INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2022, VOL. 68, NO. 4, PP. 863-866
Manuscript received July 17, 2022; revised November 2022. DOI: 10.24425/ijet.2022.143895

Application of a Control Algorithm for the Master Unit of a Distributed Photovoltaic System

Mariusz Świderski*, and Amadeusz Gąsiorek

Abstract—**The presented distributed photovoltaic system is made of divided into individual modules photovoltaic panel, consisting of several photovoltaic cells properly connected and coupling them with low-power DC / DC converters. The essence of the research is to increase the reliability of the system and the resultant efficiency of the entire system, so that it is possible to convert solar radiation energy into electricity with the greatest efficiency. The article focuses on the presentation of the implementation and tests of the overriding control algorithm, the task of which is to provide full functionality for**

a distributed photovoltaic system. The control is designed to minimize the negative effects of shadows on the operation of the photovoltaic system and conduct self-diagnostics. The conclusion for the carried out work is the formulation of hardware and interface requirements for the further development of the project.

Keywords—**photovoltaic cell; photovoltaic systems; solar energy; MPPT algorithm; DC/DC converters; shading**

I. Introduction

PHOTOVOLTAIC installations have become a permanent part of our landscape. Both small prosumer solutions located in cities, towns and villages, as well as large-scale farms located in agricultural or investment areas constitute the infrastructure in question. Regardless of the scale, in basic terms, a photovoltaic installation is built of appropriately interconnected photovoltaic panels, a grid inverter and a protection unit, as well as the necessary mechanical structure [1]. More advanced solutions use microinverters instead of a single inverter. On the DC side, they are connected to a panel or several panels [1, 2] and on the AC side, they are connected to the power grid through a protection unit. An indirect solution is the use of optimizers, which, in case of a basic installation with one inverter, are individually connected to individual panels [2, 3]. The above-mentioned solutions using microinverters or optimizers are mainly aimed at reducing the negative impact of the shading effect on the operation of the installation.

II. The Shading Phenomenon

The phenomenon of shading occurs when any surface of a photovoltaic panel is temporarily or permanently deprived of access to solar radiation [4]. The resulting shadows can be classified as permanent or temporary. Permanent shading results from the close proximity of large surface objects that cast a shadow over the panel surface and are, from the panel's point of view, independent of the sun's movements. These objects include: parts of buildings, e.g. roofs, chimneys, air intakes and launchers, and natural objects, e.g. trees, terrain elevations, etc. Temporary (periodic) shadows directly depend on the movement of shadow-producing objects and the sun itself. For example, temporary shadows include: snow, leaves, bird droppings, dust, soot, as well as small antenna masts or elements of the lightning protection system [2, 4].

Shading can be parameterized by its duration, which informs how long a given panel area is shaded. In addition to the duration, another parameter that characterizes the shading is the percentage of radiation that has been absorbed by objects located near the panel [2, 4].

Shading is one of the most undesirable phenomena in photovoltaic installations due to the drastic decrease in energy generation because the system adjusts its work to its weakest link. This drop is disproportionate to the shaded area. Moreover, the shading causes the panel to heat up locally, which in turn leads to a further decrease in its energy efficiency, in extreme situations these hot spots can cause a fire in the installation.

III. Distributed Photovoltaic Sysyem

The general concept of a solar energy generation system (power plant), based on distributed photovoltaic panels, is shown in Figure 1.

The system is divided into the following blocks: N low voltage photovoltaic modules (PVPN), N low power DC / DC converters (CNVN), each converter is connected to a single PVPN module, a common energy storage (EC) - based on LiION technology, inverter (GCN), which enables the transmission of energy to the power grid and the Global Control Block (GCB).

The CNVN block primarily contains a flyback converter equipped with a local controller (LCT). The LCT block monitors: PVPN output voltage (u_{PV}), PVPN output current (i_{PV}) and module temperature (T_{PV}). Based on these quantities, the LCT implements, to a certain extent, the Maximum Power Point Tracking (MPPT) algorithm. Local control systems of individual converters cooperate with the GCB block via a global, isolated, data (control) bus.

The authors are from the Faculty of Control, Robotics and Electrical Engineering, Poznan University of Technology, Poland (e-mail: mariusz.swiderski@put.poznan.pl, amadeusz.gasiorek@put.poznan.pl).

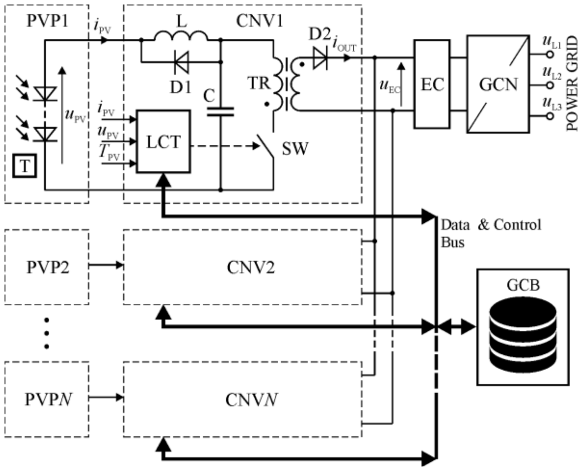


Fig.1. Block diagram of a solar power plant with a distributed photovoltaic panels system [5]

IV. DEVELOPED ALGORITHM

The developed algorithm for the GCB block is executed cyclically. In operation, the GCB controls all CNVN converters, namely the LTC blocks. Based on the module list generated during the global controller configuration, each local module can perform a number of operations, one of which is carrying out measurements, which are essentially the full characteristic of the PVPN connected to CNVN. Based on the data received from the LTC module, the proprietary algorithm is implemented, the task of which is to find the most advantageous MPP point. The maximum power point (MPP) data is then sent to the LCT controller which in the vicinity of this point locally applies the perturb and observe algorithm. Diagnostics is an additional functionality available during the communication between GCB and LTC. The developed algorithm is able to assess whether the PVPN operates in accordance with the assumed characteristics. Communication with individual LTCs and calling the algorithm are shown in Figure 2.

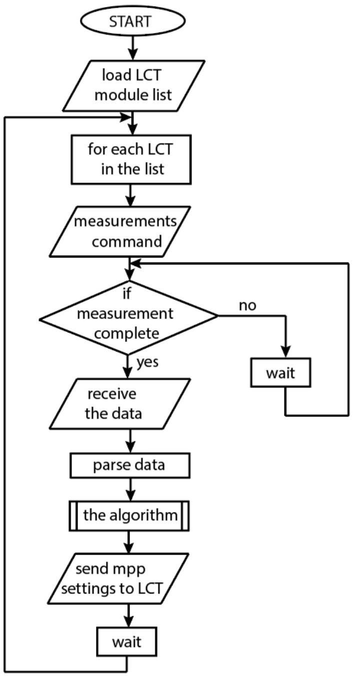


Fig.2. Block diagram of calling the main algorithm

The heart of the developed solution is a system service currently running under the control of the Windows family system. The service itself is written in the high-level language Python [6]. This language is well suited to handling large amounts of data. Thanks to specialized libraries, obtaining advanced solutions is faster than using other languages. The libraries used in the developed solution are: numpy - used mainly for operations on arrays [7], scipy, and more precisely the signal module - used for signal processing [8] and matplotlib included for the purposes of presenting the results in the paper [9].

The algorithm can be divided into two parts: diagnostic and MPP determination. The input data in both parts are two vectors received by the GCB from the LTC: Uarr - voltage vector for the selected PVP and Iarr - current vector for the same PVP. Both of these vectors are 20-element (number freely configurable by the user) and are used to determine the $I = f(U)$ characteristic for each of the PVP modules. Additionally, for the diagnostic part, the algorithm defines an experimentally selected Earr array, which defines the maximum positive and negative deviations of the $I = f(U)$ characteristic from the adopted pattern.

This array is used for the diagnostic. Based on the Uarr and Iarr arrays, the power (Parr) is calculated and stored in the new table $P = f(U)$. Next, the condition is checked whether all elements of the Iarr array fall within the designated band determined by the Earr table. If an element of the array does not fit, an index is added to the Rf table - containing information about deviations from the rated parameters of the current-voltage characteristic of the PVP panel. Finally, when the sum of all indices in Rf is greater than zero, a diagnostic message is displayed. The operation of the diagnostic algorithm is shown in Figure 3.

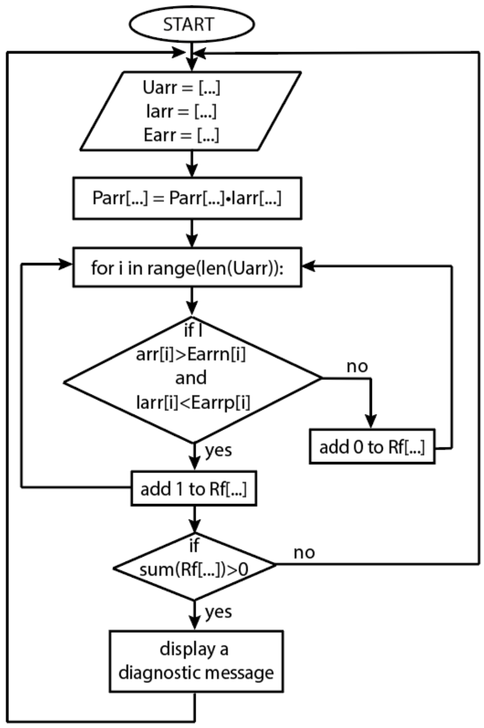


Fig.3. Block diagram of the diagnostic algorithm

MPP is determined using the Parr array by applying the find_peaks [10] function on it. This function works in such a way that for any numpy array it finds all indices for which the values is maximal in that array. In a situation when the function returns a single index, it is automatically a MPP, if it turns out that there are more indices, then for each of the indices the power is compared and the index with the highest power is selected. The presence of two or more peaks is indicated by the diagnostic function. To send the inverter settings to the LTC, instead of operating on power, the index is converted to the current and voltage pair at the MPP. As a result, the LTC module does not have to perform unnecessary math operations. The operation of the algorithm that determines the MPP is shown in Figure 4.

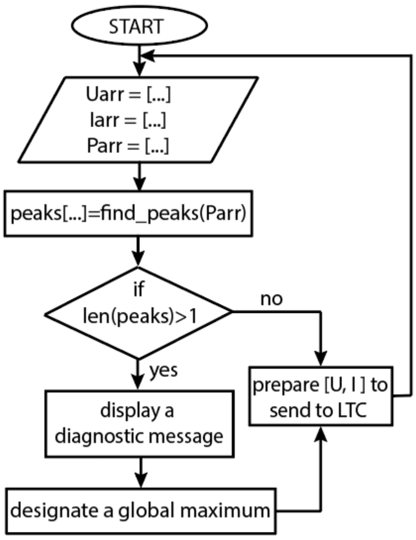


Fig.4. Block diagram of the algorithm determining the MPP

V. THE ALGORITHM TEST

In order to verify the correctness of the algorithm's operation, an experimental setup was prepared, consisting of 36 photovoltaic cells. It is shown in Figure 5. Each cell had dimensions of $52x52mm$ and the following values of the main rated parameters:

- short-circuit current: $I_{SC} = 0.86$ A,
- open circuit current: $U_{OC} = 0.50$ V.

Based on the previously performed tests [11], the cells were connected to obtain $U_{PV} = 9$ V i $I_{PV} = 1.72$ A which translates into a maximum power of $P_{PV} = 15.5$ W.

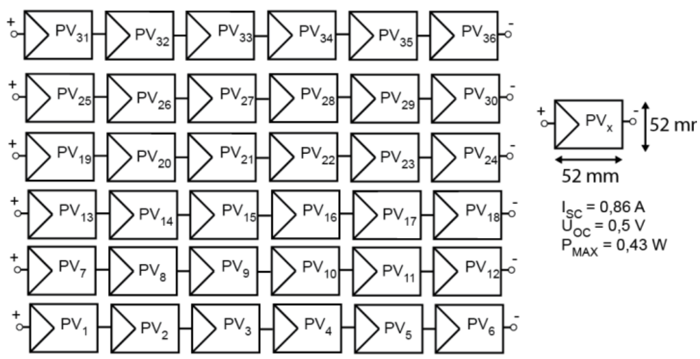


Fig.5. Configuration of the experimental setup of cells that form the PV panel [11]

A system of interconnected PV cells prepared in this way forms the PVPN module. Using the experimental setup, three cases were generated: model - without shading (case 1), with slight shading (case 2) and with shading resulting in a few local maxima on the $I = f(U)$ characteristic (case 3).

As described in chapter 4, the algorithm starts the analysis after receiving data from LTC modules, diagnostics is performed first, it is checked whether the parameters of each PVP module are within the given band. For case 3, it can be observed that at a voltage of approx. $7\,V$ there appears a deviation, which is shown in Figure 6.

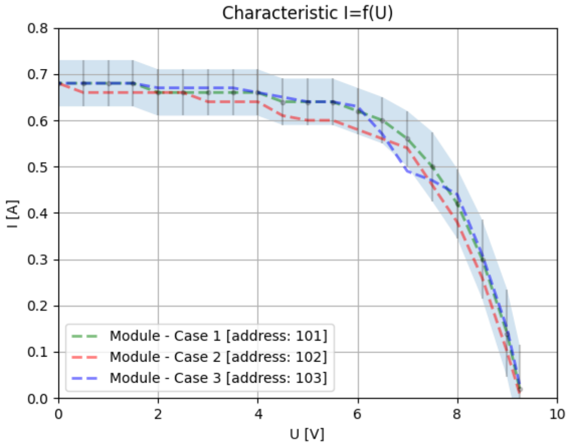


Fig.6. Graphical representation of the effect of the diagnostic algorithm

Such a deviation is immediately signaled in the console window with the message shown in Figure 7.

```
alg.py [C:\Program Files\Python\Python38\python.exe]
Diagnostics:
Check: Case 3 [address: 103]
```

Fig.7. Diagnostic message

The next stage of the algorithm's operation is to determine the MPP, it is graphically presented in Figure 8. In this part it can be observed that case 3 is characterized by two maxima (small x signs on the characteristic), and cases 1 and 2 have only one maximum. Hence, the global maximum search procedure was started.

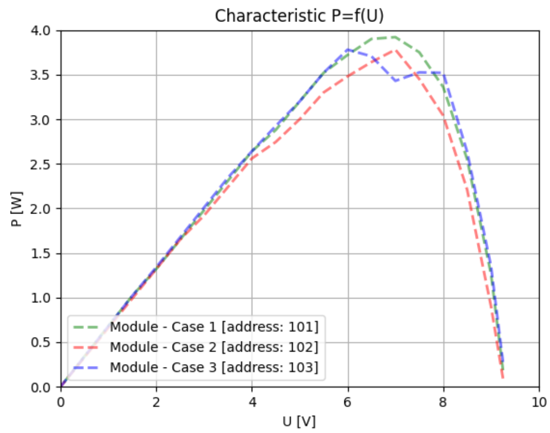


Fig.8. Graphical representation of the MPP determination algorithm

The result of this procedure is to display a message in the console window for case 3 with many maxima and display the best. The message is shown in Figure 9.

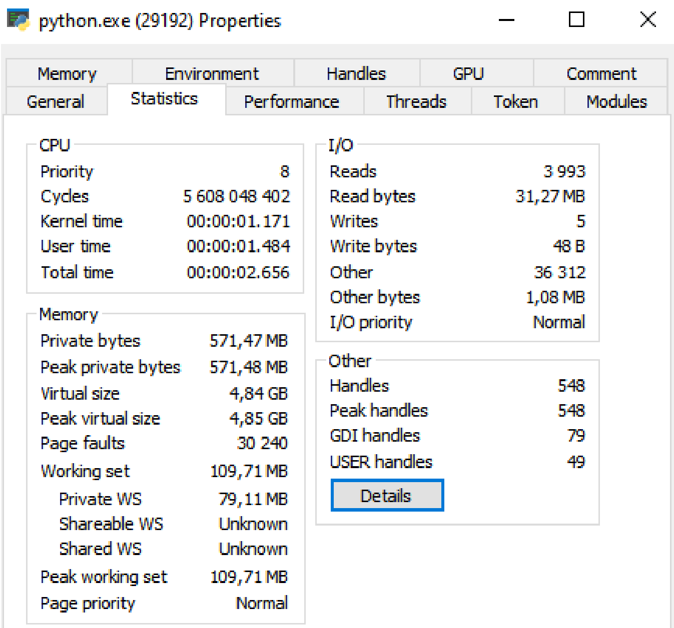
```
<terminated> alg.py [C:\Program Files\Python\Python38\python.exe]
Diagnostics:
Check: Case 3 [address: 103]

MPPT:
The MPP voltage for Module - Case 1 [address: 101] is [7.]
The MPP current  for Module - Case 1 [address: 101] is: [0.56]
The MPP voltage for Module - Case 1 [address: 101] is [7.]
The MPP current  for Module - Case 1 [address: 101] is: [0.56]
Multiple Max Points: Module - Case 3 [address: 103]
The MPP voltage for Module - Case 3 [address: 103] is: 6.0
The MPP current  for Module - Case 3 [address: 103] is: 0.63
```

Fig.9. Diagnostic message for MPP

Of course, drawing the characteristics has been added for the purpose of checking and illustrating the results of the work in the article. The target algorithm without debugging not only does not display this data, but does not calculate it at all. The results of the configured algorithm are only diagnostic messages and preparation of [U, I] pairs which are returned to the LTC.

As part of the tests, it was checked what resources such a prepared Windows service requires. The results of resource monitoring are shown in Figure 10.

The presented data shows that the algorithm with the functions of drawing characteristics for the calculation of three modules and communication used approx. 517 MB of operating memory, at the time of counting the processor load was approx. 12% and approx. 31 MB were transferred through the I/O ports. The algorithm was tested on the DELL PowerEdge R750xs device with the following configuration: Processor: Intel Xeon Gold 5315Y, RAM: 4x 64GB DDR4 RDIMM, RAID controller: PERC H755, Network card: ConnectX®-5 DP (OCP 3.0).



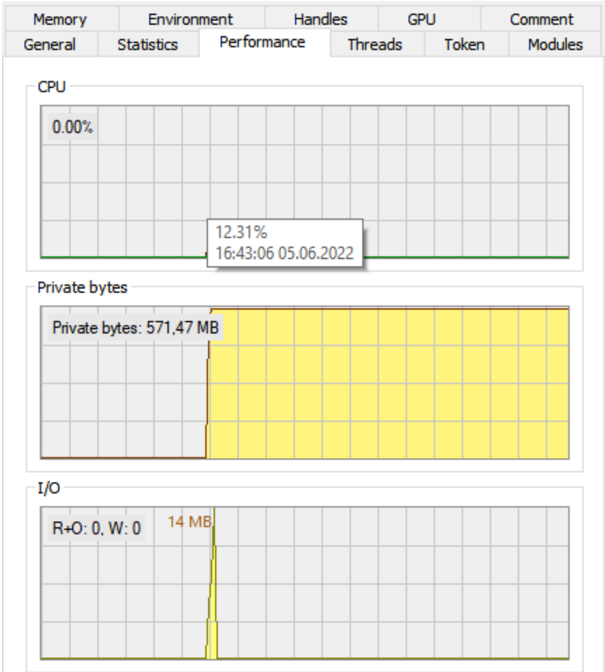


Fig.10. Presentation of used system resources needed to run the algorithm

CONCLUSIONS

The distributed system managed by the developed algorithm presented in the article perfectly copes with the identification of shadows and determining the correct MPP for such modules. An additional diagnostic function is extremely important to prevent possible future damage to the installation. The described system can work with any remote or local server which task is to run GCB. Meanwhile, the requirements for the hardware configuration of a device with a working distributed system control service can be met by any modern server unit or PC with a gaming configuration and appropriate cooling.

REFERENCES

[1] Ch. Roberts, "Guide to the Installation of Photovoltaic Systems," London, Microgeneration Certification Scheme, 2012, ISBN 978-0-9574827-0-8.
[2] R. Tytko, „Fotowoltaika Podręcznik dla studentów," uczniów, instalatorów, inwestorów w. 2022, 2022, wydanie VI, ISBN: 9788381112567.
[3] B. Szymański, „Instalacje Fotowoltaiczne Poradnik," 2021, ISBN: 978-83-65874-00-9.
[4] A. Luque, S. Hegedus, "Handbook of Photovoltaic Science and Engineering," England, Wiley, 2003, ISBN 0-471-49196-9.
[5] M. Gwóźdź, M. Świderski, "Solar Power Plant with Distributed System of PV Panels," Przegląd Elektrotechniczny, 2019, vol. 95 no. 2, pp. 55-58, https://doi.org/10.15199/48.2019.02.11
[6] "Welcome to Python.org" [Online]. Available: www.python.org. [Accessed: 04. Jun. 2022].
[7] "NumPy" [Online]. Available: numpy.org. [Accessed: 04. Jun. 2022].
[8] "SciPy" [Online]. Available: scipy.org. [Accessed: 04. Jun. 2022].
[9] "Matplotlib — Visualization with Python" [Online]. Available: matplotlib.org. [Accessed: 04. Jun. 2022].
[10] "scipy.signal.find_peaks — SciPy v1.8.1 Manual" [Online]. Available: docs.scipy.org/doc/scipy/reference/generated/scipy.signal.find_peaks.html [Accessed: 04. Jun. 2022].
[11] M. Gwóźdź, M. Świderski, "Influence of shading effect on operation of solar plant with system of distributed PV panels," Przegląd Elektrotechniczny, 2020, vol. 96 no. 7, pp. 76-79, https://doi.org/10.15199/48.2020.07.14