

NUMERICAL MODELING OF CASCADED PHOTOVOLTAIC CELLS IN OPEN SOURCE SIMULATION ENVIRONMENT

Sergiu-Alex RANGA, Botond Sandor KIREI, Marina Dana TOPA

Technical University of Cluj-Napoca, Cluj-Napoca, Romania

Str. Memorandumului, Nr. 28, Cluj-Napoca, Romania, +40264202453, Email: botond.kirei@bel.utcluj.ro

Abstract: This paper aims to provide an easy to understand guide to PV modeling in Scilab/Xcos. It also highlights the importance of the Newton-Raphson method and its role in plotting the P-V and I-V characteristics. The current literature is abundant in Simulink numerical models, but they lack the detailed implementation of the Newton-Raphson method. An important contribution of this paper is the graphical implementation of the Newton-Raphson method, tailored for solving the current equation of the (cascaded) PV cells.

Keywords: Photovoltaic Array, Scilab/Xcos, P-V and I-V curve, Simulation

I. INTRODUCTION

In the last decade, a proliferation of photovoltaic energy generation is prevalent, largely due to the fact that the other sources of energy (oil, natural gases, etc.) have diminished over the years. As technology evolved so did the methods of modeling PV modules. One of the earliest examples uses a combination of Matlab m-file and C-language programming [1]. This can be problematic for someone who is not experienced in using a simulation environment. Other significant studies are [2] and [3]. Both employ lines of code written in C and neither show the readers the Simulink blocks used. The difficulty that arises with these articles is that the reader must understand the C language as well as have an understanding of Matlab [4].

Paper [4] is by far the most beginner-friendly available, offers an extensive look at the modeling of photovoltaic cells and modules. It presents a step-by-step guide, explaining all the parameters that affect the current vs. voltage (I-V) and power vs. voltage (P-V) characteristics of the solar PV array. This way, readers have an easier time understanding the intricacies of developing solar models.

There are two types of solar cells on the market: single-junction solar cells and multi-junction solar cells (double-junction, triple-junction, etc.). An increase in the number of junctions leads to an increase in the solar efficiency of the cell [5]. The idea behind multi-junction cells is to convert as much of the solar irradiation as possible by stacking up junctions made from different materials, which have different band gap energies. The maximum theoretical efficiency of a single junction solar cell is 33.16%, while an infinite number of junctions would hit a limit at 88.6% efficiency. An example of a single-junction cell would be n-CdS/p-CdTe, whereas a double-junction one would be n-CdS/p-CdTe/n-In2S3/p-CuInS [6].

The models can be implemented in a simulation environment with a graphical interface such as Simulink, but this paper proposes an open-source platform, Scilab/Xcos. The advantages of this software are that it comes with a free license, it evolves constantly as

developers add new features, the existing problems can be solved in a short amount of time due to its large number of users. Being open-source means anybody can bring improvements to it and can adapt it to meet their business requirements. Xcos also encourages experimentation as it is readily available and anyone interested in researching and developing new simulation models can do so without the need of a fee. There are also some disadvantages: there aren't many ready-made models available and there is also a lack of models for photovoltaic cells and arrays. We intend to fill in the gap. Moreover, the Xcos or Simulink implementation of the Newton-Raphson method is missing from the current literature.

The paper is organized as follows: in section II, the single-diode photovoltaic cell current equations are given; in section III, the detailed modeling of the PV cell in Xcos is done; section IV presents the simulation results; finally, conclusions are drawn in section V.

II. SINGLE-DIODE PHOTOVOLTAIC CELL MODEL

The equivalent electrical model of a PV cell is presented in Figure 1. The photovoltaic cell acts just as a current source that is generating I_{ph} . The p-n junction of the cell is modeled with a diode, which is passed by the current noted I_D in the figure. Due to nonidealities in the p-n junction some electrons may flow over the junction; this is modeled by the shunt resistance R_{sh} . The connecting wires are

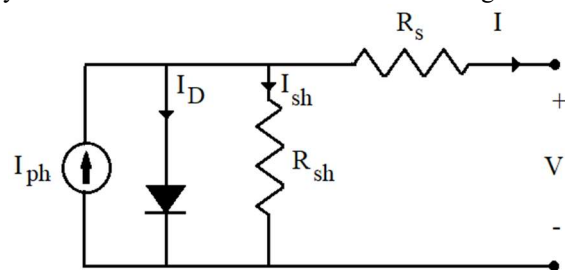


Figure 1. Equivalent circuit of a PV cell

modeled with a resistance connected in series R_s . Usually the value of R_{sh} is large whilst that of R_s is small. The current output is noted I and the voltage V [7].

A single PV cell generates between 0.5 and 0.8 volts depending of the semiconductor and technology used in its fabrication. Such values are too low to be of any use, so in practice between 36 and 72 cells are connected in series to form a PV module and these in turn are grouped in larger units called PV arrays. The equivalent electrical model of a PV array is shown in Figure 2 [7], [8].

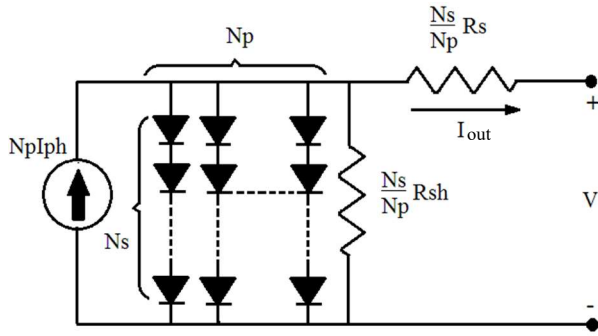


Figure 2. Equivalent circuit of a PV array

The current output [4] is determined with the help of equation (1):

$$I_{out} = N_p I_{ph} + N_p I_0 e^{\frac{V}{N_s} + \frac{I_{out} R_s}{N_p}} + I_{sh} \quad (1)$$

where I_{out} is the current output, N_p is the number of PV modules connected in parallel, I_{ph} is the photo-current, I_0 the saturation current, N_s the number of PV cells connected in series, R_s the series resistance, n the ideality factor of the diode, V_t the diode thermal voltage and I_{sh} the shunt current.

The diode thermal voltage [4] is calculated using equation (2):

$$V_t = \frac{kT}{q} \quad (2)$$

where V_t is the thermal voltage of the diode, k is the Boltzmann's constant ($k = 1.3805 \times 10^{-23}$ J/K), T is the operating temperature and q is the electron charge ($q = 1.6 \times 10^{-19}$ C).

The shunt current [4] is expressed in equation (3):

$$I_{sh} = \frac{\frac{V}{N_s} + \frac{I_{out} R_s}{N_p}}{R_{sh}}, \quad (3)$$

where I_{sh} is the shunt current, V is the voltage, R_{sh} is the shunt resistance.

The photo-current [4] is expressed in equation (4):

$$I_{ph} = [I_{sc} + K_i(T - 298)] \frac{I_r}{1000} \quad (4)$$

where I_{ph} is the photo-current, I_{sc} is the short circuit current, K_i is the short-circuit current of the cell at 25°C (298.15 K) and 1000 W/m², T is the operating temperature and I_r is the

solar irradiation (W/m²).

The saturation current [4] of the diode is given in equation (5), formulated below:

$$I_0 = I_{rs} \left(\frac{T}{T_r} \right)^3 e^{\left[q E_{g0} \left(\frac{1}{T} - \frac{1}{T_r} \right) \frac{1}{nk} \right]} \quad (5)$$

with I_{rs} the reverse saturation current, T_r the nominal temperature ($T_r = 298.15$ K), E_{g0} the band gap energy of the semiconductor ($E_{g0} = 1.1$ eV) and n the ideality factor of the diode.

The reverse saturation current [4] is conveyed below, in equation (6):

$$I_{rs} = \frac{I_{sc}}{e^{\frac{q V_{oc}}{N_s k n T} - 1}} \quad (6)$$

where I_{sc} the short circuit current and V_{oc} the open circuit voltage.

A key step in finding the current output I_{out} is the use of a numerical solver. In this work Newton-Raphson's method was graphically implemented. This method is used for finding successively better approximations to the roots of a real-valued function, as seen in equation (7), where the unknown is the output current of the PV cell I_{out} :

$$f(I_{out}) = I_{out} - N_p I_{ph} - N_p I_0 e^{\frac{V}{N_s} + \frac{I_{out} R_s}{N_p}} - I_{sh} \quad (7)$$

As input of the method we have the function $f(I_{out})$, its first derivative $f'(I_{out})$ and an arbitrary value I_n for a root of the function. A better approximation of the root I_{n+1} is shown in equation (8):

$$I_{n+1} = I_n - \frac{f(I_n)}{f'(I_n)} \quad (8)$$

In this implementation, a constant iteration number (50 iterations) is used. For The first derivative of $f'(I)$, is shown in equation (9):

$$f'(I_{out}) = I_0 N_p \left(\frac{R_l}{N_s} + \frac{R_s}{N_p} \right) e^{\frac{\frac{I_{out} R_l}{N_s} + \frac{I_{out} R_s}{N_p}}{n V_t}} + \frac{N_p R_l + R_s}{R_{sh}} + 1 \quad (9)$$

III. THE NUMERICAL MODELING OF CASCADED PV CELLS

Scilab [10] is a free, open-source simulation environment with a graphical interface, used for numerical computation; it provides a computation environment for engineering and scientific applications. The software includes hundreds of math functions and uses a programming language that allows access to advanced data structures, 2D and 3D graphical functions. Xcos is a graphic editor that allows the design of dynamic hybrid systems, has been used to draw and simulate the schematics. Using the editor, models can be designed, uploaded, saved, compiled and simulated.

The models designed in Xcos are based on the

equations expressed in the previous section. In this section, the models used for a simulation of a PV cell that is exposed to a constant solar irradiation and a temperature that varies over time are presented.

The sub-system in Figure 3 was obtained from equation (3). I_{out} , R_l , N_p , N_s , R_s and R_{sh} are input ports that must be used inside an Xcos Super Block (sub-system) and need to be connected to inputs in the sub-system. These connectors are a signal link from outside of a system into a sub-system. I_{sh} is an output port (labeled 1) that must be connected to an output of a block.

The sub-system in Figure 4 was created using equation (2). The input and the output of this sub-system is the current temperature T , respectively the thermal voltage V_t . The Boltzmann constant k and the electron charge q are values implemented with constant blocks.

The sub-system in Figure 5 was created to compute the photovoltaic current I_{ph} from equation (4). The sub-system inputs are the ambient temperature T and illumination I_r .

Equation (5) is implemented in the Xcos model presented in Figure 6. T is an input port and I_0 (labeled 1) is an output one. Constant blocks were used for q , E_{g0} , n , k and Tr . The computation of the reverse saturation current I_{rs} is achieved by the instantiation of the sub-system presented in Figure 7.

The sub-system in Figure 7 is used to compute the reverse saturation current I_{rs} (equation (6)). The sole variable in eq. (6) is the ambient temperature T , while the other quantities are all constants. Constant blocks were used for q , N_s , k , n and I_{sc} . V_{oc} is dependent of the number of serially connected cells N_s and is calculated as 0.6 V multiplied by the value of N_s .

The model in Figure 8 was designed with the help of equation (7). The role of this block is to evaluate the function $f(I)$ for a given value of I_n in each iteration. The inputs of this sub-system are considered I , N_p , N_s , I_{ph} , R_s , R_l , R_{sh} , n , V_t , I_0 . The computation of the shunt current I_{sh} is achieved by the instantiation of the sub-system form Figure 3.

The first derivate $f'(I)$ is evaluated for each I_n value by the model in Figure 9, which implements equation (9). I_0 , N_p , R_l , N_s , R_s , n , V_t , R_{sh} and I_{out} are input of the sub-system and the return value $f'(I_n)$ is presented on the output port.

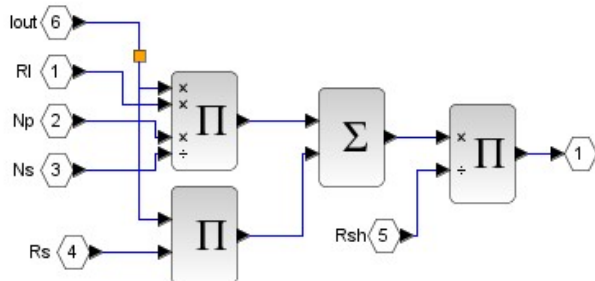


Figure 3. Xcos model of equation (3) – the computation of the shunt current I_{sh}

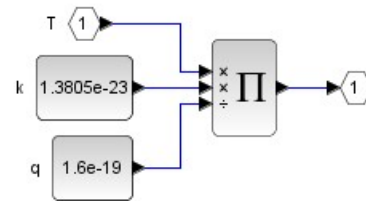


Figure 4. Xcos model of equation (3) – computing the thermal voltage V_t

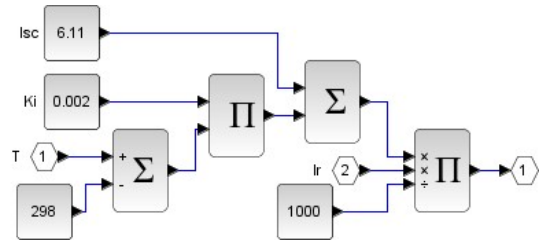


Figure 5. Xcos model of equation (4) – the computation of the photovoltaic current I_{ph}

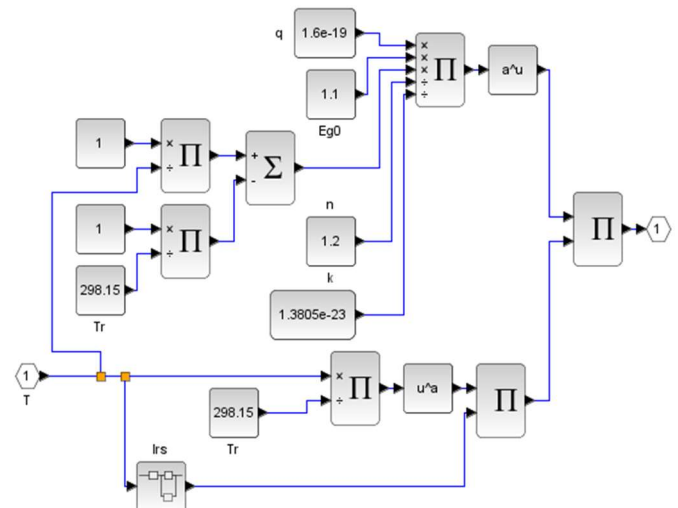


Figure 6. Xcos model of equation (5) – the computation of the saturation current I_0

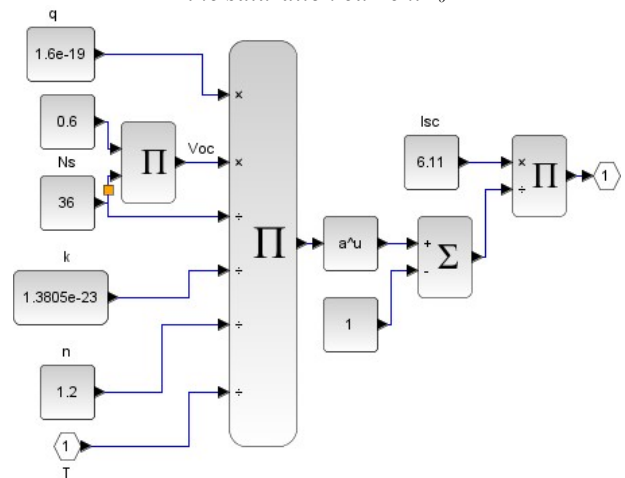


Figure 7. Xcos model of equation (6) – the computation of reverse saturation current I_{rs}

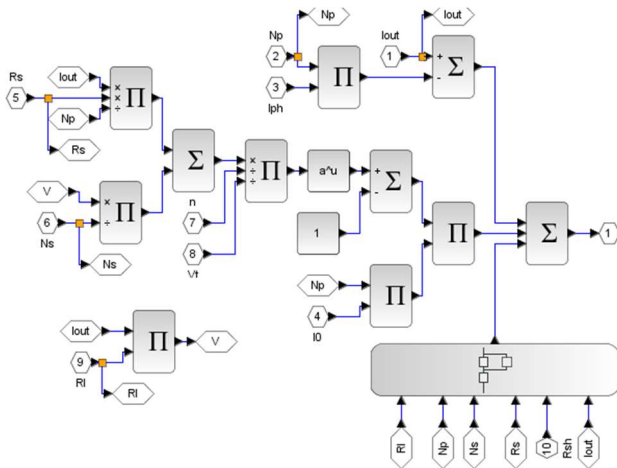


Figure 8. Xcos model for function $f(I)$ – the implementation of equation (7)

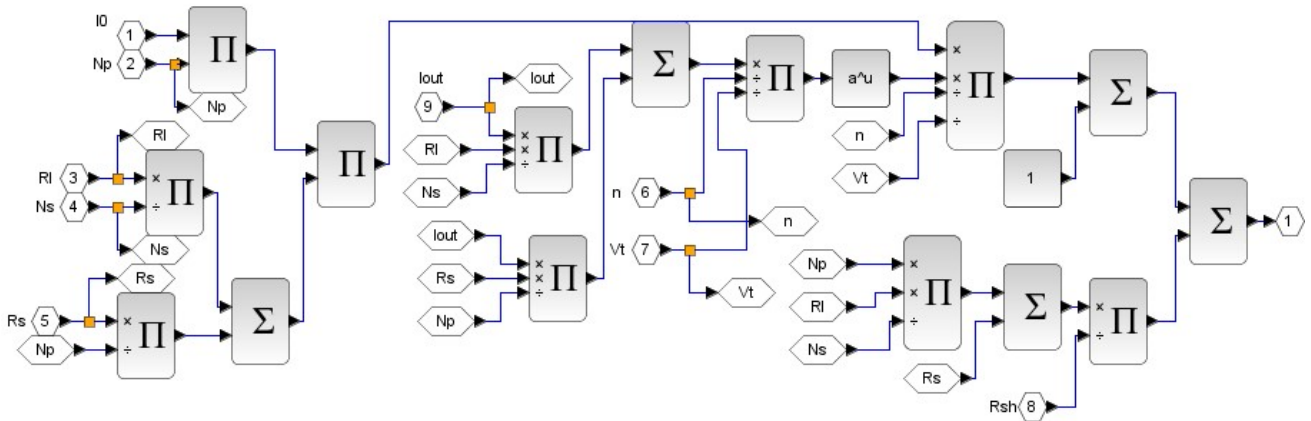


Figure 9. Xcos model for the evaluation for the first derivative, $f'(I)$

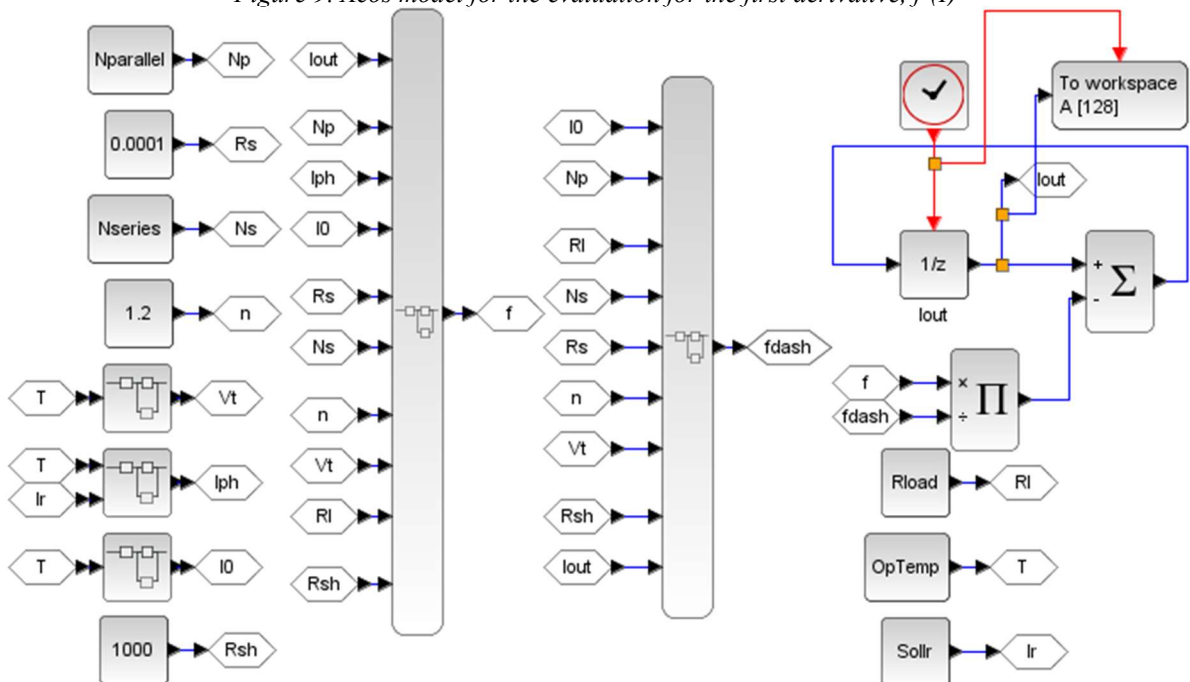


Figure 10. Xcos model for Newton-Raphson's method

As we wanted to plot current vs. voltage and the power vs. voltage characteristics, with respect to different ambient temperature and irradiation, the inputs of the model are the temperature T , the solar irradiation I_r and the load R_l connected to the cell, while the output is the PV cell current I_n . The solar cell parameters are listed in Table 1. The thermal voltage V_t , the photovoltaic current I_{ph} and the saturation current I_0 are computed with the aid of the sub-systems presented in Figure 4, 5, respectively 6. The iteration process described in equation (8) is implemented in this model. The evaluation of $f(I_n)$ and $f'(I_n)$ is achieved by instantiating the sub-systems from Figure 8, respectively Figure 9. The updated value of I_n is the quotient of $f(I_n)$ divided by $f'(I_n)$. The iteration process must be repeated for approximately 50 times (as observed, this value is enough to achieve a reasonably good approximation of the PV cell current).

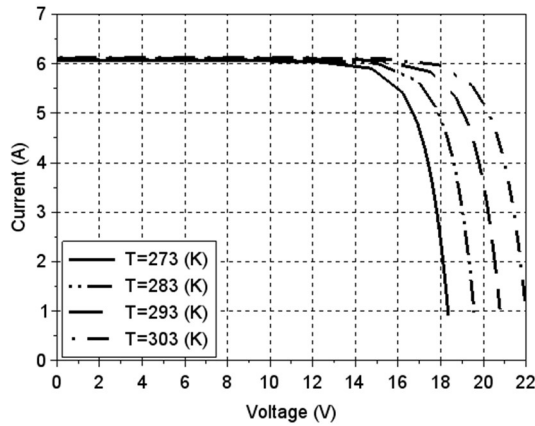


Figure 11. I-V curves for varying temperature

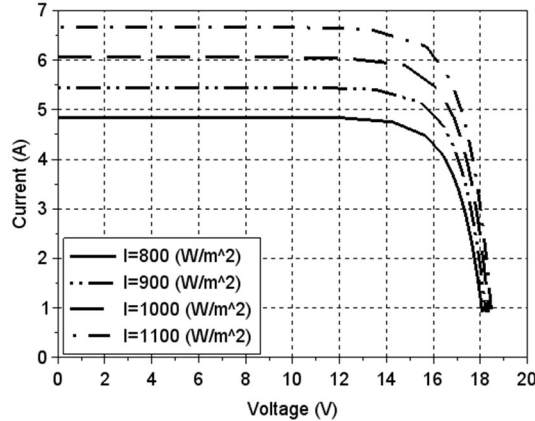


Figure 13. I-V curves for varying solar irradiation

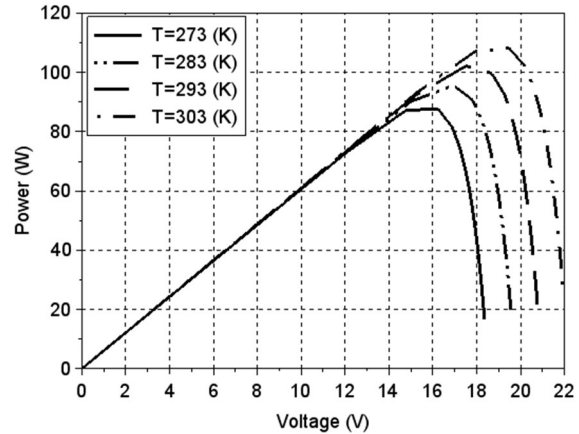


Figure 12. P-V curves for varying temperature

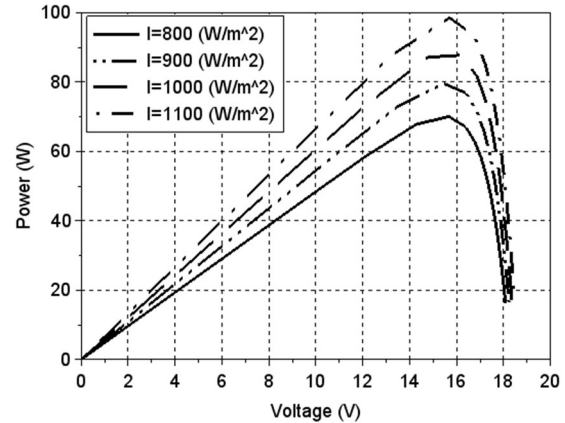


Figure 14. P-V curves for varying solar irradiation

IV. SIMULATION RESULTS

The proposed model was conceived to be configurable from the SciLab workspace. The configuration parameters are the PV cell load R_l (configured by the $Rload$ variable in the constant block in Fig. 10), ambient temperature T ($OpTemp$ variable), solar irradiation I_r ($Sollr$ variable), serially connected number of cells N_s ($Nseries$ variable) and parallelly connected number of cell strings N_p ($Nparallel$ variable). At the end of each simulation run the value of the output current is saved into the workspace. The simulations were achieved with the aid of Scilab scripts, where the parameters were set, the simulation was run and the results were saved. Using this methodology, the I-V and P-V characteristics can be plotted by a script that sweeps the value of the connected load R_l and saves the corresponding currents.

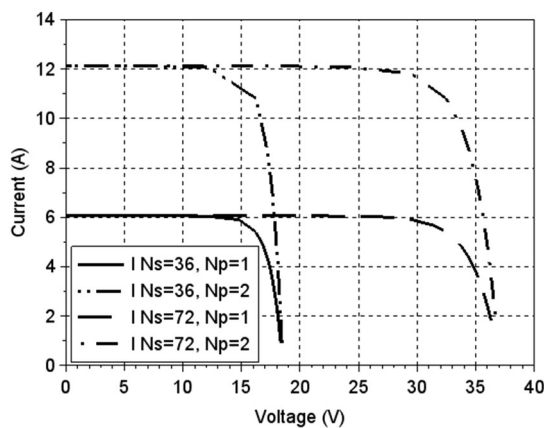
In the first numerical experiment, the variation of the I-V and P-V curves with respect to ambient temperature was studied. The I-V curves in Figure 11 were obtained with four different temperature values: 273, 283, 293 and 303 K. Solar irradiation has a constant value of 1000 W/m², $N_s=36$ and $N_p=1$. As anticipated the increase of the ambient temperature will increase the open circuit voltage V_{oc} . With the same configuration, the P-V curves in Figure 12 were obtained. As expected, with the rise of the temperature the maximum output power is increasing.

In the second numerical experiment, the variation of the I-V and P-V curves with respect to solar irradiation was studied. The I-V plot in Figure 13 were obtained for four

different solar irradiation values: 800, 900, 1000 and 1100 W/m².

TABLE 1. Parameters used for simulations

Parameter	Sign	Value
Short circuit current	Isc	6.11 A
Short-circuit current of cell at 25 °C and 1000 W/m ²	Ki	0.002 A
Electron charge	q	1.6e-19 C
Number of PV cells connected in series	Ns	1, 36 or 72
Number of PV modules connected in parallel	Np	1 or 2
Open-circuit voltage	Voc	0.6*Ns V
Ideality factor of the diode	n	1.2
Boltzmann's constant	k	1.3805e-23 J*K ⁻¹
Nominal temperature	Tr	298.15 K
Band-gap energy of the semiconductor	Eg0	1.1 eV
Series resistance	Rs	0.0001 Ω
Shunt resistance	Rsh	1000 Ω
Operating temperature	T	273 – 303 K
Solar irradiation	Ir	800 – 1100 W/m ²
Load resistance	Rl	0 – 21 Ω or 0 – 0.3 Ω

Figure 15. I-V curves for varying N_s and N_p

The temperature has a constant value of 273 K, $N_s=36$ and $N_p=1$. With the increase of the solar irradiation the short circuit current is increasing, just as expected. The P-V curves in Figure 14 were obtained with the same configuration. The maximum output power is increasing with the solar irradiation, just as anticipated by theory.

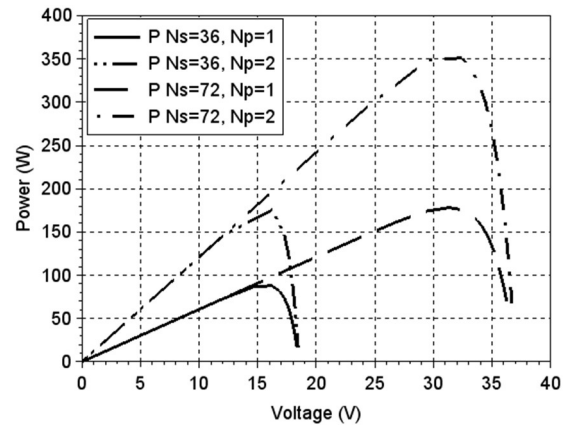
The third numerical experiment reveals the I-V and P-V curves with four N_s and N_p combinations. The different values of the serially connected cells and parallelly connected strings mimic different solar panel sizes. The I-V and P-V curves in Figures 15, respectively 16, were obtained for the next couples: $N_s=36$ and $N_p=1$; $N_s=36$ and $N_p=2$; $N_s=72$ and $N_p=1$; $N_s=72$ and $N_p=2$. The temperature and solar irradiation have constant values, 273 K and 1000 W/m². As expected, when the number of serially connected cells N_s is doubled, the open circuit voltage V_{oc} is approximately doubled, while the short circuit current I_{sc} remains the same. Doubling the number of parallelly connected cell strings will double the short circuit current I_{sc} , but the open circuit voltage V_{oc} is unchanged.

V. CONCLUSIONS

This paper provides a beginner-friendly step-by-step guide to PV modeling with easy to understand icons and offers an in-depth look at the Xcos blocks used. The paper also explains the Newton-Raphson method and how it was used for plotting the P-V and I-V curves.

The current literature is abundant in numerical simulation models created in commercial tools as Simulink, but the Simulink implementation of the Newton-Raphson method is superficially described (or not at all). In this paper, a detailed numerical simulation model is given, which is implemented in a free simulation environment, namely the SciLab's Xcos. Based on the analytical expressions used for modeling the PV cells, the corresponding Xcos schematics (sub-systems) were created. Detailed description of each schematic is given in the paper. A contribution is the graphical implementation of the Newton-Raphson method that is applied for solving the current equation of a PV cell (and cascaded PV cells). The obtained model was conceived to be run in SciLab scripts, thus it is configurable from the SciLab workspace.

With the created Xcos models, numerical experiments were carried out. In each experiment the I-V and P-V characteristics were plotted. The first experiment studied the ambient temperature effect on the I-V and P-V plots.

Figure 16. P-V curves for varying N_s and N_p

As expected, with the increasing temperature, the open circuit voltage is increasing. The second experiment reveals that the increasing solar irradiation will increase the short circuit current, just as expected. The last experiment shows the effect of different numbers of cells connected in series or cell-strings connected in parallel. This experiment mimics the behavior of several PV panel sizes.

Further work will comprise the implementation of an alternative solver, instead of the Newton-Raphson method. Some promising possibilities: polynomial curve fitting, Lambert-W function (omega function), particle swarm optimization and pattern search optimization.

REFERENCES

- [1] J. A. Gow and C. D. Manning, "Development of a photovoltaic array model for use in power-electronics simulation studies," in *IEE Proceedings - Electric Power Applications*, vol. 146, no. 2, pp. 193-200, Mar 1999
- [2] G. Walker, "Evaluating MPPT converter topologies using a Matlab PV model," in *Australian Journal of Electrical & Electronics Engineering*, Vol. 21, no. 11, pp. 49-55, 2001
- [3] F. Gonzalez-Longatt, "Model of Photovoltaic Module in Matlab™", 2DO Congreso Iberoamericano De Estudiantes De Ingeniería Eléctrica, Electrónica Y Computación, II CIBELEC 2005
- [4] X. H. Nguyen, and M. P. Nguyen, "Mathematical modeling of photovoltaic cell/module/arrays with tags in Matlab/Simulink," in *Environmental Systems Research*, 2015, paper 10.1186
- [5] M. Babar, E.A. Al-Ammar, N. H. Malik, "Numerical Simulation Model of Multijunction Solar Cell", *Journal of Energy Technologies and Policy*, Vol.2, No.7, 2012
- [6] F. Dimroth et al., "Four-Junction Wafer-Bonded Concentrator Solar Cells," in *IEEE Journal of Photovoltaics*, vol. 6, no. 1, pp. 343-349, Jan. 2016.
- [7] T. Salmi, M. Bouzguenda, A. Gastli, and A. Masmoudi, "MATLAB/Simulink Based Modelling of Solar Photovoltaic Cell," in *International Journal of Renewable Energy Research*, 2012
- [8] H.-L. Tsai, C.-S. Tu, and Y.-J. Su, "Development of Generalized Photovoltaic Model Using MATLAB/SIMULINK", in *Proceedings of the World Congress on Engineering and Computer Science*, 2008
- [9] Wikipedia – The Free Encyclopedia, Newton's method, https://en.wikipedia.org/wiki/Newton%27s_method, 2017
- [10] The Scilab website, www.scilab.org, 2017