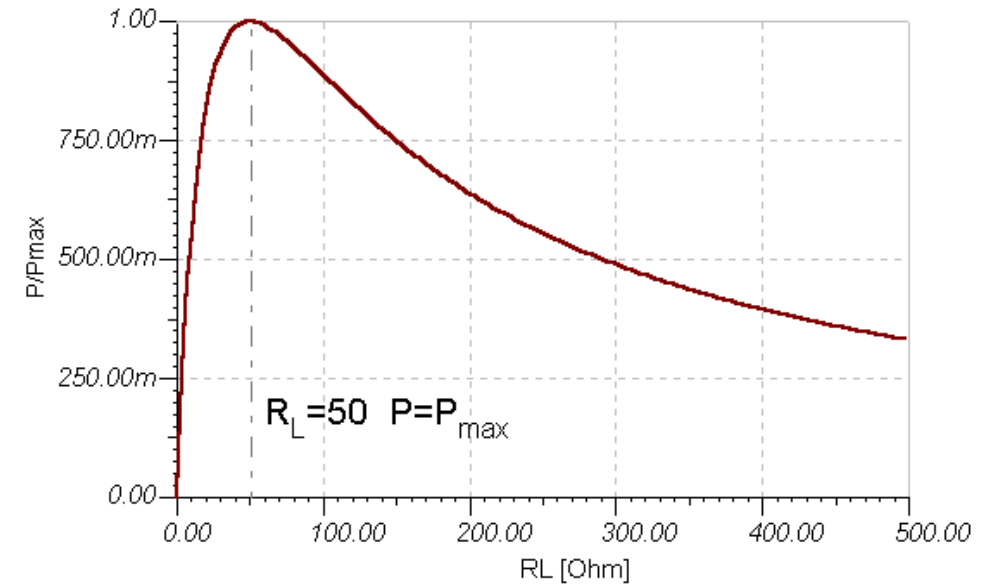
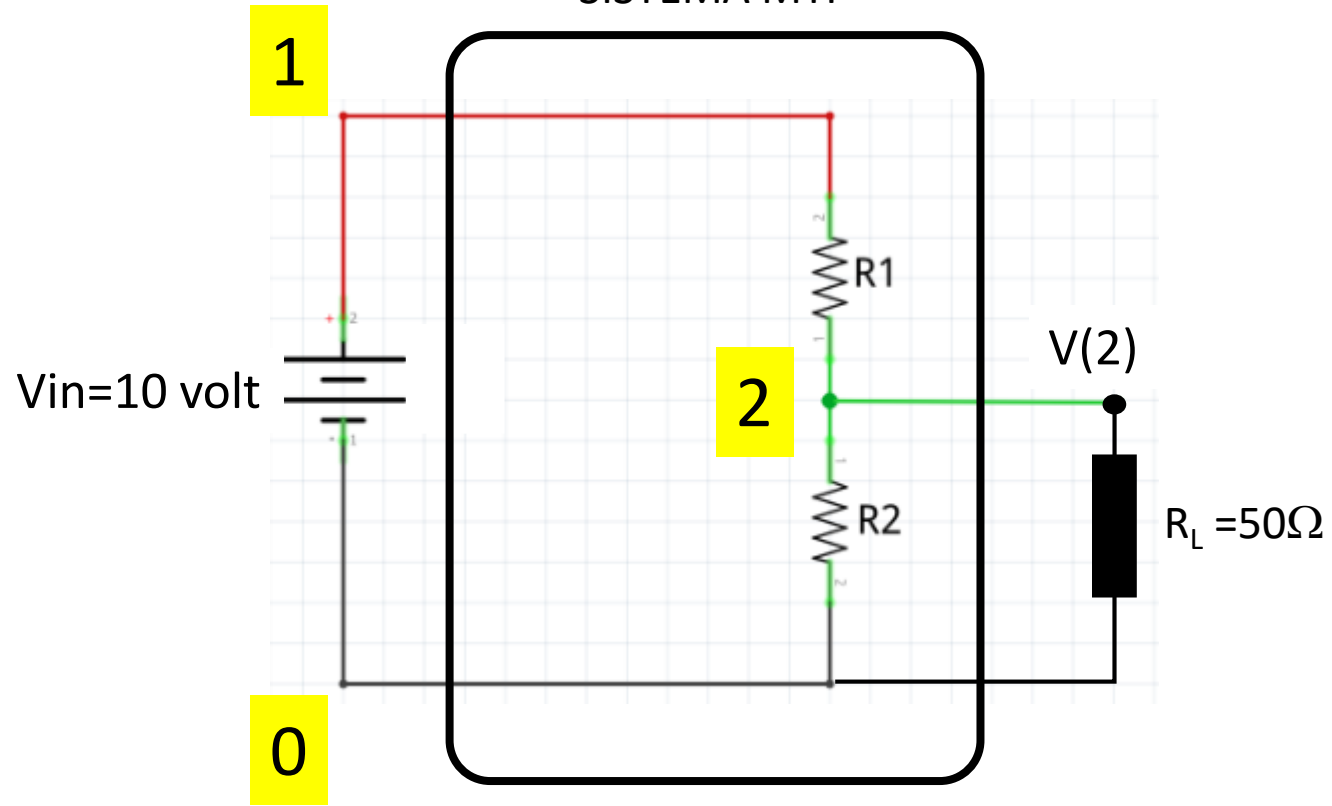


## Atividade 2: Aplicação de Estratégias Evolutivas para o projeto de circuitos elétricos

# Desenvolver as seguintes questões

1. Implementar o algoritmo da estratégia evolutiva para a determinação das resistências  $R_1$  e  $R_2$  que permitam transferir a máxima potência para a carga  $R_L$
2. Construa uma função de fitness, justifique sua resposta
3. Implemente os operadores de seleção de Roleta e Torneio, compare suas respostas
4. Grafique a função de fitness versus o número de gerações, para diferentes execuções do algoritmo. Determine a média de número de gerações na qual o algoritmo encontra as soluções do problema
5. Compare as diferentes configurações do algoritmo, tais como: Tamanho da população, Probabilidade de cruzamento, Probabilidade de mutação, taxa de elitismo e número de gerações

# SISTEMA MTP



```
import PySpice.Logging.Logging as Logging
logger = Logging.setup_logging()
```

```
from PySpice.Spice.Netlist import Circuit
from PySpice.Unit import *
```

```
circuit = Circuit('Maxima Transferencia de Potencia')
Vin = 10@u_V
circuit.V('input', 1, circuit.gnd, Vin)
R1 = 3@u_Ω
R2 = 1@u_Ω
RL = 50@u_Ω
circuit.R(1, 1, 2, R1)
circuit.R(2, 2, circuit.gnd, R2)
circuit.R(3, 2, circuit.gnd, RL)

simulator = circuit.simulator(temperature=25, nominal_temperature=25)
analysis = simulator.operating_point()

for node in analysis.nodes.values():
    print('Node {}: {:.1f} V'.format(str(node), float(node)))
```

Node 2: 2.5 V

Node 1: 10.0 V

## Definir os valores de R1 e R2

```
circuit = Circuit('Maxima Transferencia de Potencia')
Vin = 10@u_V
circuit.V('input', 1, circuit.gnd, Vin)
R1 = 3@u_Ω
R2 = 1@u_Ω
RL = 50@u_Ω
circuit.R(1, 1, 2, R1)
circuit.R(2, 2, circuit.gnd, R2)
circuit.R(3, 2, circuit.gnd, RL)
```

## Simulação usando NGSpice



## Leitura da tensão V(2)

```
for node in analysis.nodes.values():
    print('Node {}: {:.4f} V'.format(str(node), float(node)))
```

Node 2: 2.5 V  
Node 1: 10.0 V

$$P = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \\ \dots & \\ R_{L1} & R_{L2} \end{bmatrix}$$

Atualizar a População

Avaliar o par (R1, R2) usando a função de FITNESS para a máxima transferência de Potência

Operador de Seleção

Cruzamento, Mutação e Elitismo

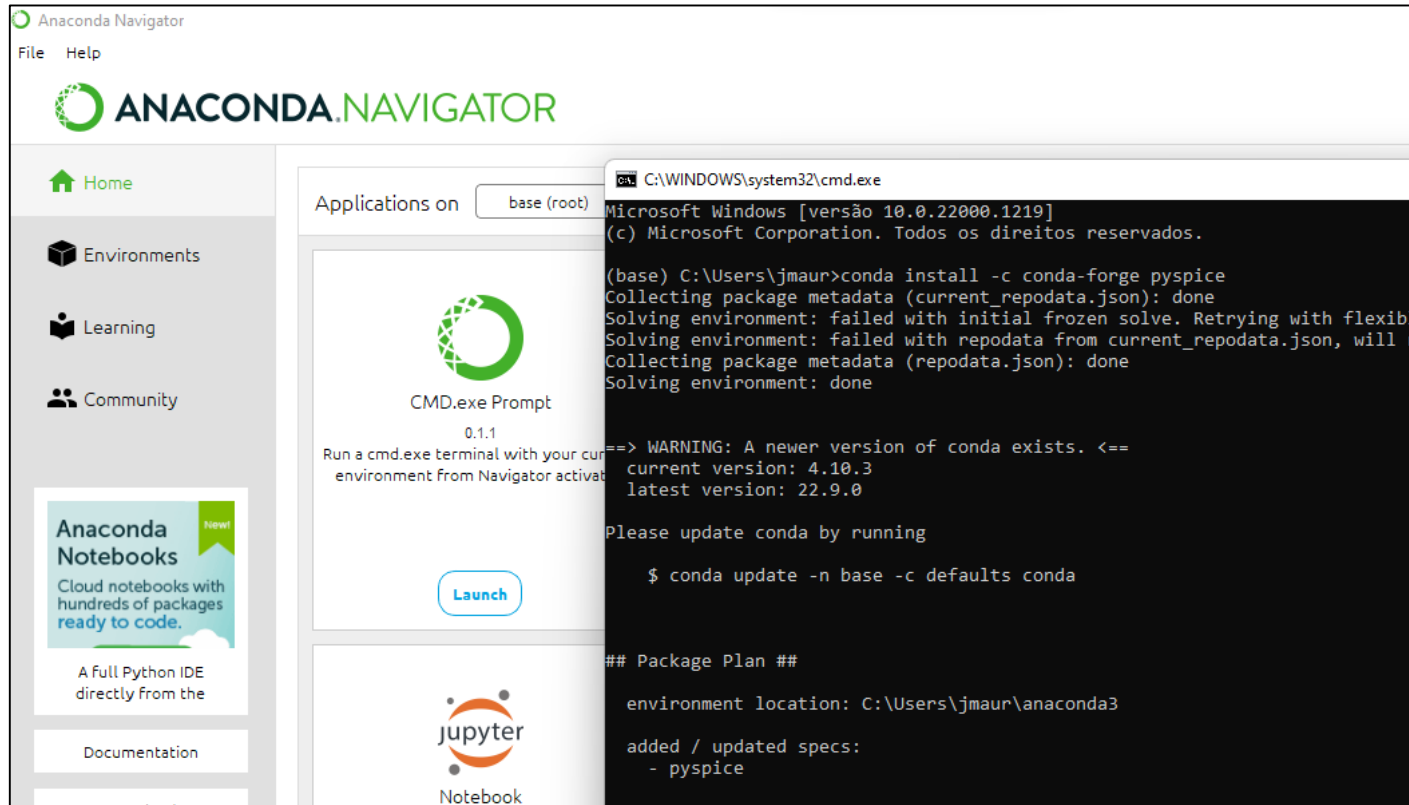
# Passos de Instalação do PYSPICE e NGSPICE usando ANACONDA

<https://pyspice.fabrice-salvaire.fr/releases/v1.4/installation.html>

# Usando Anaconda

<https://pyspice.fabrice-salvaire.fr/releases/v1.4/installation.html>

```
conda install -c conda-forge pyspice
```



The following NEW packages will be INSTALLED:

ngspice-lib	conda-forge/win-64::ngspice-lib-37-h7a24507_1
pyspice	conda-forge/win-64::pyspice-1.5-py39hcbf5309_1
python_abi	conda-forge/win-64::python_abi-3.9-2_cp39
winflexbison	conda-forge/win-64::winflexbison-2.5.22-he025d50_0

The following packages will be UPDATED:

conda	pkgs/main::conda-4.10.3-py39haa95532_0 --> conda-forge::conda-4.12.0-py39hcbf5309_0
-------	---

pyspice-post-installation --install-ngspice-dll

```
C:\WINDOWS\system32\cmd.exe
(base) C:\Users\jmaur>pyspice-post-installation --install-ngspice-dll
Get https://sourceforge.net/projects/ngspice/files/ng-spice-rework/34/ngspice-34_dll_64.zip ... -> C:\Users\jmaur\AppData\Local\Temp\tmpk8rn9w4i\ngspice-34_dll_64.zip
Download failed, trying another URL...
Get https://sourceforge.net/projects/ngspice/files/ng-spice-rework/old-releases/34/ngspice-34_dll_64.zip ... -> C:\Users\jmaur\AppData\Local\Temp\tmpk8rn9w4i\ngspice-34_dll_64.zip
Extracted C:\Users\jmaur\AppData\Local\Temp\tmpk8rn9w4i\ngspice-34_dll_64.zip in C:\Users\jmaur\anaconda3\lib\site-packages\PySpice\Spice\NgSpice\Spice64_dll
=====
* Standard ngspice init file
alias exit quit
alias acct rusage all
set x11lineararcs
*set rndseed=12
** ascii rawfile **
set filetype=ascii
** frontend debug output **
*set ngdebug
** asking after quit **
*set askquit
** set the number of threads in openmp
** default (if compiled with --enable-openmp) is: 2
set num_threads=4
set interactive

strcmp __flag $program "ngspice"
if $__flag = 0

* Load the codemodels
codemodel ../lib/ngspice/spice2poly.cm
```



## pyspice-post-installation --check-install

```
C:\WINDOWS\system32\cmd.exe

=====
(base) C:\Users\jmaur>
(base) C:\Users\jmaur>pyspice-post-installation --check-install
OS: win32


Environments:
PATH C:\Users\jmaur\anaconda3;C:\Users\jmaur\anaconda3\Library\mingw-w64\bin;C:\Users\jmaur\anaconda3\Library\bin;C:\Users\jmaur\anaconda3\Scripts;C:\Users\jmaur\anaconda3\condabin;C:\Users\jmaur\anaconda3;C:\Users\jmaur\anaconda3\Library\mingw-w64\bin;C:\Users\jmaur\anaconda3\Library\bin;C:\Users\jmaur\anaconda3\Scripts;C:\Program Files\Java\javapath;C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\system32\wbem;C:\WINDOWS\System32\OpenSSH;C:\Program Files\MATLAB\R2022b\bin;C:\Program Files\MathWorks\bin;C:\Program Files (x86)\Microsoft SQL Server\120\bin;C:\Program Files (x86)\IVI Foundation\VISA\WinNT\Bin;C:\Users\jmaur\AppData\Local\Microsoft VS Code\bin;C:\Program Files\Spice64\bin;C:\Program Files\Spice64\bin;LD_LIBRARY_PATH undefined
PYTHONPATH undefined

*****
** ngspice-34 : Circuit level simulation program
** The U. C. Berkeley CAD Group
** Copyright 1985-1994, Regents of the University of California.
** Copyright 2001-2020, The ngspice team.
** Please get your ngspice manual from http://ngspice.sourceforge.net/docs.html
** Please file your bug-reports at http://ngspice.sourceforge.net/bugrep.html
** Creation Date: Jan 29 2021 16:38:37
**
** CIDER 1.b1 (CODECS simulator) included
** XSPICE extensions included
** Relevant compilation options (refer to user's manual):
** OpenMP multithreading for BSIM3, BSIM4 enabled
** X11 interface not compiled into ngspice
**
*****

PySpice should work as expected

(base) C:\Users\jmaur>
```

<https://pyspice.fabrice-salvaire.fr/releases/v1.4/examples/resistor/voltage-divider.html>



Notebook

6.4.5

Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

Launch

## 8.15.2. Voltage Divider

[voltage-divider.py](#)  [voltage-divider.ipynb](#)  [Open in nbviewer](#) 

This example shows the computation of the DC bias and sensitivity in a voltage divider.

```
import PySpice.Logging.Logging as Logging
logger = Logging.setup_logging()

from PySpice.Spice.Netlist import Circuit
from PySpice.Unit import *
```

