

Controle de um conversor Boost para Rastreamento de Ponto de Máxima potência (MPPT) Aplicado em Painéis Fotovoltaicos

José Augusto Arbugeri - 13204572

Tulio Gomes Pereira - 14106069

Introdução

- Uso cada vez maior de tecnologia fotovoltaica para geração de energia elétrica
- A baixa eficiência das células fotovoltaicas obrigam a utilização de circuitos para alcançar o ponto de máxima potência dos painéis solares
- A potência fornecida pelas células solares dependem de variáveis independentes com irradiação e temperatura, o que dificulta o rastreamento do ponto de máxima potência.

Introdução

De forma geral, são dois os aspectos impactantes no funcionamento dos sistemas de rastreamento de máxima potência:

- Escolha do conversor cc-cc;
- Escolha do algoritmo de rastreamento.

Os conversores cc-cc podem ser entendidos como atuadores, que recebem o sinal de comando e trabalham de maneira a impor o ponto de máxima potência como ponto de operação do sistema.

As principais figuras de mérito que permitem quantificar e comparar a qualidade de diferentes métodos de rastreamento, são:

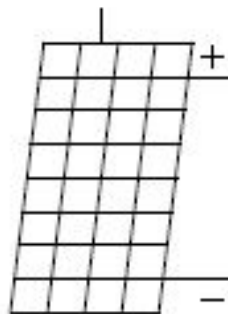
- Precisão;
- Rapidez da busca;
- Oscilações em regime permanente;
- Custo e simplicidade de implementação.

Na literatura existem diversas técnicas de rastreamento de máxima potência, contudo, as três mais comumente citadas são:

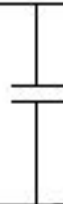
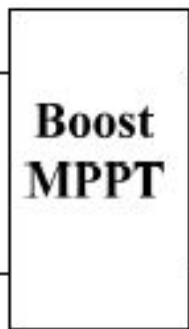
- Método da tensão constante;
- Método perturba e observa;
- Método condutância incremental.

Sistema de geração

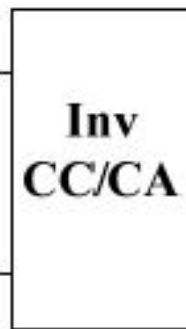
Arranjo Fotovoltáico



**Boost
MPPT**



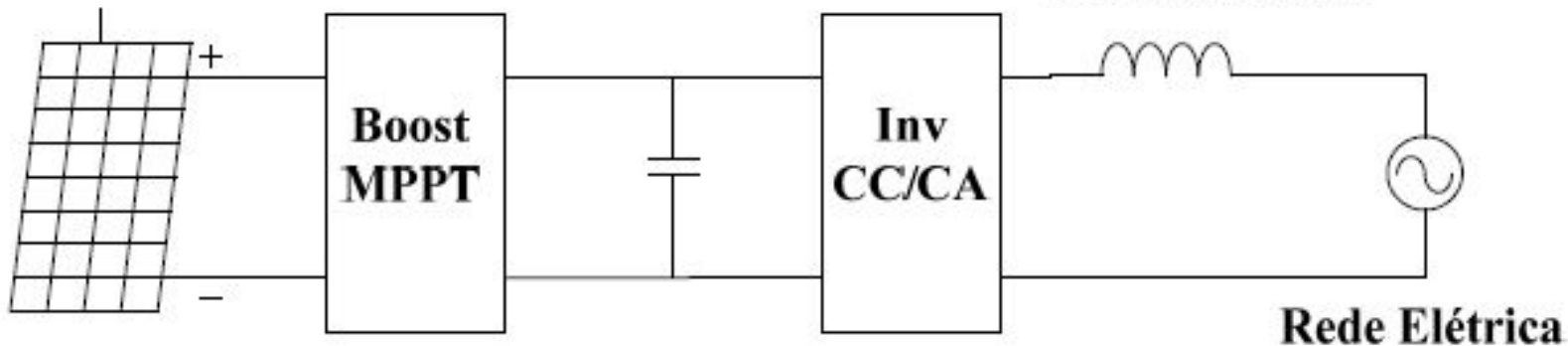
**Inv
CC/CA**



Filtro de saída

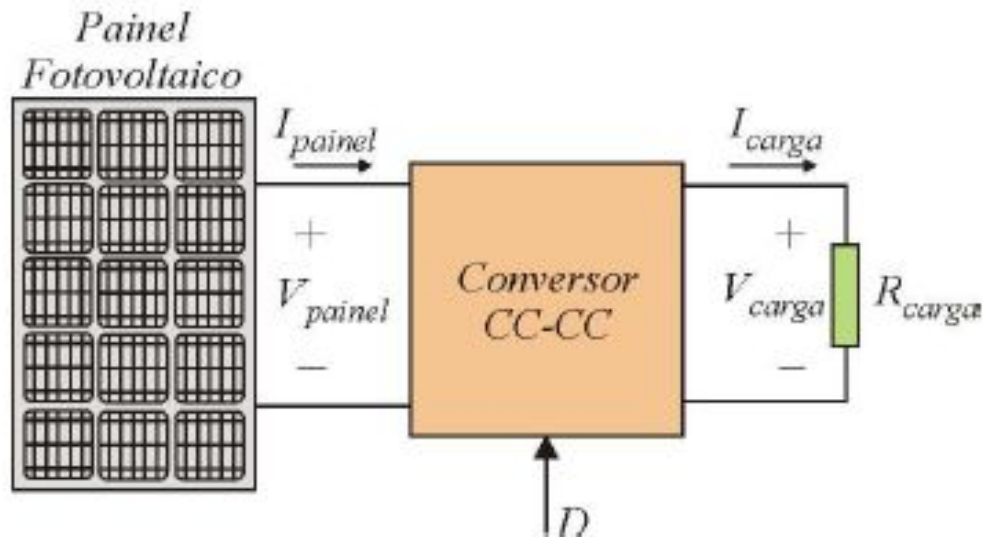


Rede Elétrica



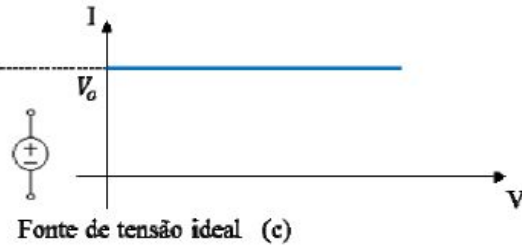
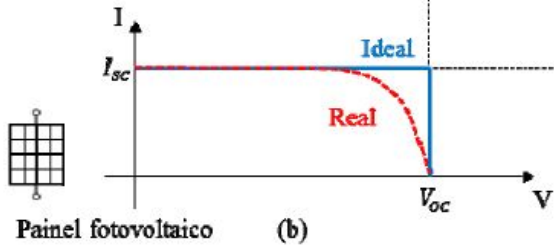
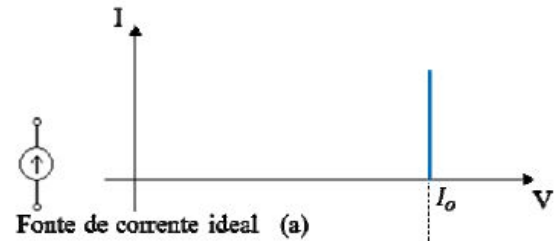
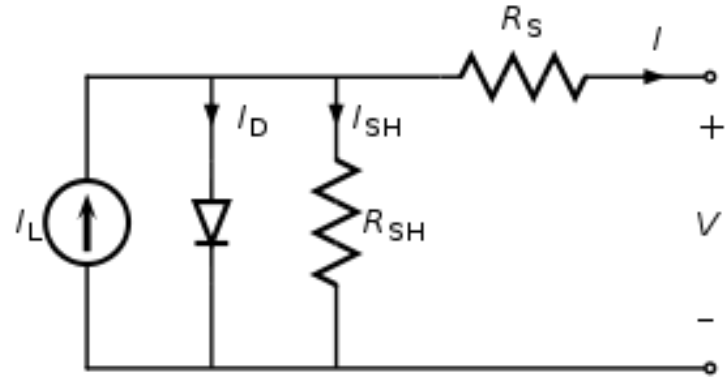
Sistema de geração

- Resumida



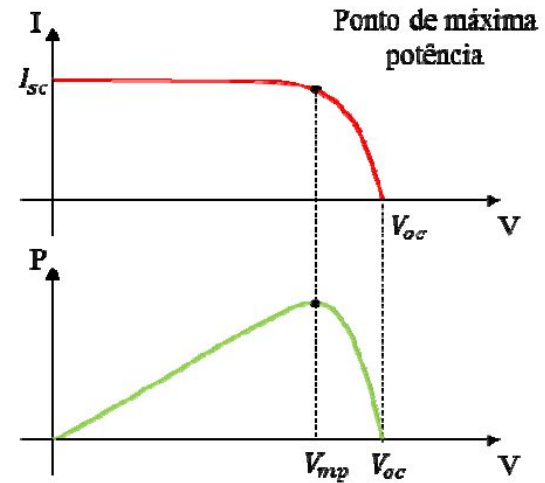
Painel Solar

- Características



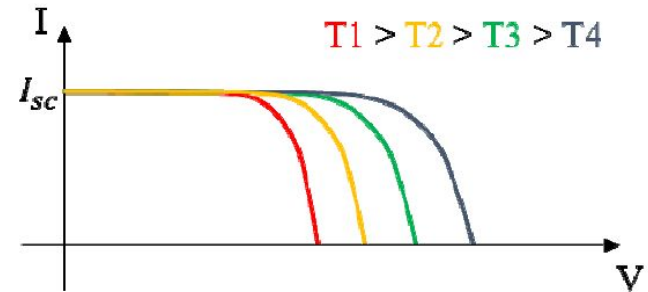
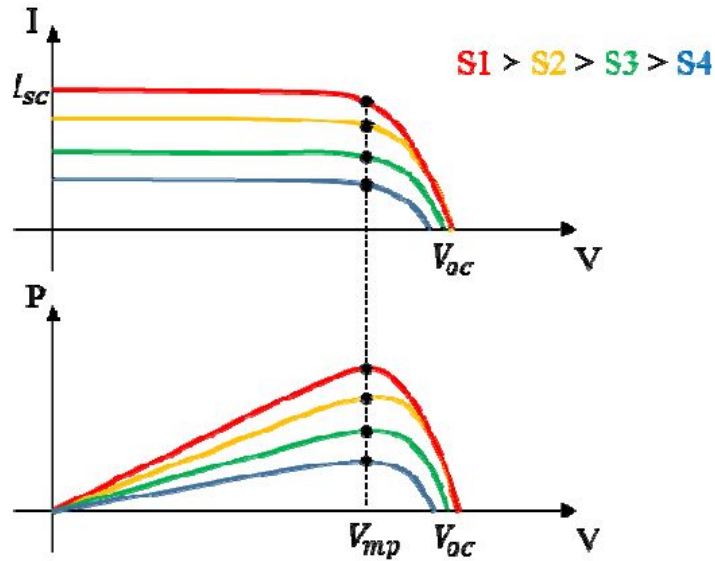
Painel Solar

- Curvas
- Ponte de máxima potência



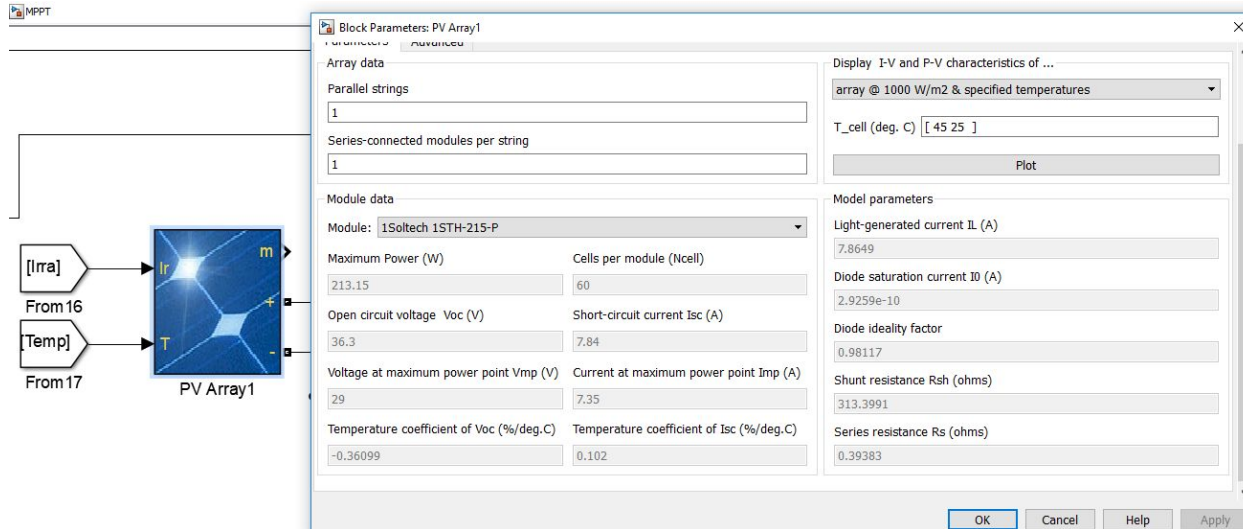
Painel Solar

- Condições ambientais



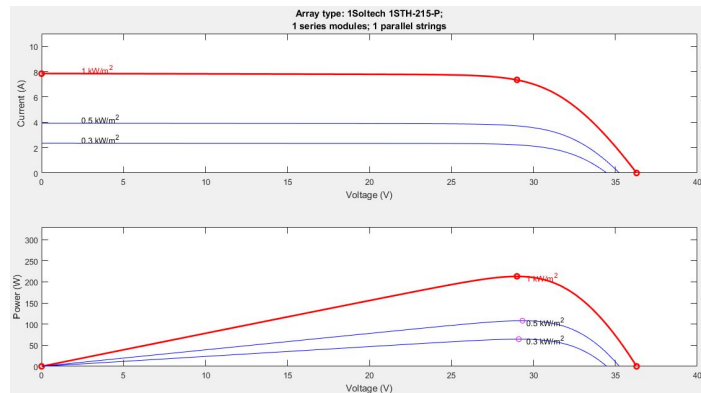
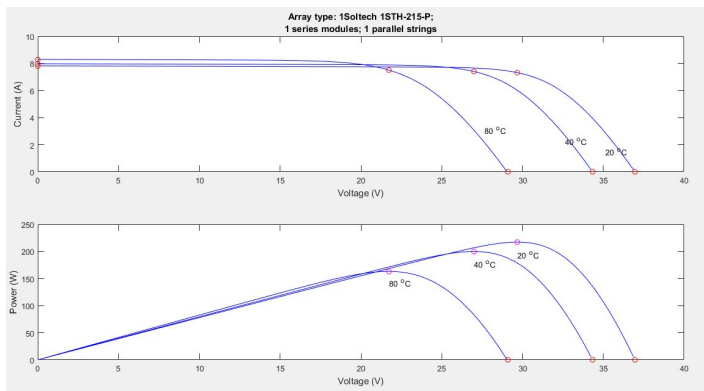
Painel Solar

- Modelo Simulink



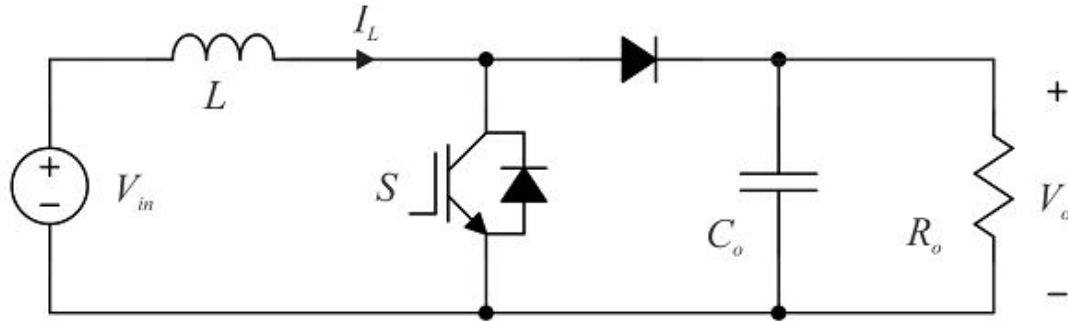
Painel Solar

- Modelo simulink
- Variações ambientais



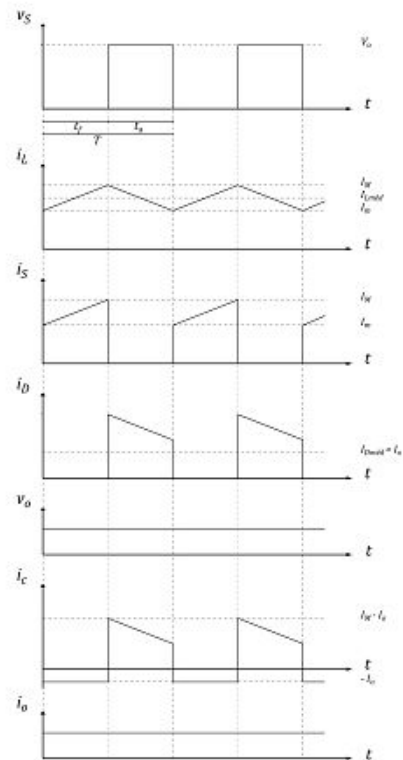
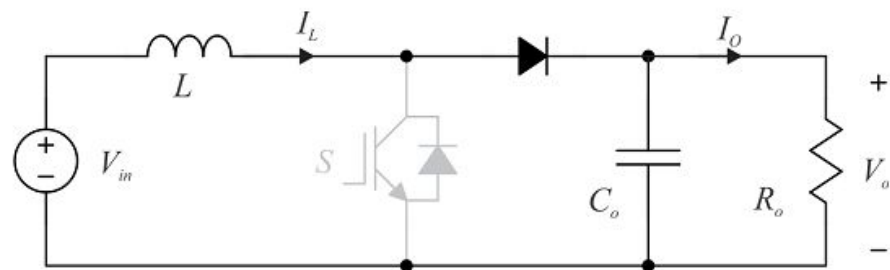
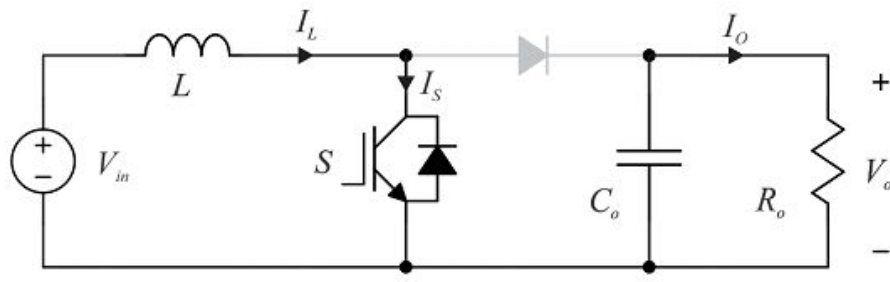
Conversor DC-DC

- Boost



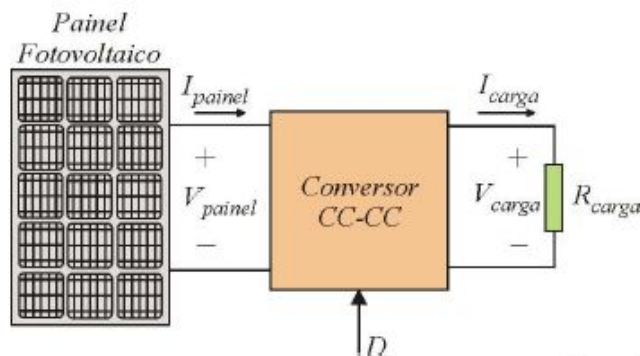
Conversor DC-DC

- Etapas de funcionamento
- MCC



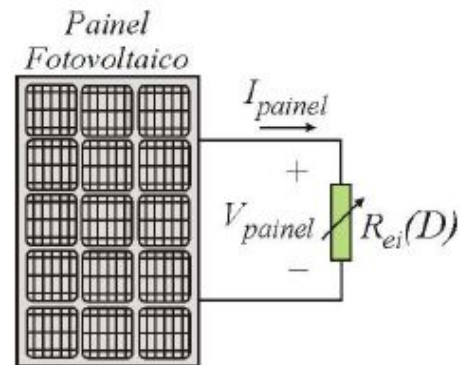
Conversor

- Resistencia de entrada
- Ganho estatico



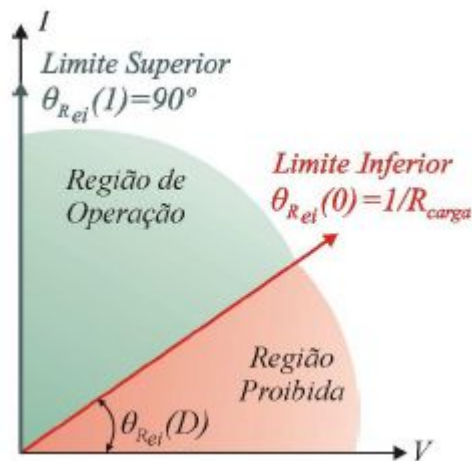
$$\frac{V_o}{V_{in}} = \frac{1}{1-D}$$

$$R_{ei}(D) = (1-D)^2 \cdot R_{carga}$$



Resistencia/Região de operação para demais DC-DC

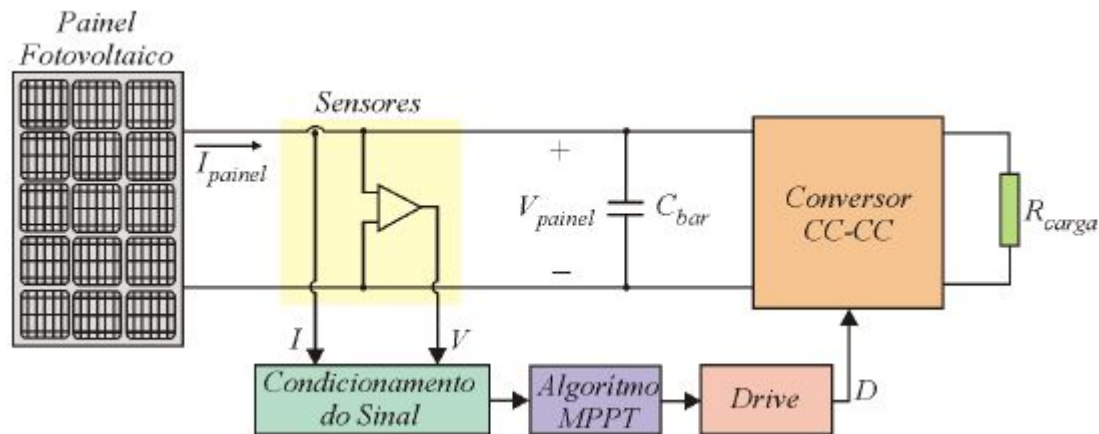
$$R_{ei}(D) = (1 - D)^2 \cdot R_{carga}$$



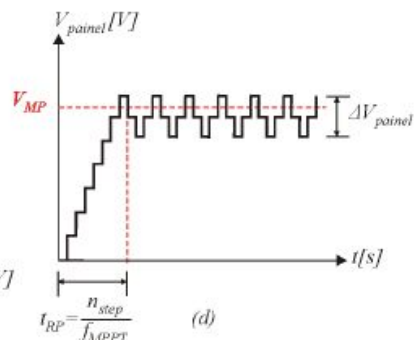
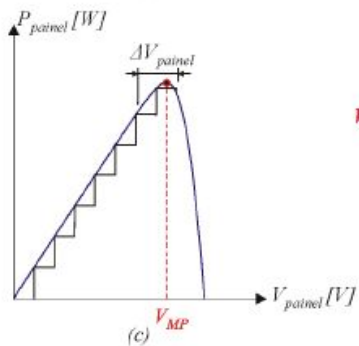
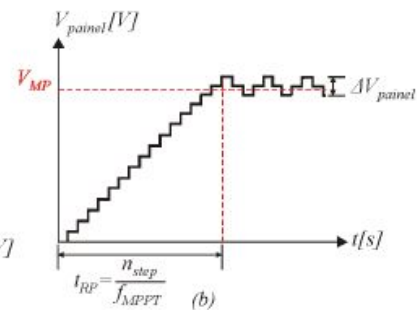
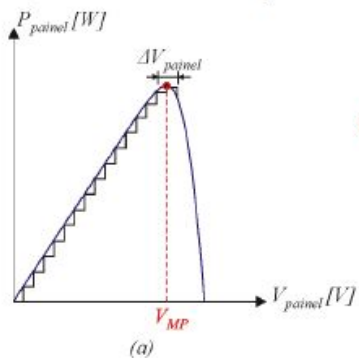
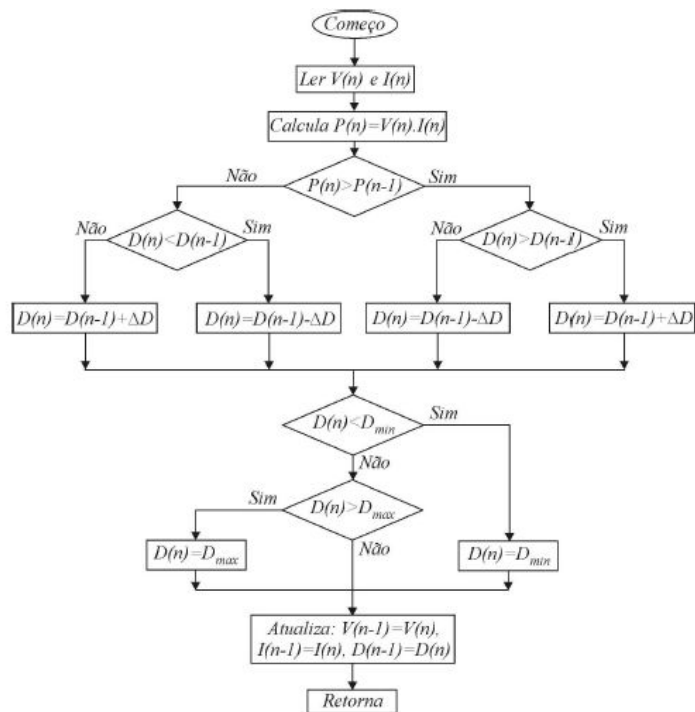
Conversor	G_V	$R_{ei}(G_V)$	$\theta_{rei}(G_V)$	Limites de $\theta_{rei}(D)$
Buck	D	$R_{ei}(G_V) = \frac{R_{carga}}{G_V^2}$	$\theta_{rei}(G_V) = \text{atan}\left(\frac{G_V^2}{R_{carga}}\right)$	$0^\circ < \theta_{rei}(D) < \text{atan}\left(\frac{1}{R_{carga}}\right)$
Boost	$\frac{1}{1-D}$	$R_{ei}(G_V) = \frac{R_{carga}}{G_V^2}$	$\theta_{rei}(G_V) = \text{atan}\left(\frac{G_V^2}{R_{carga}}\right)$	$\text{atan}\left(\frac{1}{R_{carga}}\right) < \theta_{rei}(D) < 90^\circ$
Buck-Boost, C'uk, Sepic e Zeta	$\frac{D}{1-D}$	$R_{ei}(G_V) = \frac{R_{carga}}{G_V^2}$	$\theta_{rei}(G_V) = \text{atan}\left(\frac{G_V^2}{R_{carga}}\right)$	$0^\circ < \theta_{rei}(D) < 90^\circ$

Sensor para MPPT

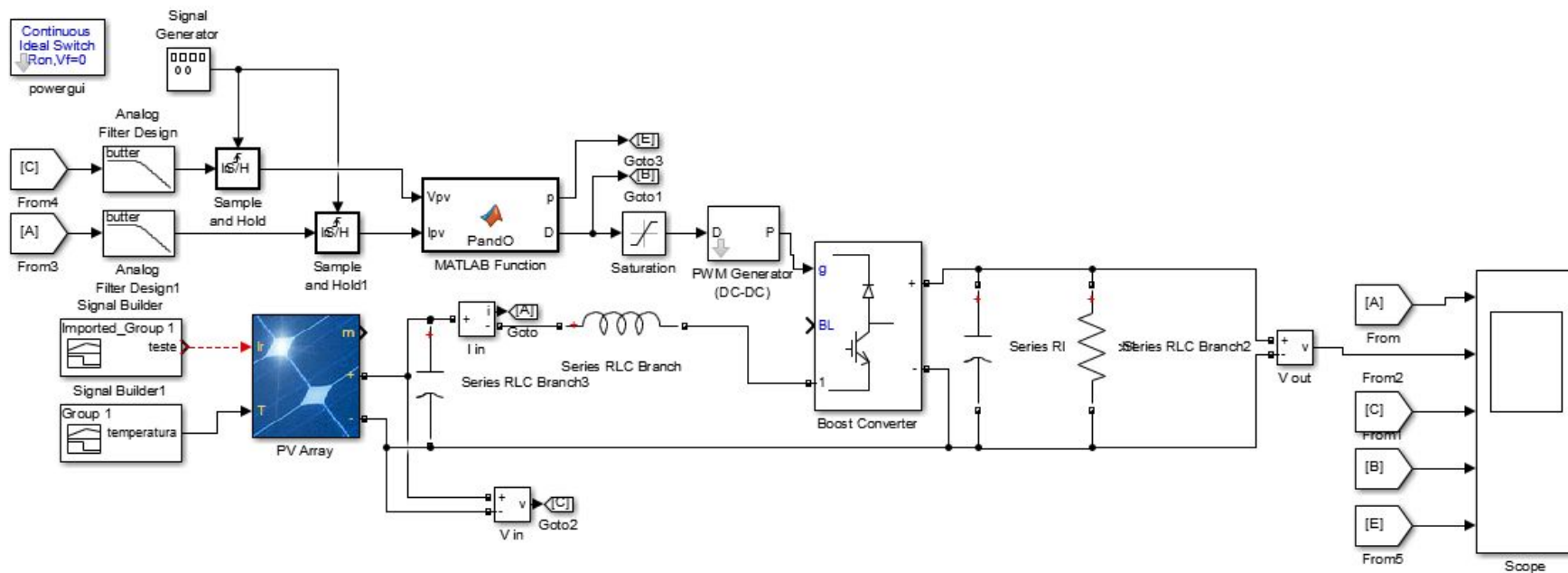
- Tensão e Corrente



Pertuba e Observa (P&O)

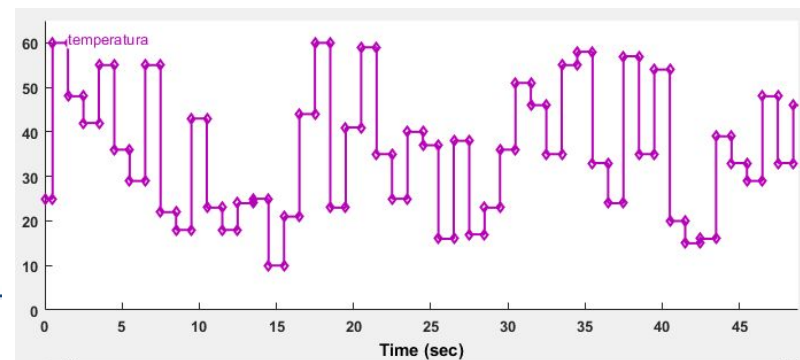
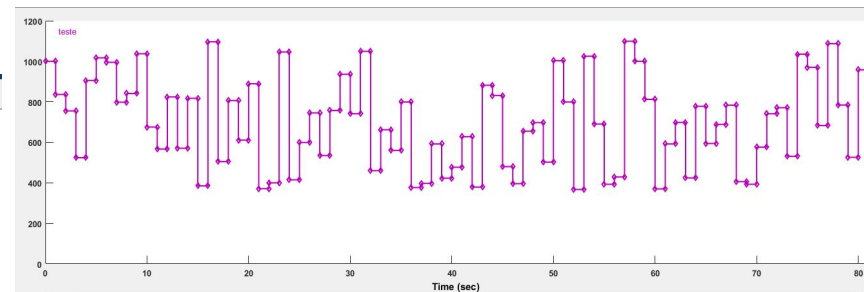


DATASET

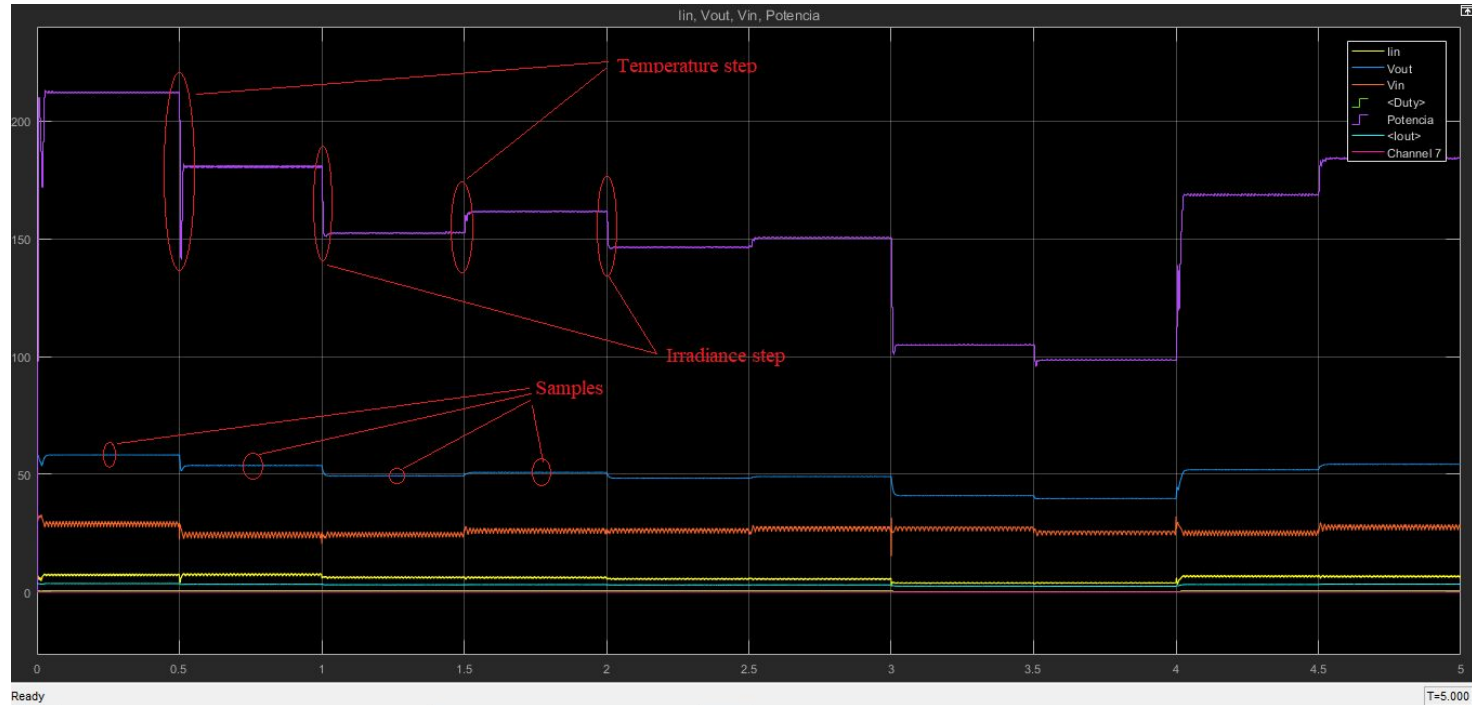


Gerando DATASET

```
entrada.m  X  +
1  s = 1000; %amostras geradas
2
3  radiacao.time = zeros(s*2,1);
4  radiacao.signals.values = zeros(s*2,1);
5  radiacao.signals.dimensions = 1;
6  radiacao.signals.label = 'teste';
7  radiacao.time(1) = 0;
8  radiacao.signals.values(1)= 1000;
9  radiacao.signals.values(2)= 1000;
10
11  for i = 2:2:s*2-1
12      radiacao.time(i) = (i)/2;
13      radiacao.time(i+1) = (i)/2;
14  end
15
16  for i = 3:2:s*2
17      rando = randi([300,1100],1,1) ;
18      radiacao.signals.values(i)= rando;
19      radiacao.signals.values(i+1)= rando;
20  end
21
22  radiacao.time(2000) = 1000;
23
24  save radiacao.mat radiacao
```



Gerando DATASET - simulação



Dados do DATASET

Tamanho: 2093 linhas e 7 colunas

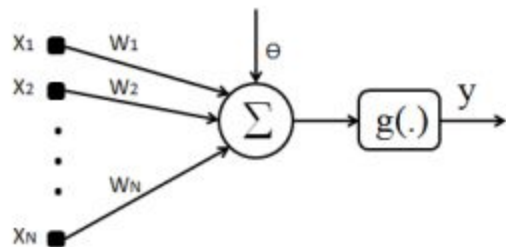
```
In [123]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.neural_network import MLPRegressor
```

```
In [125]: df = pd.read_csv('data/data_train.csv')
df.head()
```

Out[125]:

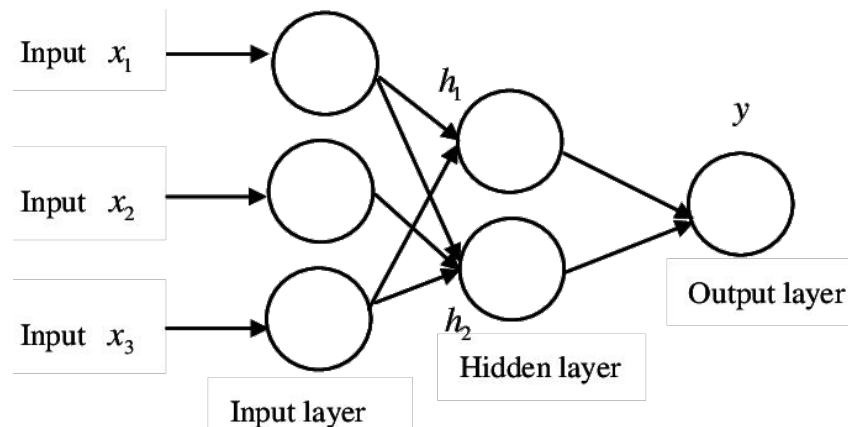
	Input_current	Output_voltage	Input_voltage	Duty_Cycle	Power	Irradiance	Temperature
0	6.818701	58.373037	29.782506	0.485	212.117783	1000	25
1	7.313437	53.815293	23.990136	0.557	180.751131	1000	60
2	6.056522	49.442379	24.298210	0.505	152.417456	836	60
3	5.735578	50.894729	26.911060	0.475	161.503052	836	48
4	5.376322	48.436413	26.198545	0.457	146.294526	755	48

Redes Neurais



Redes Neurais

- Forward Propagation e Simulação



```
In [191]: from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.neural_network import MLPRegressor

# Divide nosso conjunto de dados em Treino e Teste 0.7/0.3
X_train, X_test, y_train, y_test = train_test_split(x,y,test_size=0.15)

# Treinamos o modelo usando X_train e verificamos o score com o conjunto de teste
# Ajustamos as parametros do Regressor para que tivessemos um bom resultado.
modelo = MLPRegressor(activation='tanh', solver='lbfgs',alpha=0.00000001, hidden_layer_sizes=(150,150,150))
modelo.fit(X_train, y_train)
modelo.score(X_test, y_test)
```

Redes Neurais

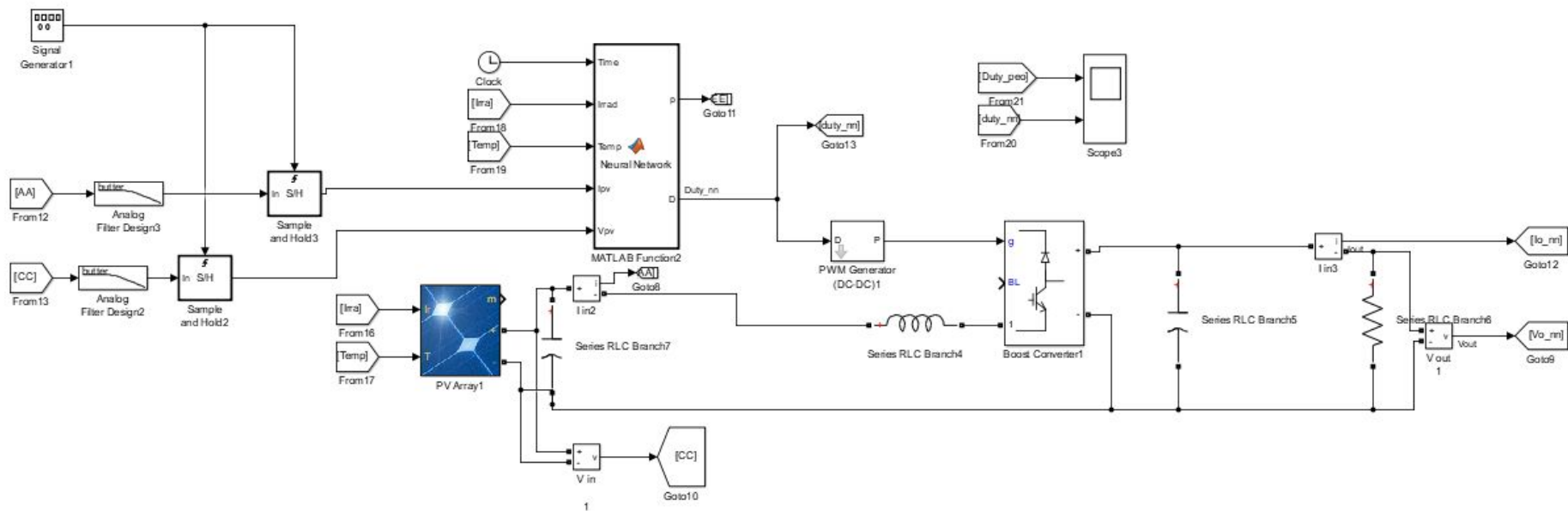
- Export dos pesos e bias para o Simulink

```
In [192]: coefs=np.asarray(modelo.coefs_[0])
np.savetxt("pesos_0.csv", coefs, delimiter=",")
coefs=np.asarray(modelo.coefs_[1])
np.savetxt("pesos_1.csv", coefs, delimiter=",")
coefs=np.asarray(modelo.coefs_[2])
np.savetxt("pesos_2.csv", coefs, delimiter=",")
coefs=np.asarray(modelo.coefs_[3])
np.savetxt("pesos_3.csv", coefs, delimiter=",")

bias=np.asarray(modelo.intercepts_[0])
np.savetxt("bias_0.csv", bias, delimiter=",")
bias=np.asarray(modelo.intercepts_[1])
np.savetxt("bias_1.csv", bias, delimiter=",")
bias=np.asarray(modelo.intercepts_[2])
np.savetxt("bias_2.csv", bias, delimiter=",")
bias=np.asarray(modelo.intercepts_[3])
np.savetxt("bias_3.csv", bias, delimiter=",")
```

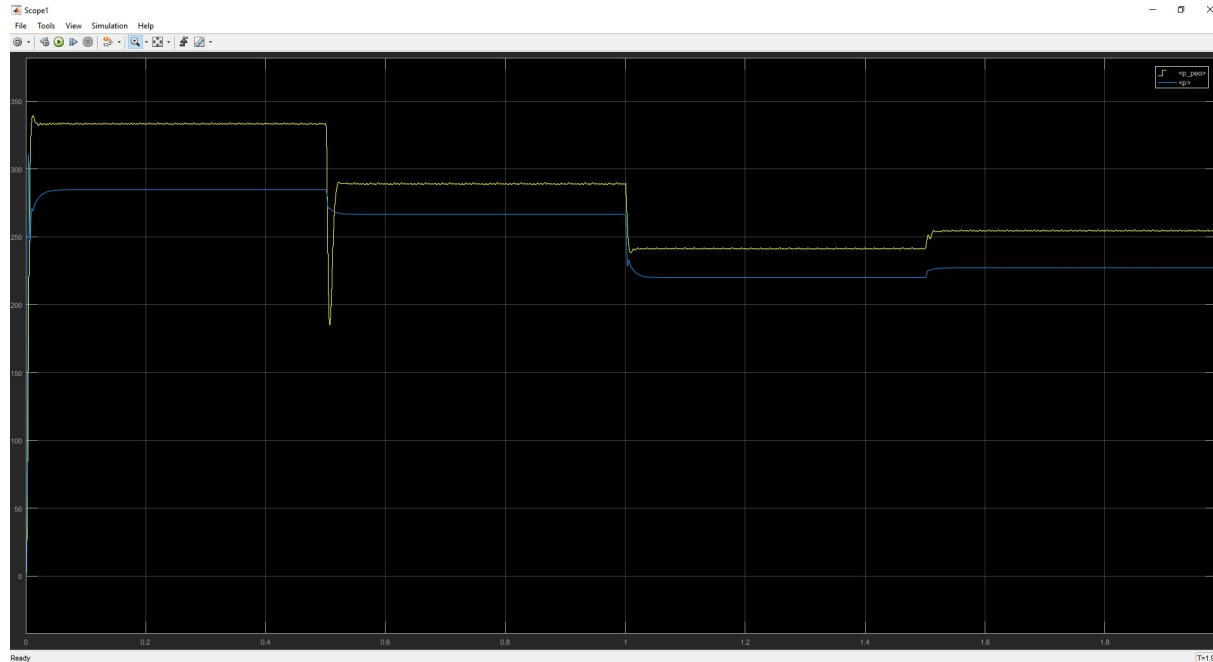
Redes Neurais

- Simulação no Simulink



Resultados

- Primeiras Simulações de Dificuldades
- Rede muito pequena - 2 Camadas e 4 neurônios em cada



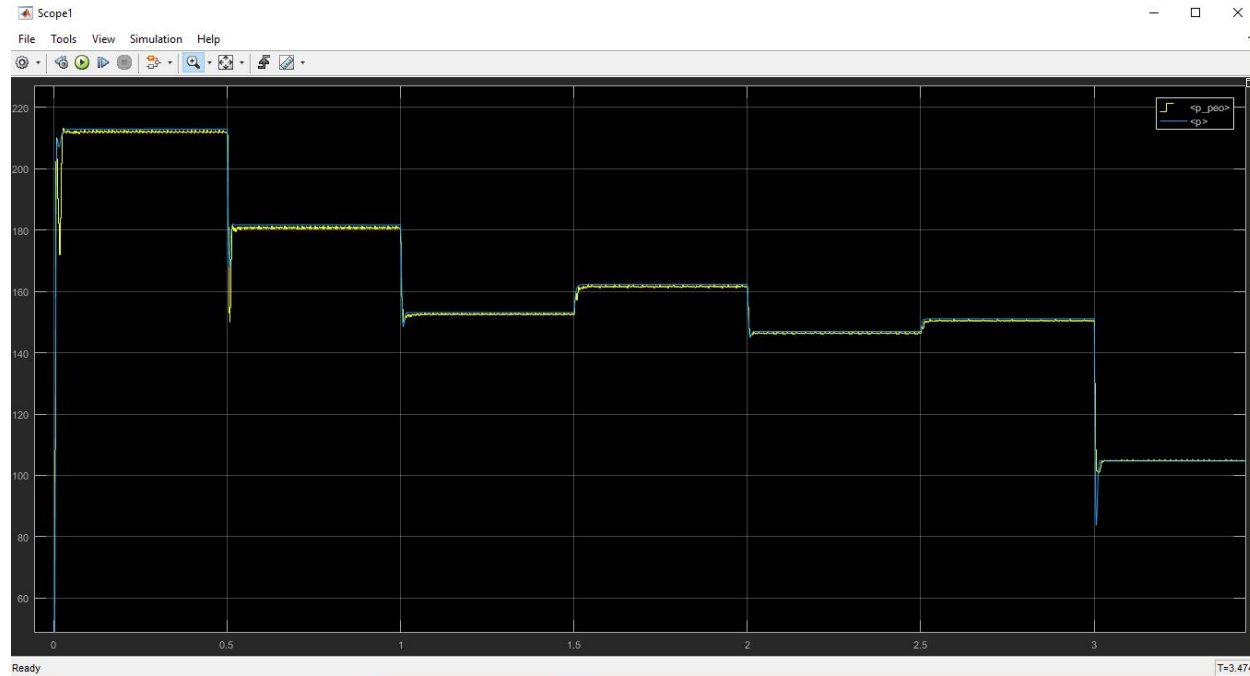
Resultados

- Primeiras Simulações de Dificuldades
- Ativação ReLU



Resultados

- Simulação com 3 camadas e 150 neurônios, ativação Tanh, score de 99,48%
- Comparação de Potência



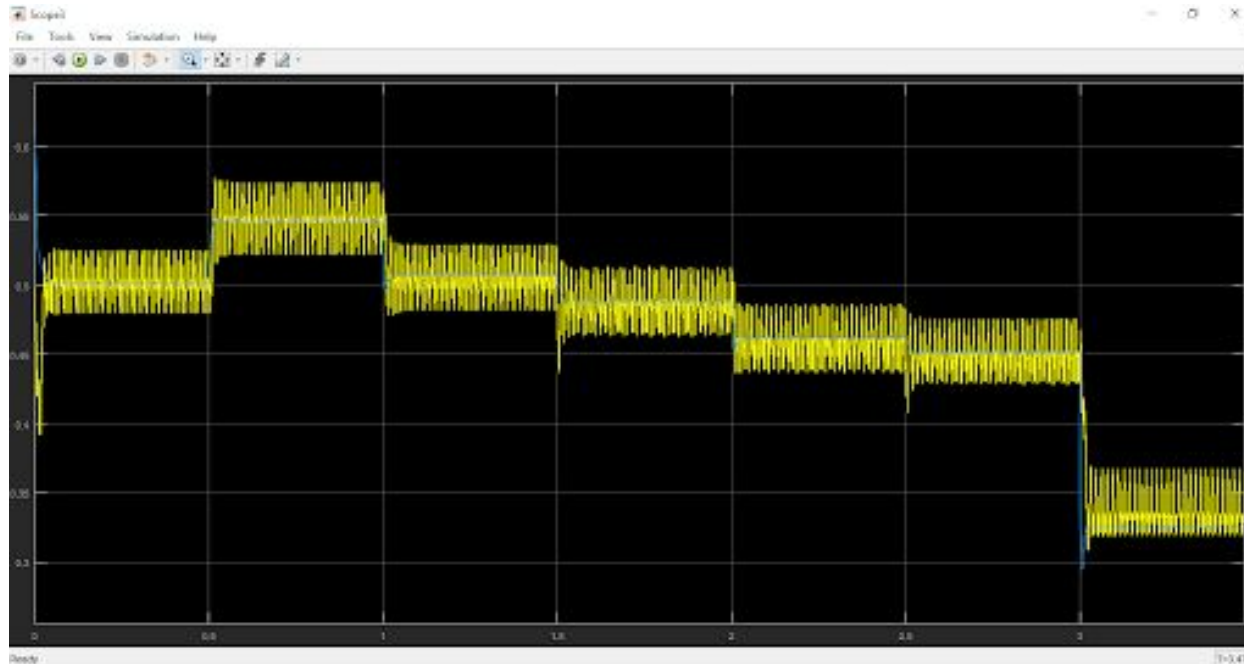
Resultados

- Detalhe para a resposta ao transitório



Resultados

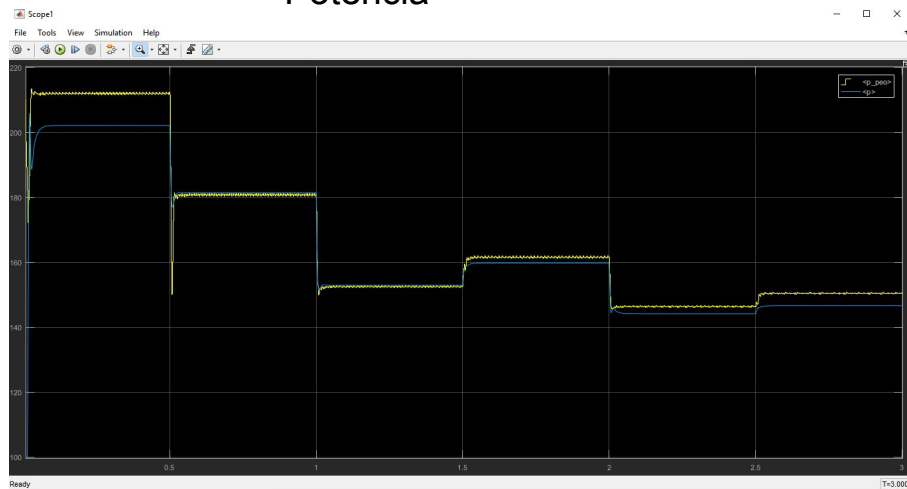
- Simulação com 3 camadas e 150 neurônios, ativação Tanh, score de 99,48%
- Comparação de Duty Cycle
-



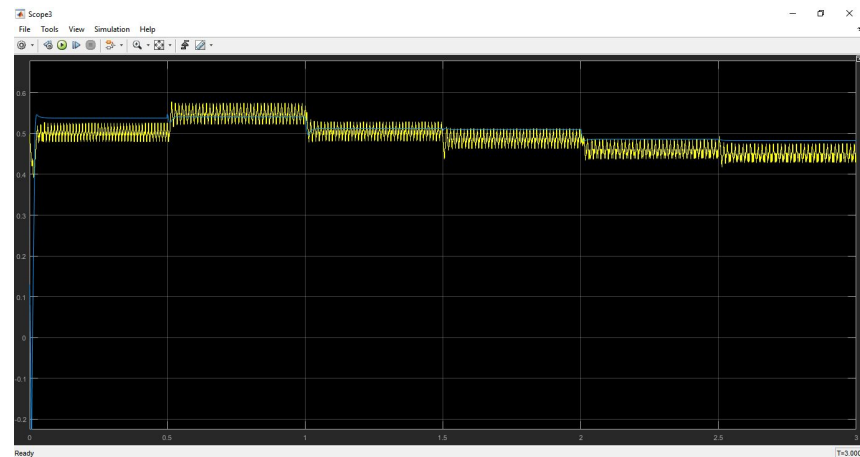
Resultados

- Simulação com 3 camadas e 10 neurônios, ativação Tanh, score de 99,35%

Potência



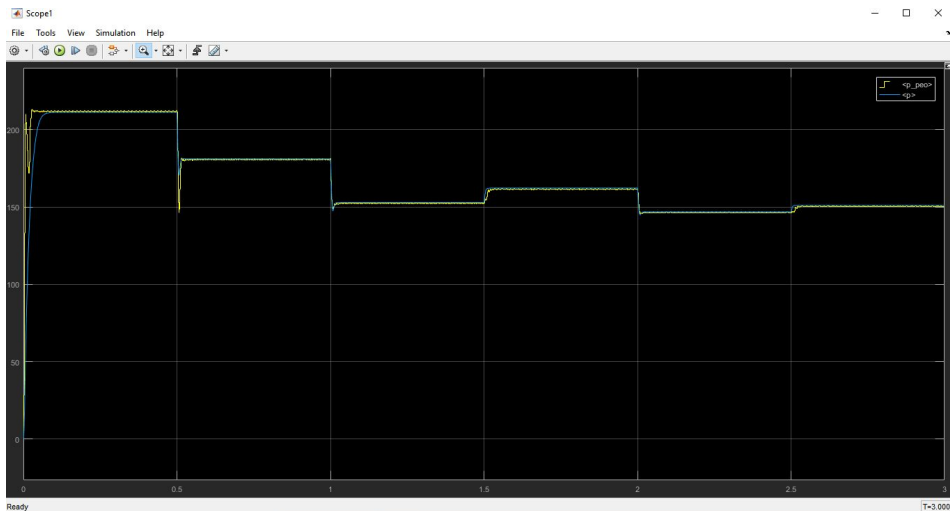
Duty Cycle



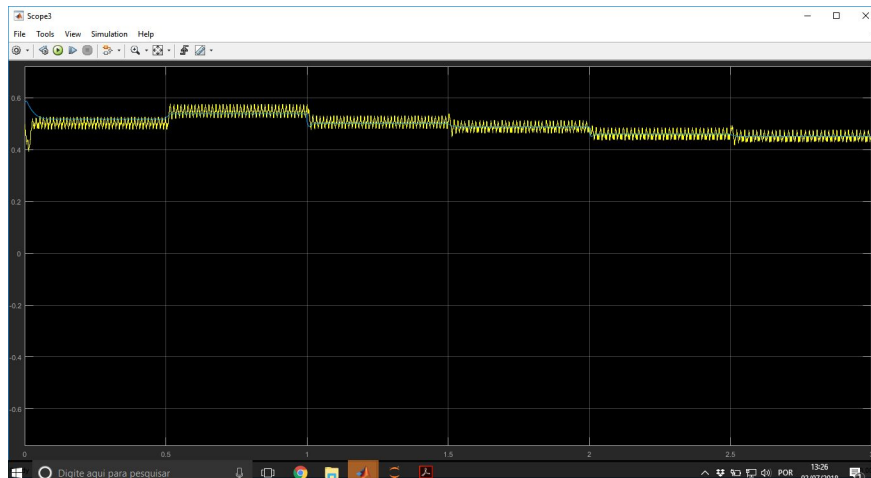
Resultados

- Simulação com 3 camadas e 50 neurônios, ativação Tanh, score de 99,40%

Potência



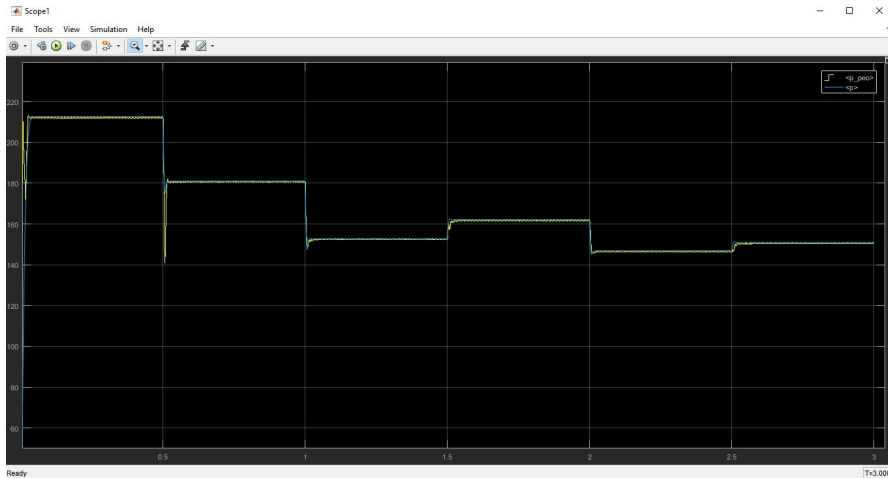
Duty Cycle



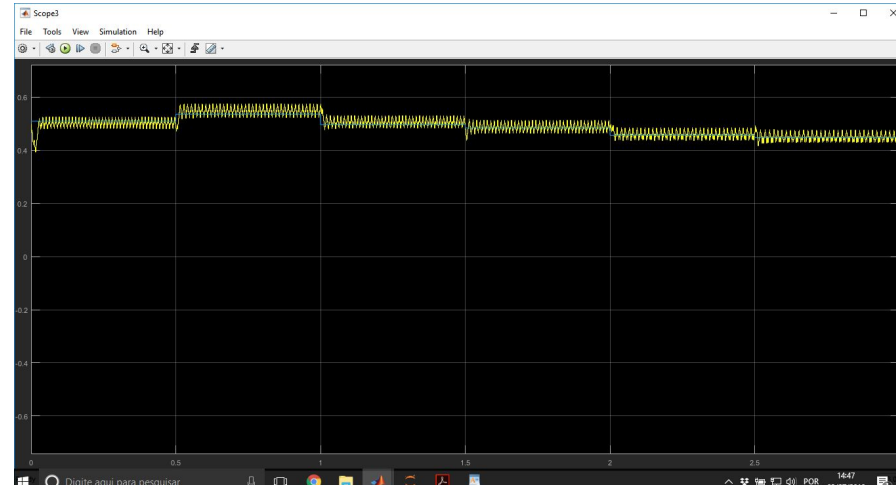
Resultados

- Simulação com 3 camadas e 75 neurônios, ativação Tanh, score de 98,20%
Com entradas apenas Irradiância e Temperatura

Potência



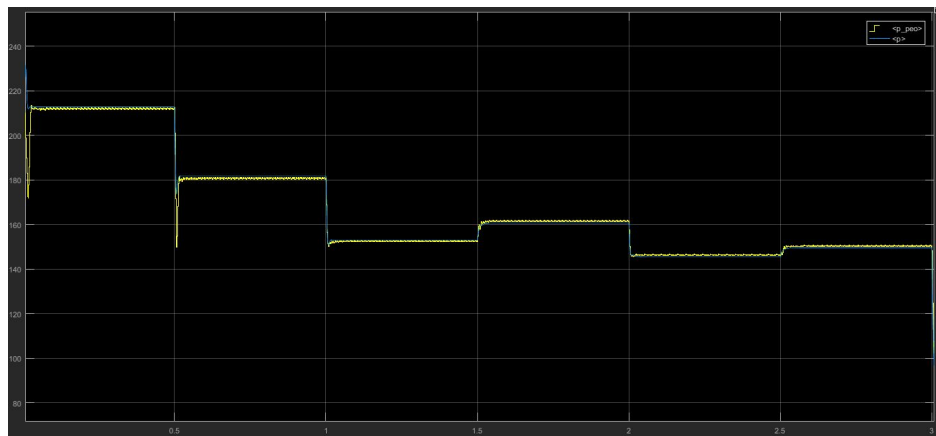
Duty Cycle



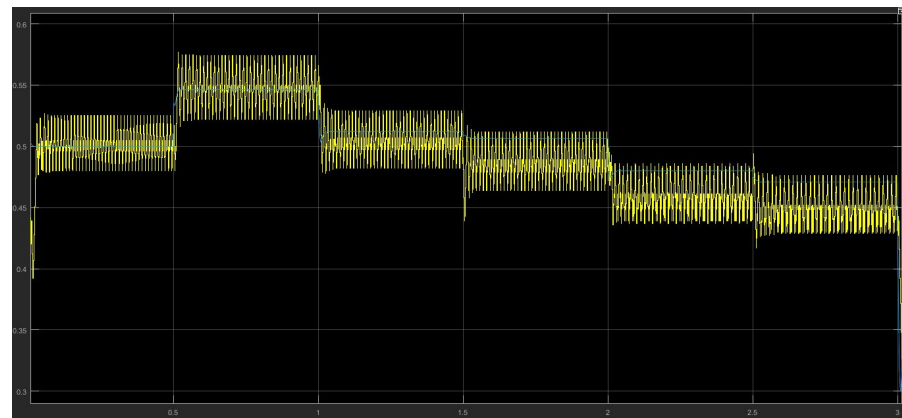
Resultados

- Simulação com 2 camadas e 250 neurônios, ativação Tanh, score de 99,27%

Potência



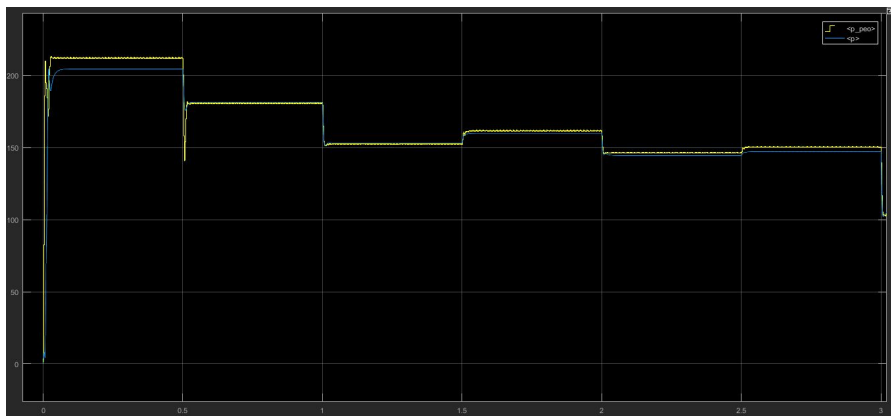
Duty Cycle



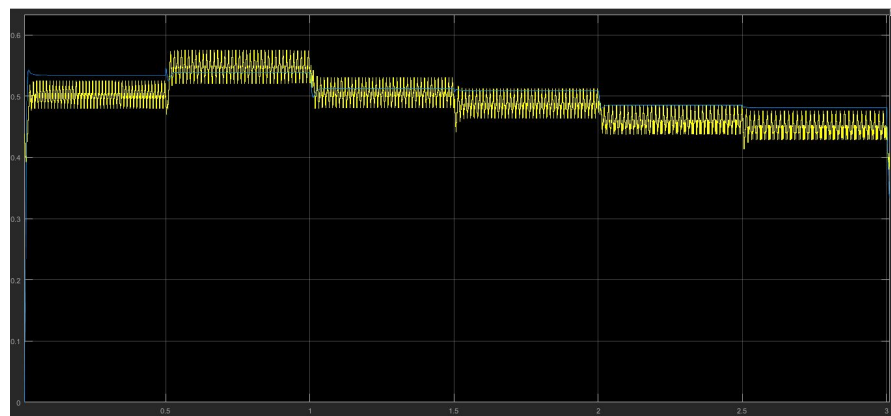
Resultados

- Simulação com 10 camadas e 50 neurônios, ativação Tanh, score de 99,39%

Potência



Duty Cycle



Conclusão

Dentre os resultados obtidos, alguns foram bons e outros não tão satisfatórios.

- Com relação a função de ativação a Tanh teve um desempenho considerável melhor que a Relu, não conseguimos treinar uma boa rede que funciona-se com ativação Relu.
- Quando treinamos redes com 3 hidden layer só obtivemos resultados satisfatórios quando a rede era tinha mais de 50 neurônios, esse teste foi feito com 10 25 35 50 150.
- O painel influencia no treinamento, caso fosse necessário usar outro painel seria necessário gerar um novo data-set ou talvez fazer alguma calibração no data-set
- Melhorias no sistema podem ser obtidas para se generalizar a adição de redes com diferentes tamanhos de camadas e diferentes ativações para se verificar o comportamento nestas condições.
- Devido a demora no tempo de simulação e a dificuldade de se importar funções externas ao Simulink não foi possível se realizar simulações durante um grande período de tempo com diversos parâmetros diferentes.

OBRIGADO