. 189–220, 1962. DOI: 10.1146/annurev.ns.12.120162.001201. eprint:

http://dx.doi.org/10.1146/annurev.ns.12.120162.001201. Disponível em: http://dx.doi.org/10.1146/annurev.ns.12.120162.001201.

MINSKY, Marvin; PAPERT, Seymour. Perceptrons. MIT Press, 1969.

MITCHELL, Tom M. Machine Learning. [S.l.]: McGraw-Hill, 1997.

MOORE, B. Principal component analysis in linear systems: Controllability, observability, and model reduction. **IEEE Transactions on Automatic Control**, v. 26, n. 1, p. 17–32, fev. 1981. ISSN 0018-9286. DOI: 10.1109/TAC.1981.1102568.

NAKAJIMA, Hiroshi et al. Single event effect characterization of the mixed-signal ASIC developed for CCD camera in space use. **Nuclear Instruments and Methods in Physics Research, Section A: Accelerators, Spectrometers, Detectors and Associated Equipment**, Elsevier, v. 731, p. 166–171, 2013. ISSN 01689002. DOI: 10.1016/j.nima.2013.05.146. Disponível em: http://dx.doi.org/10.1016/j.nima.2013.05.146.

NEVILLE H. FLETCHER. The High Current Limit for Semiconductor Junction Devices. v. 45, p. 862–872, 1957.

NORMAND, Eugene. Single Event Upset at Ground Level.

OLDHAM, T.R.; MCLEAN, F.B. Total ionizing dose effects in MOS oxides and devices. **IEEE Transactions on Nuclear Science**, v. 50, n. 3, p. 483–499, jun. 2003. ISSN 0018-9499. DOI: 10.1109/TNS.2003.812927. Disponível em: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1208572>.

OLDHAM, Timothy R. Scaling and Single Event Effects (SEE) Sensitivity Scaling and Single Event Effects (SEE) Sensitivity. **Radiation Effects**, IEEE Nuclear e Space Radiation Effects Conference, California, 2003.

OLIVEIRA, Juliano Alves de et al. Electronic system for data acquisition to study radiation effects on operating MOSFET transistors. **AIP Conference Proceedings - RTFNB 2013**, v. 1265, p. 130–134, 2014. DOI: 10.1063/1.4901778. Disponível em: http://scitation.aip.org/content/aip/proceeding/aipcp/10.1063/1.4901778.

OLIVEIRA, Juliano Alves et al. Deep Neural Network for Single Event Effects Classification. Anais do XIII Encontro Acadêmico de Modelagem Computacional, p. 86–102, 2020.

OWENS, A. Series in Sensors: Compund Semiconductor Radiation Detectors. [S.l.: s.n.], 2012. P. 568. ISBN 9781439873137.

OWENS, Alan; PEACOCK, A. Compound semiconductor radiation detectors. **Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment**, v. 531, n. 1, p. 18–37, 2004. ISSN 01689002. DOI: 10.1016/j.nima.2004.05.071. PAULI, Wolfgang. Über den Zusammenhang des Abschlusses der Elektronengruppen im Atom mit der Komplexstruktur der Spektren. **Zeitschrift für Physik**, Springer, v. 31, n. 1, p. 765–783, 1925.

PEREZ, Luis; WANG, Jason. The effectiveness of data augmentation in image classification using deep learning. **arXiv preprint arXiv:1712.04621**, 2017.

RIEMER, Bernard W et al. SINGLE EVENT EFFECTS TEST FACILITY AT OAK RIDGE NATIONAL, p. 1–12, 2015.

ROSENBLATT, Frank. Perceptron simulation experiments. **Proceedings of the IRE**, IEEE, v. 48, n. 3, p. 301–309, 1960.

RUMELHART, David E; HINTON, Geoffrey E; WILLIAMS, Ronald J. Learning internal representations by error propagation. [S.1.], 1985.

RUTHERFORD, E.; GEIGERG, H. An Electrical Method of Counting the Number of α -PArticles from Radio-Active Substances. **Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences**, The Royal Society, v. 81, n. 546, p. 141–161, 1908. ISSN 0950-1207. DOI: 10.1098/rspa.1908.0065. eprint: http://rspa.royalsocietypublishing.org/content/81/546/141.full.pdf. Disponível em: http://rspa.royalsocietypublishing.org/content/81/546/141.

SALAHUDDIN, Sayeef; NI, Kai; DATTA, Suman. The era of hyper-scaling in electronics. **Nature Electronics**, v. 1, n. 8, p. 442–450, ago. 2018. ISSN 2520-1131. DOI: 10.1038/s41928-018-0117-x. Disponível em: <http://www.nature.com/articles/s41928-018-0117-x>.

SCHWANK, James R; SHANEYFELT, Marty R; DODD, Paul E. Radiation Hardness Assurance Testing of Microelectronic Devices and Integrated Circuits : Radiation Environments , Physical Mechanisms , and Foundations for Hardness Assurance. v. 60, n. 3, p. 1–59, 2008.

SEXTON, Fred W; MEMBER, Senior. Destructive Single-Event Effects in Semiconductor Devices and ICs. **IEEE TRANSACTIONS ON NUCLEAR SCIENCE**, v. 50, n. 3, p. 603–621, 2003.

SHANNON, Claude E. A mathematical theory of communication. **The Bell system technical journal**, Nokia Bell Labs, v. 27, n. 3, p. 379–423, 1948.

SILVEIRA, M A G et al. A Commercial off-the-shelf pMOS Transistor as X-ray and Heavy Ion Detector. **Journal of Physics: Conference Series**, v. 630, p. 012012, jul. 2015. ISSN 1742-6588. DOI: 10.1088/1742-6596/630/1/012012. Disponível em: <http://stacks.iop.org/1742-6596/630/i=1/a=012012?key=crossref.20adeec1fdc5f7f7298e6e9e9c1078f0>.

SRIVASTAVA, Nitish et al. Dropout: a simple way to prevent neural networks from overfitting. **The journal of machine learning research**, JMLR. org, v. 15, n. 1, p. 1929–1958, 2014.

SROUR, JR; MARSHALL, Cheryl J; MARSHALL, Paul W. Review of displacement damage effects in silicon devices. **IEEE Transactions on Nuclear Science**, IEEE, v. 50, n. 3, p. 653–670, 2003.

STATHAKIS, D. How many hidden layers and nodes? **International Journal of Remote Sensing**, Taylor Francis, v. 30, n. 8, p. 2133–2147, 2009. DOI: 10.1080/01431160802549278.

STREETMAN, Ben G; BANERJEE, Sanjay et al. **Solid state electronic devices**. [S.l.]: Prentice hall Englewood Cliffs, NJ, 1995. v. 4.

STURESSON TEC-QEC, F. Space Radiation and its Effects on EEE Components Single Event Effects (SEE) Mechanism and Effects. In:

SYNOPSYS. Sentaurus Device User Guide. [S.l.: s.n.], 2009. P. 2009.

SZE., S. M.; NG, Kwok K. **Physics of Semiconductor Devices**. 3rd. [S.l.]: A Jonh Wiley Sons, 2007. ISBN 0471143235.

TIBSHIRANI, Robert. Regression shrinkage and selection via the lasso. **Journal of the Royal Statistical Society: Series B (Methodological)**, Wiley Online Library, v. 58, n. 1, p. 267–288, 1996.

TSOI, Ah Chung; BACK, Andrew. Discrete time recurrent neural network architectures: A unifying review. **Neurocomputing**, Elsevier, v. 15, n. 3-4, p. 183–223, 1997.

TSOI, Ah Chung; BACK, Andrew D. Locally recurrent globally feedforward networks: a critical review of architectures. **IEEE Transactions on Neural Networks**, IEEE, v. 5, n. 2, p. 229–239, 1994.

VELLA, Eleonora et al. Unraveling exciton dynamics in amorphous silicon dioxide: Interpretation of the optical features from 8 to 11 eV. **Physical Review B**, APS, v. 83, n. 17, p. 174201, 2011.

WANG, Zhiguang; YAN, Weizhong; OATES, Tim. Time series classification from scratch with deep neural networks: A strong baseline. In: IEEE. 2017 International joint conference on neural networks (IJCNN). [S.l.: s.n.], 2017. P. 1578–1585.

WIDROW, Bernard; HOFF, Marcian E. Adaptive switching circuits. [S.1.], 1960.

WYTHOFF, Barry J. Backpropagation neural networks: a tutorial. **Chemometrics and Intelligent Laboratory Systems**, Elsevier, v. 18, n. 2, p. 115–155, 1993.

ZHENG, Q. et al. Total Ionizing Dose Influence on the Single-Event Upset Sensitivity of 130-nm PD SOI SRAMs. **IEEE Transactions on Nuclear Science**, v. 64, n. 7, p. 1897–1904, jul. 2017. ISSN 0018-9499. DOI: 10.1109/TNS.2017.2706287.

ZHOU, Yi-Tong; CHELLAPPA, Rama. Computation of optical flow using a neural network. In: ICNN. [S.l.: s.n.], 1988. P. 71–78.

ZHU, Weiqiang; MOUSAVI, S Mostafa; BEROZA, Gregory C. Seismic signal denoising and decomposition using deep neural networks. **IEEE Transactions on Geoscience and Remote Sensing**, IEEE, v. 57, n. 11, p. 9476–9488, 2019.

ZIEGLER, J. F.; BIERSACK, J. P.; ZIEGLER, M. D. **TRIM - Setup and Input**. [S.l.: s.n.], 2008. P. 8.2. ISBN 9780965420716 096542071X.

ZIEGLER, James F; ZIEGLER, M. D.; BIERSACK, J. P. SRIM - The stopping and range of ions in matter (2010). Nuclear Instruments and Methods in Physics Research, Section B: Beam Interactions with Materials and Atoms, v. 268, n. 11-12, p. 1818–1823, 2010. ISSN 0168583X. DOI: 10.1016/j.nimb.2010.02.091.

ZWORYKIN, V.K.; MORTON, G.A.; MALTER, L. The Secondary Emission Multiplier-A New Electronic Device. **Proceedings of the IRE**, v. 24, n. 3, p. 351–375, 1936. ISSN 0096-8390. DOI: 10.1109/JRPROC.1936.226435. Disponível em: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1686095>.

APÊNDICE A – ARTIGO - XIII ENCONTRO ACADÊMICO DE MODELAGEM COMPUTACIONAL



Deep Neural Network for Single Event Effects Classification.

Juliano Alves de Oliveira¹, Marco Antônio Assis de Mello¹, Marcilei Aparecida Guazzelli da Silveira¹, Rudolf Theoderich Buhler¹, Vitor Ângelo Paulino de Aguiar², Nilberto Heder Medina² e Renato Camargo Giacomini¹

¹ Centro Universitário da FEI, São Bernardo do Campo/SP, Brazil
² Instituto de Física da USP, São Paulo/SP, Brazil

Abstract

This work proposes the use of deep neural networks, specifically a DeepConvLSTM network, for the classification of single effect events (SEE) that occur in electronic devices. Also proposed are three architectural variations derived from the original proposal, DeepConvGRU, DeepConvBiLSTM and DeepConvBiGRU, with the purpose of improving the system classification capacity. Finally, a simple neural network interpretability algorithm is proposed to observe the importance of each signal feature to better understand the phenomenon and possible generations of insights into the electrical characteristics of the phenomenon.

Keywords: Deep Neural Networks, Single Evente Effects, LSTM, GRU, classification

1 INTRODUCTION

The study of ionizing particles is relevant in several areas of knowledge such as aerospace, military, medicine, radio-protection, among others, because these applications can suffer from the effects of ionizing radiation on their electronic circuits (20; 14; 22). The detection and classification of particles have also a notable application in high energy physics research, which presents in its experiments a large data volume. Such data must be processed in real time, in order to define whether it should be stored or not, for further analysis. The smart detectors use classification algorithms, which identify known particle patterns and discards them, thus avoiding unnecessary storage of data (8). There are several approaches and circuits used in particle detection, some are based on CMOS pixel sensors (CPS) that rely on circuits and sensor elements on the same substrate (33). Other applications involve Monolithic Active Pixel Sensors (MAPS) which are a set of sensors

Contato: Juliano Alves de Oliveira, oliveira.juliano@outlook.com

forming a binary activation grid, in this way, for each collision that reaches a certain area, a set of pixels are disturbed causing its activation, and signaling that there was activity in that region (29). These examples of detectors are based on a set of sensors that allows great event coverage and have good spatial discrimination of the particle. In this work, we use a single sensor element, a commercial off-the-shelf transistor (COTS) PMOS 3N163. Although a set of sensor elements has a broader response to the event, the convenience of having a single sensor element with a single signal, in response to the event, was chosen because it allows further signal processing without the need to synchronize the multiple answers that would be obtained in other configurations.

Single Event Effect (SEE) is a short-lived phenomena that occur when an ionizing particle collides with a semiconductor device, creating in its path a cloud of electron-hole pairs, which are assimilated by the electric current flowing in the target device. It raises an anomaly in its operation. Within the various sub-classifications of the SEE, the Single Event Transient (SET) is characterized as the one that generates a momentary current peak in the device (15; 34). This type of event will be the object of study of this work. As its name indicates, the SET has a transient nature. In this way, this phenomenon presents temporal dependence, unlike a Single Event Upset (SEU), for example, that occurs in digital devices and can be interpreted as a logical inversion in the device (memory or register). Because of this transitory nature of SET (see fig. 10), a classification system that as input such data must be able to deal with this temporal variable. Traditional statistical methods present difficulties in the classification of problems with temporal dependence (30), but there are several advances in recent years in machine learning area, leveraged mainly by an increase in computational performance and access to large volumes of data. The strategy of bringing all these advances for the field of particle classification and detection was adopted in this work. One of the techniques that has been showing great gain in recent research on several non trivial problems such as image processing is the use of deep neural networks (DNN). It allows the discovery and extraction of characteristics of the input signals, and with each layer added, these characteristics are stacked and combined, creating new characteristics, increasing the information diversity and bringing new ways for better classifiers. It also presents the advantage of allowing the use of raw data with the need of little pre-processing. The different configurations of the DNN allow expressive gains depending on the addressed problem. For this particular case two DNN architectures based on (35) working together were adopted. The first one is known as Convolutional Neural Network (CNN), that allows the extraction of characteristics in a spatial way through filters that are obtained during the learning process. Such filters run through the entire input vector searching for the pattern to which each filter was designed to detect. The second configuration that works in conjunction with CNN is known as Long Short Term Memory (LSTM) which are cells able to learn characteristics and relate them over time, and unlike Recurrent Neural Networks (RNN) that present problems with information that presents a long dependence in time, the LSTM's are able to deal with this problem. They are frequently used in the processing of texts and sound, which are the main cases that present temporal relation (37). The adoption of machine learning techniques is already widely used in the field of high particle physics (HEP) in several fields. For example, the use of Generative Adversarial Networks (GANs) in the simulation of high energy phenomena. With a computational cost lower than traditional simulation



methods it furnishers closer results (19). Other successful applications include, but are not limited to: hit reconstruction (1), particle tracking finding in individual detector system (13), multiple particles detection using information provided by multiple systems (4; 26; 3), and also using machine learning on initial trigger system (27) and early Higgs boson searches and discovery (26) e (2).

This paper explores the application of machine learning techniques for the classification of electrical signals generated from the SEE, as well as the use of the algorithm to analyze second order effects, present in the measured signals.

Section 2 better describes the DNN used architecture, pre-processing of data, the experimental setup used to obtain the data, as well as its operability. Section 3 will discuss the obtained results. Lastly, in section 4, conclusions and future work will be exposed.

2 Methods and techniques

In this section, the materials and methods used during the experiments, as well as the methods adopted in the event acquisition process, will be described.

2.1 Experimental setup

The Pelletron 8UD accelerator is an electrostatic accelerator of the Tandem category, which can achieve energies above 100 MeV for experiments that require the use of heavier ions. This category of accelerator uses a mechanism where an ion with negative charge is accelerated by a positive electric field until a region where the charge is inverted, being accelerated again after this inversion.

The possible configurations allow, in addition as well as to experiments inside the vacuum spreading chamber, irradiation in equipment outside this environment, since some targets are difficult to be used in vacuum. Using a window with a thin sheet of aluminum or mylar, it is possible to route the ions from within the chamber to the device in question. In this configuration, the window can be positioned at 15 or 45 degree scattering angles (Figure 1), but limited to external proton beams up to ${}^{48}Ti$, due to high energy loss in the air and in the windows about to leave.



Fig. 1: Schematic of experimental setup used in radiations.

At the date of the experiment, the available ions in the Laboratório Aberto de Física Nuclear (LAFN - Open Laboratory of Nuclear Physics) are: ${}^{1}H$, ${}^{6,7}Li$, ${}^{10,11}B$, ${}^{12,13}C$,

 $^{16,17,18}O$, ^{19}F , $^{28,29,30}Si$, $^{35,37}Cl$, ^{48}Ti , $^{63,65}Cu$ and $^{107,109}Ag$. In figure 2 it is possible to consult the LET values used in the function experiments of the silicon penetration depth.



Fig. 2: Heavy ions used in experiments. LET as function of the depth in the silicon.

Table 1 discriminates the ions used in the study in question, which include ${}^{12}C$, ${}^{16}O$, ${}^{19}F$, ${}^{28}Si$, ${}^{35}Cl$, ${}^{63}Cu$ and ${}^{107}Ag$. These ions were scattered at 15 degrees through a gold foil of 275 μ g.cm⁻², which provides LETs of 2 to 45 MeV/mg/cm².

Ion	Energy (MeV)	Depth in $Si(\mu m)$	LET in $Si(MeV/mg/cm^2)$
^{12}C	45.0	55.4	2.6
^{16}O	52.5	38.9	4.6
^{19}F	42.0	23.8	6.8
^{28}Si	66.0	21.6	12.8
^{35}Cl	75.0	19.5	17.1
^{63}Cu	82.0	14.1	30.7
$\overline{^{107}Ag}$	97.5	13.1	39.8

Table 1: BEAM ENERGY, PENETRATION DEPTH AND LET IN SILICON.

A second experiment was carried out to obtain new data to evaluate predicted effects in simulation. In this second round the newest line present in the LAFN named SAFIIRA (See figure 3) was used.

This new line was built to meet the international standards that govern ionizing radiation studies. Some of these requirements, such as low ion flow, high coverage area, high uniformity, sample handling and various combinations of ions and energies (5). In this second round of the experiment a single ion beam of O was used using energy of 63MeV. Table 2 contains information about the beam used in the second round.

Table 2: BEAM ENERGY, PENETRATION DEPTH AND LET IN SILICON.

Ion	Energy (MeV)	Depth in $Si(\mu m)$	LET in $Si(MeV/mg/cm^2)$
^{16}O	63	50.4	4.6





Fig. 3: Overview of the new line SAFIIRA.

2.2 Device under test (DUT)

The PMOS 3N163 transistor was used. The use of this type of device is due to several benefits shown in the study of ionizing radiations, the ease of acquiring the device, biasing and manipulation of the component. Its metal encapsulation is easy to remove thus facilitating the exposure of the silicon to the particles. Previous studies with such component available, showing the responsiveness of this component to events involving radiation, as a good dosimeter (12) and also as SEE detector (32).

In all setups the device package is removed, in order to expose the sensitive regions of the device directly to the irradiated beams. In figure 4 it is possible to observe the decapsulated component.



Fig. 4: Detail of the decapsulated device.

The first configuration used in the tests adopted bias of a polarization 0.13V between drain and source (VDS) and -4.5V between gate and source (VGS). The device operates in its linear region, making the response to the nearest event without distortion by the amplifying properties that could be imparted to the device due to its polarization. The drain current was amplified, and the current measured through a successive approximation 12-bit A/D converter preceded by a Sample-and-Hold circuit. Thus any change in current flowing through the device is recorded and associated with an effect caused by the emission of radiation in the device. Due to the low sampling rate in relation to the duration of a

SEE event, only 0.02% of the possible events were recorded (31), resulting in the need to create a new system to perform data collection. The second configuration implemented a series of improvements in the experiment. The first one was the polarization circuit (Figure 5), which gave the experiment the ability to polarize the device under test in several operating regions. The sampling rate has increased significantly with the use of a Rohde & Schwarz model 1-GHz RTO1012 oscilloscope, which provides the sampling capability of 10G samples/s, which is equipped with a real-time digital trigger feature. This monitoring system proved to be of great value because of the memory economy it made possible, since it only recorded the events in which the voltage values exceeded a threshold, rejecting possible spurious signs.



Fig. 5: Schematic of the 3N163 transistor bias system.

The device under test was positioned inside the vacuum chamber (Figure 6), using a plate where the polarization circuit, connectors and supports for the transistor were located. The device was irradiated at 15 degrees, as shown in figure 7 (38).



Fig. 6: Device under test in the vacuum chamber.



Fig. 7: Schematic view of the arrangement used to perform the SEE tests.

During the radiation simulations, other waveforms were detected resulting from the action of the ionizing particles in the port region of the simulated 3N163 device, in this way a new round of experiments was necessary to obtain more data to confront the simulated results. For this new experiment, the same configuration of polarizing circuit described previously was used, as it proved to be a good apparatus to obtain the pulses from the radiation, but the arrangement of the plate in the camera occurred differently from the



previous one, in this new line the device was directly positioned in the direction of the beam, there was no scattering of the beam made by a gold foil, in this way it was possible to record a larger number of events during the experiment. The figure 8 contemplates the positioning of the device to be tested and all the polarization apparatus necessary for its operation.



Fig. 8: Disposition of the DUT in the camera used in the second experiment.

2.3 Data analysis

The data obtained during experiments in the Pelletron are of a transient nature, have a peak which records the moment when the electron-hole pairs generated during the traversal of the ionizing particle by the device are assimilated to the main current of the device. Each recorded pulse has 2000 points, spaced in $10\eta s$, which results in a $20\mu s$ event capture window. Approximately 123K samples were collected, being 7 different elements, as can be seen in figure 9, there are a number of different samples for each ion, this is due to several factors such as the fluency of the particle during the experiment.



Fig. 9: Percentage of each element in total of 123k samples. Each square represents approximately 1.5K samples.

The challenge of classifying these different ions is that, given a certain event for an element, different responses are obtained because the characteristics of the signal depend,

besides the type of ion and its energy, also depend on the angle of arrival and depth of penetration of the particle, which are parameters difficult to control and which make the classification task more complex. In figure 10 a sample of each ion is observed and in figure 11 the distribution of 100 ions of the same type.



Fig. 10: One sample for each ion.

Fig. 11: Multiple samples for the same ion.

A first problem that must be faced is the unbalance of classes, which ends up hampering the obtaining of a classification model (24), to deal with this problem was created a method of data augmentation, which consists of generating synthetic data based on the data real. The process randomly selects a sample and promotes a horizontal displacement, based on the peak value, in 10% to 90% of the total signal window. This technique allows the generation of more data for the classes with a reduced number of samples besides promoting greater diversity in the signal, since the acquisition method centralizes the peak at the moment of triggering, which eliminates the bias of always having the maximum centralized value. Figure 12 shows the generation of some samples by the date augmentation method. As generated 41k samples (including the originals) for training purpose.



Fig. 12: Data augmentation for same ion sample.



2.4 Neural Network Architecture

There are several jobs that use DNNs for time series classification. The first approach taken was to implement the proposed neural network in (25), in this work if the performance of the fully convolutional network (FCN)(41) architecture is extended by extending its architecture with the use of Long Short Term Recurrent Neural Networks (LSTM RNN) modules named new LSTM-FCN module, or the use of LSTM RNN with attention, called ALSTM-FCN, the use of these two new approaches (LSTM-FCN and ALSTM-FCN) improved, according to the work, the classification performance in the datasets used to benchmark this type of task, which are provided by the University of California Riverside (UCR)(9). The use of this architecture was not efficient in this work due to difficulties in the training phase, since the partial results obtained with many training epochs did not show great advances compared to the results obtained by the baseline model. The second approach was inspired by the architectures used by the Mcfly(40) framework of the Netherlands eScience center (NLeSC) research group, that systematically generates neural networks based on two base architectures, CNN and DeepConvLSTM. From these two base architectures, the framework performs a search for the best hyperparameters that optimize the ranking result. In a preliminary test, with a portion of the data, the DeepConvLSTM architecture was selected because it presented better performance in a representative sample of the data, thus only the hyperparameters for this architecture will be detailed. The following parameters are optimized by the framework: the number of Conv layers (N), number of LSTM Layers (M), number of filters for each Conv Layer (#ConvFilters) and the size of each hidden layer of each LSTM layer (#LSTMCells).

The first layer of the proposed model is batch normalization, the technique that is applied in this step has the purpose of normalizing the distribution of the data that is received, allowing a speed gain in neural network training because it avoids the internal covariate shift that occurs, mainly in the inner layers of the neural network, and allows to increase the learning rate used by the network (23). The batch normalization method occurs in three steps: The first one calculates the average of each data batch, then the batch variance is calculated and the data distribution is scalled and shifted. The algorithm 1 describes the actions performed:

Algorithm 1 Batch Normalizing Transform, applied to activation x over a mini-batch. Adapted from (23).

Require: Values of x over a mini-batch: $B = \{x_{1...m}\};$ **Ensure:** $y_i = BN_{\gamma,\beta}(x_i)$

 $\mu_B \leftarrow \frac{1}{m} \sum_{i=1}^m x_i \text{ \{mini-batch mean\}}$ $\sigma_B^2 \leftarrow \frac{1}{m} \sum_{i=1}^m (x_i - \mu_B)^2 \text{ \{mini-batch variance\}}$ $\hat{x}_i \leftarrow \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \text{ \{normalize\}}$ $y_i \leftarrow \gamma \hat{x}_i + \beta \equiv BN_{\gamma,\beta}(x_i) \text{ \{scale and shift\}}$

In the algorithm 1 γ and β are parameters learned during the neural network training step.

Moving on to the next layers of our neural network we come to the first major convolutional block of our architecture, which consists of a convolutional layer, a batch normalization layer and finally a ReLU (Retificated Linear Unit) activation layer. Convolutional networks are typically applied to image problems (2d) because they can learn patterns and look for them in various regions of an image, thus surpassing a fully connected conventional neural network (6). The power of convolutional neural networks lies in the sharing of trained weights, which decreases the number of weights to be trained and consequently makes training faster, its time sub-sampling and local receptive fields (which can be translated as the size of the convolution filters). The idea behind of a convolutional network is simple, the goal is to acquire a n number of filters, each filter being responsible for obtaining a characteristic of the input data, these characteristics, which are simple a priori, are combined in deeper layers into the network thus generating new more complex features and so on. The convolutional term comes from the fact that each filter traverses all the input data in search of this strained feature as a result of a map from which data pattern manifests itself in the data under analysis.

Another element that is part of the convolutional block is the ReLU which is responsible for transforming your entry or set or set of entries to a specific domain, which in this case is to zero entries that have negative values and replicate, in its output, positive values. This activation layer is defined in eq. 1, and is widely used because it reduces gradient fade problems, which occurs with sigmoid (16) activations, and has a reduced computational cost because it is just comparisons, summation, and multiplications.

$$f(x) = max(0, x) \tag{1}$$

After the N convolutional blocks are composed and the blocks containing the LSTM cells are introduced into the structure. This type of architecture has applications in a number of fields of study because of its data sequence processing capabilities (18; 36; 28).

The basic concepts behind of LSTM is memory cells and port units, that were originally introduced in (21). The heart of the LSTM operation is based on the state of the cell, which is changed as the data continues through the gates, each gate being responsive by performing an operation on the main data stream of the cell. The first gate is responsible for the acceptance or not of the data that were proposed by the network exit. This is known as the "forget gate layer" and has as a decision function a sigmoid(σ), which has output values between 0 and 1, being the value 1 indicative of keeping the data forward and 0 representing the total removal of the data. This gate is governed by Eq.2:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \tag{2}$$

Where:

- W_f = Weight matrix
- h_{t-1} = Previous output bias vector
- $x_t = \text{Input}$


• $b_f = \text{Bias vector}$

Continuing in the state of the cell, the data now passes through the gate known as the "input gate layer" which is responsible for determining what information is to be stored in the state of the cell. First, the sigmoid layer determines which values will be updated, and then a tanh layer generates a vector of possible new candidate values, \tilde{C}_t , that will be added to the state. There are two equations in this part of the cell Eq.3 and Eq.4 that work together to derive the result from this section:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{3}$$

$$\tilde{C} = \tanh(W_C[h_{t-1}, x_t] + b_C) \tag{4}$$

At this point, the old cell state C_{t-1} is updated to the current state C_t . The update process consists in multiplying the old state by f_t , such that the cell "forgets" the values previously stored from the decision, and finally add $i_t * \tilde{C}_t$, the result of these two operations presents the new candidates scaled by how much we decide to update each state value. In this way we have:

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{5}$$

Finally, the output value will be selected based on the state of the cell, which should still pass through a filtering process of the values. The first step of this filtering is to run a sigmoid function that delimits which parts of the cell state go to the output. This pre-selected value passes through a tanh function that will be responsible for mapping the cell state to a range between -1 and 1, such that only the selected values will be part of the final response. This layer is know as "selection gate layer". Eq.6 and Eq.7 refer to the process of filtering and selecting output values.

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o)$$
 (6)

$$h_t = o_t * \tanh(C_t) \tag{7}$$

The next element of the structure is known as Dropout. This cell is responsible for randomly eliminating a percentage of the elements contained in a given layer during the training phase. This type of procedure is used to avoid overfitting deep nets by generating variations of the same net, which during the validation stage benefits from this variation and obtaining a better average result (39).

The time distributed layer is used to assign each slice of the time series a layer, which in this case is a fully connected network.

The softmax layer applies Eq.8 to a vector of K real numbers, and normalizes them to a distribution probability that is proportional to the exponential value of the input numbers.

$$\sigma(Z)_i = \frac{e^{Z_i}}{\sum_{j=1}^K e^{Z_j}} \tag{8}$$

Finally, the Take Last layer is used to capture the last sequence response returned by the softmax layer as the final response.

Derived from this initial architecture, in this work, we propose some modifications in order to improve the model accuracy as well as the performance, so some modifications will be proposed. The first of these is the addition of Gated Recurrent Unit (GRU) cells, which have been proposed in (10) instead of LSTM cells, as they proved to be as efficient as LSTM in similar scenarios, but with better performance because they perform fewer mathematics operations within its (11). The equations governing the cell GRU are:

$$z_t = \sigma_g(W_z x_t + U_z h_{t-1} + b_z) \tag{9}$$

$$r_t = \sigma_g(W_r x_t + U_r h_{t-1} + b_r) \tag{10}$$

$$h_t = (1 - z_t) \cdot h_{t-1} + z_t \cdot \sigma_h(W_h x_t + U_h(r_t \cdot h_{t-1}) + b_h)$$
(11)

Where:

- $x_t = \text{input vector}$
- $h_t = \text{output vector}$
- $z_t = update gate vector$
- r_t = reset gate vector
- W, U and b = parameter matrices and vector
- σ_g = Activation function sigmoid
- σ_h = Activation function hyperbolic tangent

The second proposal is to use bidirectional layers with both LSTM and GRU cells, this type of approach creates the same sequence of cells in parallel with the original but in the opposite direction of data flow, with the purpose of using future data in predicting the next values. This type of architecture has obtained satisfactory results in Natural Language processing applications (17).

2.5 Training model

The training was performed with 80% of the total available data, and 10% of this portion was used to validate and monitor the evolution of the neural network. The remaining 20% was reserved for the model testing stage. An SVM classifier was obtained with the same data for the purpose of comparing the results obtained by the trained model. In total 100 epochs were performed in the training phase. To avoid stagnation of neural network training on a plateau, a loss monitoring function has been added to each epoch. This function evaluates whether there has been improvement in function loss by monitoring whether the loss of the neural network decreases, if there are no improvements the function reduces the learning rate of the neural network to the next epoch. This method allows the optimization function to be able to minimize the function to find the global minimum (7).



3 RESULTS AND DISCUSSION

In Fig. 13 the confusion matrix obtained by the SVM model in the test data is shown, showing the classification for each class.



Fig. 13: Confusion matrix for SVM baseline model on test data.



The result obtained in Fig. 13 shows the classification for the test base. Although it presents satisfactory results for some classes of ions, like Oxygen and Carbon, it presents many errors of classification between the Copper (radii 135pm) and Titanium (radii 140pm), both metals of transition, that by presenting atomic rays similar effects, making classification difficult in these cases.

In Fig. 14 the resulting confusion matrix obtained for the DeepConvLSTM network after the training process.

The results obtained by the LSTM network for the data set, indicated in the Fig. 14, present better results when compared with the base model (SVM). It is possible to notice that the disambiguation between the Titanium and Copper classes, and also among other classes, showing better performance in relation to the base model.

Classes		SVM		DeepConvLSTM-NN			Complex
	Precision	Recall	F1-Score	Precision	Recall	F1-Score	Samples
Cl	0.97	0.99	0.98	0.99	1.00	1.00	1002
Cu	0.82	0.82	0.82	0.98	0.95	0.96	975
F	0.99	1.00	0.99	1.00	1.00	1.00	1006
Ο	1.00	1.00	1.00	1.00	1.00	1.00	1041
Ag	0.93	0.90	0.91	0.96	0.92	0.94	1012
Si	0.94	0.94	0.94	0.96	0.99	0.97	974
Ti	0.75	0.75	0.75	0.91	0.93	0.92	990
Summary	0.91	0.91	0.91	0.97	0.97	0.97	7000

Table 3: SUMMARY OF RESULTS FOR EACH MODEL IN DIFFERENT METRICS

Other metrics (Precision, Recall and F1-Score) were used to evaluate, on a case-bycase basis, the performance between the two proposed models and are demonstrated in the Table 3. For the majority metrics used, the model obtained through the use of LSTM cells presented equal or higher results, bringing the importance of the temporal relation of the input data, because there were gains of prediction performance in the model that uses this relation.

3.1 Proposed models

The variations proposed in this work that can be named DeepConvGru, DeepConv-BiLSTM and DeepConvBiGRU were trained in 10 rounds with random weight initialization to avoid weight initialization bias, and each round featured 300 epochs. In fig. 15 shows the average training time of each model.

Looking at fig.15 it is possible to notice that the training times for architectures with GRU cells, both architectures with bidirectional structures present longer training time. Accuracies for each model were performed on the same test dataset and can be seen in fig.16.





Fig. 15: Training time for each model.

Fig. 16: Training time for each model.

Looking at the results obtained in fig.16, both GRU cell models are significantly better than their counterpart with LSTM cells, but these average differences are in the order of 10^{-2} . The table 4 shows the ANOVA test results comparing the results obtained by each model in relation to DeepConvLSTM.

Model	F-statistic	p-value
DeepConvGRU	0.4600	0.5062
DeepConvBiLSTM	3.5233	0.0768
DeepConvBiGRU	8.6162	0.0088

Table 4: Anova Test related to DeepConvLSTM Accuracy

Looking at the table 4 you can see that only the DeepConvBiGru model has a p-value less than 0.05, showing a significant improvement over the original model.

3.2 Model Interpretability

Model interpretability is important for understanding how the model makes decisions, and it is possible to gain insight into the problem being addressed. In Fig. 17 and Fig. 18 show how each model addresses a single signal.

In Fig. 17 and Fig. 18 show the most important regions for correct signal classification. There are no major differences between the regions of attention of the model, but the









BiLSTM feature importance. (b) DeepConvBiGRU feature importance. Fig. 18: Feature importance with bidirectional.

model with GRU cells consider the region after the signal peak as important, which can be explained by having less information retention mechanisms, but most of the noise of the signal. Signal is considered as regions not important for classification.

4 CONCLUSIONS

The proposed goal was successfully achieved, the neural network developed using LSTM cells provided better classification and disambiguation results between close classes compared to the previous model (SVM) used as baseline. The DeepConvGRU, DeepConv-BiLSTM and DeepConvBiGRU networks showed approximate results from the original model, but the DeepConvBiGRU model showed significant improvements in the statistical test with p-value less than 0.05, and despite having a longer training time, this does not affect its application final use. With the models created it was possible to define levels of relevance for each region of the curve, which allows to have new ideas about the analysis of the phenomenon and its mechanisms. The next steps involve migration of this model to embedded systems, allowing the creation of low cost particle detectors or systems capable of dealing with the anomalies imposed by the event.

ACKNOWLEDGEMENT

We are grateful to the CITAR Project, Centro Universitário da FEI and University of São Paulo for providing the facilities for the conduction of the experiments and data analysis. This publication is financially supported by Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

REFERENCES

- Aad, G., Abbott, B., Abdallah, J., Khalek, S. A., Abdinov, O., Aben, R., Abi, B., Abolins, M., and AbouZeid, e. a. (2014). "A neural network clustering algorithm for the ATLAS silicon pixel detector". *Journal of Instrumentation*, 9(9).
- [2] Aaltonen, T., Adelman, J., Álvarez González, B., Amerio, S., Amidei, D., Anastassov, A., Annovi, A., Antos, J., and Apollinari, e. a. (2010). "Search for the higgs boson using neural networks in events with missing energy and *b*-quark jets in $p\bar{p}$ collisions at $\sqrt{s} = 1.96$ TeV". *Phys. Rev. Lett.*, 104:141801.
- [3] Abazov, V. M., Abbott, B., Abolins, M., Acharya, B. S., Adams, M., Adams, T., Agelou, M., and Agram, e. a. (2005). "Measurement of $\sigma(p\overline{p} \rightarrow z) \cdot \text{Br}(z \rightarrow \tau \tau)$ at $\sqrt{s} = 1.96$ TeV". *Phys. Rev. D*, 71:072004.
- [4] Abramowicz, H., Caldwell, A., and Sinkus, R. (1995). "Neural network based electron identification in the zeus calorimeter". Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 365(2-3):508-517.
- [5] Aguiar, V., Medina, N., Added, N., Macchione, E., Aguirre, F., Ribas, R., Nascimento, S., Escudeiro, R., Allegro, P., Perego, C., Fagundes, L., Duarte, J. G., Scarduelli, V., Morais, O. B., Almeida, E. A., Joaquim, P. M., Souza, M., Cecotte, A., Martins, R., Brage, J., Leistenschneider, E., Oliveira, R., Assis, R., Leite, A., Terassi, J., Abreu, J., Simoes, R., Joaquim, A., Servelo, W. A., Silva, S. C., Minas, J., Silva, M., Silva, V., Kshinskiy, D., and Silveira, M. (2017). "New Setup for SEE Measurements in South America". *RADECS 2017.*
- [6] Bengio, Y. (1997). "Convolutional Networks for Images, Speech, and Time-Series Parsing View project MoDeep View project". Technical report n^o.
- [7] Bengio, Y. Practical recommendations for gradient-based training of deep architectures. title.
- [8] Casadei, D., Aracena, I., Banerjee, S., Beauchemin, P.-H., Calvet, S., Cranmer, K., Damazio, D., Djilkibaev, R., Eifert, T., Hillier, S., Idarraga, J., Johns, K., Kaushik, V., Konoplich, R., Kowalewski, R. V., Lei, X., Mann, A., Mermod, P., Mincer, A., Morel, J., Nemethy, P., Pinder, A., Taylor, R. P., Watson, A., and Zhao, L. (2011). "ATLAS NOTE The implementation of the ATLAS missing E T triggers for the initial LHC operation". Technical report n^o.
- [9] Chen, Y., Keogh, E., Hu, B., Begum, N., Bagnall, A., Mueen, A., and Batista, G. The UCR Time Series Classification Archive. title. www.cs.ucr.edu/~eamonn/time_series_data/.
- [10] Cho, K., Van Merriënboer, B., Bahdanau, D., and Bengio, Y. (2014). "On the properties of neural machine translation: Encoder-decoder approaches". arXiv preprint arXiv:1409.1259.
- [11] Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. (2014). "Empirical evaluation of gated recurrent neural networks on sequence modeling". arXiv preprint arXiv:1412.3555.
- [12] de Oliveira, J. A., de Melo, M. A. A., da Silveira, M. A. G., and Medina, N. H. (2014). "Electronic system for data acquisition to study radiation effects on operating MOSFET transistors". AIP Conference Proceedings - RTFNB 2013, 1265:130–134.
- [13] Denby, B. H. (1988). "Neural Networks and Cellular Automata in Experimental High-energy Physics". Comput. Phys. Commun., 49:429–448.
- [14] Ding, L., Gerardin, S., Bagatin, M., Bisello, D., Mattiazzo, S., and Paccagnella, A. (2016). "Radiation tolerance study of a commercial 65 nm CMOS technology for high energy physics applications". Nuclear Instruments and Methods in Physics Research, Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 831:265–268.
- [15] Edwards, R., Dyer, C., and Normand, E. (2004). "Technical standard for atmospheric radiation single event effects, (see) on avionics electronics". IEEE, pages 1–5.
- [16] Glorot, X., Bordes, A., and Bengio, Y. (2011). "Deep Sparse Rectifier Neural Networks". Technical report nº.
- [17] Graves, A., Jaitly, N., and Mohamed, A.-r. (2013). "Hybrid speech recognition with deep bidirectional lstm". IEEE, pages 273–278.
- [18] Graves, A., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H., and Schmidhuber, J. "A Novel Connectionist System for Unconstrained Handwriting Recognition".
- [19] Guest, D., Cranmer, K., and Whiteson, D. (2018). "Deep Learning and its Application to LHC Physics".
- [20] Gutiérrez, O., Prieto, M., Sánchez-Reyes, A., and Gómez, A. (2019). "TID characterization of COTS parts using radiotherapy linear accelerators". *IEICE Electronics Express*, 16(7):20190077–20190077.
- [21] Hochreiter, S. and Schmidhuber, J. (1997). "Lstm can solve hard long time lag problems". pages 473–479.
- [22] Hodson, R. F., Morgan, D., Ladbury, R. L., Chen, Y., Bay, M., and Zinchuk, J. (2019). "Radiation single event effects (see) impact on complex avionics architecture reliability".
- [23] Ioffe, S. and Szegedy, C. (2015). "Batch normalization: Accelerating deep network training by reducing internal covariate shift". In: 32nd International Conference on Machine Learning, ICML 2015, pages 448–456.
- [24] Iosifidis, V. and Ntoutsi, E. (2018). "Dealing with bias via data augmentation in supervised learning scenarios".
- [25] Karim, F., Majumdar, S., Darabi, H., and Chen, S. (2018). "Lstm fully convolutional networks for time series classification". IEEE Access, 6:1662–1669.
- [26] Khachatryan, V., Sirunyan, A. M., Tumasyan, A., Adam, W., Bergauer, T., Dragicevic, M., Erö, J., Fabjan, C., and Friedl, e. a. (2014). "Observation of the diphoton decay of the Higgs boson and measurement of its properties". *European Physical Journal C*, 74(10):1–49.
- [27] Kohne, J. K. et al. (1997). "Realization of a second level neural network trigger for the H1 experiment at HERA". Nucl. Instrum. Meth., A389:128–133.
- [28] Li, X. and Wu, X. (2015). "CONSTRUCTING LONG SHORT-TERM MEMORY BASED DEEP RECURRENT NEURAL NETWORKS FOR LARGE VOCABULARY SPEECH RECOGNITION". Technical report n^o.



- [29] Mager, M. (2016). "ALPIDE, the Monolithic Active Pixel Sensor for the ALICE ITS upgrade".
- [30] Makridakis, S., Spiliotis, E., and Assimakopoulos, V. (2018). "Statistical and Machine Learning forecasting methods: Concerns and ways forward". PLoS ONE, 13(3):1–26.
- [31] Medina, N. H., Silveira, M. A. G., Added, N., Aguiar, V. A. P., Aguirre, F., Giacomini, R., Macchione, E. L. A., de Melo, M. A. A., Oliveira, J. A., Santos, R. B. B., Seixas, L. E., and Tabacniks, M. H. (2013). "Brazilian facilities to study radiation effects in electronic devices". 2013 14th European Conference on Radiation and Its Effects on Components and Systems (RADECS), pages 1–7.
- [32] Medina, N. H., Silveira, M. A. G., Added, N., Aguiar, V. A. P., Giacomini, R., Macchione, E. L. A., De Melo, M. A. A., Santos, R. B. B., and Seixas, L. E. (2015). "First successful SEE measurements with heavy ions in Brazil". *IEEE Radiation Effects Data Workshop*, 2015-Janua(January):1–3.
- [33] Morel, F., Hu-Guo, C., Bertolone, G., Claus, G., Colledani, C., Dorokhov, A., Dozì, G., Dulinski, W., Fang, X., Goffe, M., Himmi, A., Jaaskelainen, K., Senyukov, S., Specht, M., Szelezniak, M., Pham, H., Valin, I., Wang, T., and Winter, M. (2013). "Recent citations MISTRAL & ASTRAL: two CMOS Pixel Sensor architectures suited to the Inner Tracking System of the ALICE experiment". pages 23–27.
- [34] Normand, E. (1996). "Single-event effects in avionics". IEEE Transactions on nuclear science, 43(2):461-474.
- [35] Ordóñez, F. J. and Roggen, D. (2016). "Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition". Sensors, 16(1).
- [36] Sak, H. H., Senior, A., and Google, B. "Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling". Technical report n^o.
- [37] Sepp, H. and J?urgen, S. (1997). "LONG SHORT-TERM MEMORY". Neural Computation, 9(8):1-32.
- [38] Silveira, M. A. G., Melo, M. A. A., Aguiar, V. A. P., Rallo, A., Santos, R. B. B., Medina, N. H., Added, N., Seixas, L. E., Leite, F. G., Cunha, F. G., Cirne, K. H., Giacomini, R., and de Oliveira, J. A. (2015). "A Commercial off-the-shelf pMOS Transistor as X-ray and Heavy Ion Detector". *Journal of Physics: Conference Series*, 630:012012.
- [39] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). "Dropout: a simple way to prevent neural networks from overfitting". The journal of machine learning research, 15(1):1929–1958.
- [40] van Kuppevelt, D., Meijer, C., van Hees, V., Bos, P., Spaaks, J., Kuzak, M., Huber, F., Hidding, J., and van der Ploeg, A.mcfly: deep learning for time series. title.
- [41] Wang, Z., Yan, W., and Oates, T. (2017). "Time series classification from scratch with deep neural networks: A strong baseline". pages 1578–1585.

APÊNDICE B – ARTIGO - 2018 IEEE 19TH LATIN-AMERICAN TEST SYMPOSIUM (LATS)

Single Event Effect: simulations and analysis on 3N163 PMOS transistor.

Juliano Oliveira*, Marcilei Aparecida Guazzelli[†], Marco Antônio Assis* and Renato Giacomini* *Electrical engineering department, Centro Universitário da FEI, São Bernardo do Campo,SP-BR [†]Physical department, Centro Universitário da FEI, São Bernardo do Campo,SP-BR

Abstract—This work addresses the simulation of a commercial p-channel MOSFET (3N163) using Sentaurus TCAD tool to observe the behavior of this device operating under heavyion environment, in order to study Single Event Effect (SEE) mechanisms and its effects. The simulated results were used to understand experimental data collected on field and make a comparison between real and simulated data. It also allowed interpretation of experimental data, as well as elimination of spurious noises and artifacts, which are not related to SEE effects, but are imposed by environment and experimental facilities.

Index Terms—SEE, Simulation, MOSFET, 3N163, Sentaurus.

I. INTRODUCTION

TOWADAYS electronics is present in many appli- cations, and consequently the electronic devices must be robust and able to work in harsh environments including the presence of ionizing radiation. Operating these devices in harsh environments, such as aerospace, nuclear reactors or particle accelerators [1], the ionizing radiation can modify the device behavior [2], which may provoke operational errors and bad functionality. These failurescan be partially attributed to ionizing particles present in this kind of environments. The Single Event Effect (SEE) is the object of study of this work. Single Event Effects (SEE) are short-duration events which are produced by a single ionizing particle on semiconductor devices. This event, which can be a current or voltage pulse, is caused by the ionized trace created by the charged particle along its path through the semiconductor. If the charge generated in the active region of the device, by the ionized particle, is large enough, it will cause a change in the normal operation of the circuit, producing transient effects that may change the device [3]. There are two main groups of SEE classification, destructive and non-destructive, and inside of each group we have Single Event Upsets (SEU)[4], Single Event Functional Interrupts (SEFI)[5] and Single Event Transients (SET) for non-destructive events and Single Event Latch-up (SEL)[6], Single Event Burnout (SEB) and Single Event Gate Rupture (SEGR)[7] for destructive events. In this work the type of event can be classified as SET, because of the characteristics of this event is a transient of voltage into the junction, the results depends of its biasing and the device still working normally after the event. The 3N163 p-channel mosfet transistor was chosen for this work because it has already been shown to be a good low- cost X-ray dosimeter[8], and it has potential for the detection of a SEE [9]. This device is largely used

in industry for several applications, presenting the following characteristics: very high input impedance, gate high breakdown, ultra-low leakage, fast switching and low capacitance. The simulation of the device and the ionizing particle are done through the Sentaurus tool, from Synopsys. Two TCAD tools from Synopsys were used, Sentaurus Structure Editor (SDE) and Sentaurus Device, where SDE is responsible for creating a structure containing the dot grids, the device's doping profiles and their respective electrical contacts, and the Sentaurus Device tool is a device stimulator able to work with 2D and 3D circuits, which provides the possibility to simulate a wide range of devices, in addition to providing multidimensional simulations, electro-thermal, mode-mixed devices and circuits. It incorporates advanced physical models and robust numerical methods for simulations [10].

II. EXPERIMENTAL SETUP

In order to evaluate and to extract information from the results obtained in the field experiments using several sources of ionizing radiation, such as carbon ions, chlorine and others [11], a simulation of the real device was created. For the creation of the model, microscope images of the device were evaluated to identify the dimensions of the component (Figure 1), other values of the dimensions of the device were based on common values of the technology used in this device [12], in this case the exact values adopted in the manufacturing process are difficult access.



Fig. 1. Top view of 3N163 structure.

978-1-5386-1472-3/18/\$31.00©2018 IEEE

This particular device has a geometric construction where the drain, port and source regions are successively intercalated, the device being formed by the combination of these subsystems. Figure 2a shows the final result obtained in the construction of the device. Technology parameters such as X_{OX}^{1} and doping profiles among other parameters which were initially adopted are described in [13]. With these values the first structure of the device was generated figure 2b.



Fig. 2. 3N163 structure simulation final results.

The first phase of this work focused on the creation of a structure that simulates the electrical characteristics of the 3N163 device, because without it, it would be impossible to continue the studies proposed here. Therefore, ID^2 versus VG^3 curves were extracted for values of $VD^4 = -0.1V$ and VD = -1.0V of six samples of the 3N163, to be used as references to the simulated device can be see in figure 3.



Fig. 3. Six IdxVg curves obtained from 3N163 samples .

The curves present in figure 4 were obtained by varying the value of VG in steps of 25mV until reaching the value of 5V. The standard deviation values were estimated from these simulated curves. This average curve (figure 4) was used as a reference for the simulation structure adjustments, because although they are the same model of transistor they presented slight differences of current for a same value of polarization, this is due to a large part because they belong to different batches of manufacturing among other small differences of process that end up reflecting in the final result of the production.

¹Oxide thickness.



Fig. 4. Mean IdxVg curve obtained from 6 3N163 samples .

In order to perform the radiation simulation, it was necessary to emulate the same biasing conditions that were present in the real experiment, in order to have the scenario as close to reality as possible, for which the Sentaurus simulator has a simulation mode named Mixed mode. The schematic of the circuit described by SPICE can be seen in the figure 5.



Fig. 5. Circuit used for polarization.

The experiment performed on Pelletron, previously mentioned on [11], made the tests using seven sources of radiation as follow:¹*H*, ^{6,7}*Li*, ^{10,11}*B*, ^{12,13}*C*, ^{16,17,18}*O*, ¹⁹*F*, ^{28,29,30}*Si*, ^{35,37}*Cl*, ⁴⁸*Ti*, ^{63,65}*Cu* e ^{107,109}*Ag*, and four of them were chosen for the simulation of the particle emission on the developed structure. The selection criterion took into account the depth that each beam could reach in the device, to allow the effects at different depths. Table I lists the selected beams, and theirs respective penetration on device structure and LET(Linear Energy Transference), these values were extracted using SRIM tool [14] that is able to simulate the particle emission and theirs characteristics given the material target, type of particle emitted and particle energy. These values are used as input parameters for the simulations. Each simulation considered a window for the 15*ns* event, that is, the simulation occurs in a time transient of this size.

TABLE I Beam energy, penetration depth and LET in silicon used in simulation.

Ion	Energy (MeV)	Penetration in Si(µm)	LET in Si(MeV/mg/cm ²)
^{12}C	45.0	55.4	2.6
^{16}O	52.5	38.9	4.6
^{28}Si	66.0	21.6	12.8
^{35}Cl	75.0	19.5	17.1

²Drain current.

³Gate voltage.

⁴Drain Voltage

III. RESULTS

The biasing process was a important step of this work. Simulate the same electrical characteristics is a key point to obtain good results during the ionization simulation process. The results can be seen on figure 6, which represents a satisfactory similarity between the reference curve and the simulated.



Fig. 6. Real IdxVg versus simulated IdxVg.

Two main regions were chosen to strike the particle on device, drain and gate. These particular points were chosen because the main contribution to the effect was observed in these places; source region region does not affect the final response. The results can be seen in figure 7 and 8.



Fig. 7. Ionizing particle on drain region.



Fig. 8. Ionizing particle on gate region.

Figure 7 shows the result for the emission of several ions with different energies in the drain region. The result of current obtained is within the expected, showing variations in intensity of current peak and duration of the event because we have ions with different characteristics for each curve presented. However, in the figure 8, which are the results obtained for the same ions of the results shown in the previous figure but emitted in the region of the gate, we notice a behaviour contrary to what is expected, in this case, the electron- lacuna do not cooperate with the addition of drain current, but the oxidative trapping loads change the polarization point of the device, thus changing the driving ability of the device momentarily.

IV. CONCLUSIONS

The data obtained through the simulation showed good agreement to experimental data. Looking at the results obtained in the simulation, we can conclude that during the field experiment, all the data collected came from an emission directly irradiated in the drain region, because the trigger of the acquisition system was not prepared for negative pulses, like those related to gate region strikes. This is an interpretation only possible with simulation work. Finally, other scenarios should be considered in new experiments in order to measure the events that occur in other regions of the device.

REFERENCES

- S.Duzellier, Radiation effects on electronic devices in space, 3rd ed. Harlow, England: Addison-Wesley, 1999.
- [2] R. T. Oldham, Scaling and Single Event Effects (SEE) Sensitivity Scaling and Single Event Effects (SEE) Sensitivity, IEEE Nuclear and Space Radiation Effects Conference, 2013.
- [3] A. Johnston, Radiation Damage of Electronic and Optoelectronic Devices in Space, Presented at 4th International Workshop on Radiation Effects on Semiconductor Devices for Space Application, p. 1–9, 2001.
- [4] N. Eugene, Single Event Upset at Ground, IEEE Transactions on Nuclear Science, p. 2742-2750, 1996.
- [5] W. R. Bernard, G. Franz, R. Oak, SINGLE EVENT EFFECTS TEST FACILITY AT OAK RIDGE NATIONA, p. 1-12, 2015.
- [6] H. Nakajima, Single event effect characterization of the mixed-signal ASIC developed for CCD camera in space use, Nuclear Instruments and Methods in Physics, p. 166–171, 2013.
- [7] F. W. Sexton, Destructive Single-Event Effects in Semiconductor Devices and ICs, IEEE TRANSACTIONS ON NUCLEAR SCIENCE, v. 50, n. 3, p. 603–621, 2003.
- [8] J. A. Oliveira, M. A. A. Melo, M. A. G. Silveira, N. H. Medina, Electronic system for data acquisition to study radiation effects on operating MOSFET transistors, AIP Conference Proceedings - RTFNB 2013,v. 1265, p. 130–134, 2014
- [9] N.H. Medina, M.A.G. Silveira, N. Added, V.A.P. Aguiar, R. Giacomini, Member, IEEE, E.L.A. Macchione, M.A.A. de Melo, R.B.B. Santos, Member, IEEE, and L.E. Seixas Jr., *First successful SEE measurements* with heavy ions in Brazil, IEEE Radiation Effects Data Workshop, 2015-January, January, p. 1–3, 2015.
- [10] SYNOPSYS. Sentaurus Device User Guide., [S.l.: s.n.], 2009. p. 2009.
- [11] Medina, N. H. and Silveira, M. A G and Added, N. and Aguiar, V. A P and Giacomini, R. and Macchione, E. L A and De Melo, M. A A and Santos, R. B B and Seixas, L. E., *First successful SEE measurements with heavy ions in Brazil*, IEEE Radiation Effects Data Workshop, pg 1-3, 2015
- [12] Asensio, L.J., Carvajal M.A., Lopez-Villanueva, J.A., Vilches, M., A.M. Lallena, A.M., Palma, A.J., *Evaluation of a low-cost commercial MOSFET as radiation dosimeter*, Sensors and Actuators A 125, p. 288-295, 2006.
- [13] L.J. Asensio, M.A. Carvajal, J.A. López-Villanueva, M. Vilches, A.M. Lallena, A.J. Palma, *Evaluation of a low-cost commercial MOSFET as radiation dosimeter*. Elsevier, Granada, v. 125, p. 288-295, 2005.
- [14] Ziegler, James F and Ziegler, M. D. and Biersack, J. P., SRIM The stopping and range of ions in matter, Nuclear Instruments and Methods in Physics Research, pg 1818-1823, 2010