

Article

Trusted Simulation Using Proteus Model for a PV System: Test Case of an Improved HC MPPT Algorithm

Abdelilah Chalh ^{1,*}, Aboubakr El Hammoui ¹, Saad Motahhir ², Abdelaziz El Ghzizal ¹, Umashankar Subramaniam ³ and Aziz Derouich ¹

¹ Innovative Technologies Laboratory, EST, SMBA University, Fez 30000, Morocco; aboubakr.elhammoui@usmba.ac.ma (A.E.H.); abdelaziz.elghzizal@usmba.ac.ma (A.E.G.); aziz.derouich@usmba.ac.ma (A.D.)

² Engineering, Systems and Applications Laboratory, ENSA, SMBA University, Fez 30000, Morocco; saad.motahhir@usmba.ac.ma

³ Renewable Energy Lab, College of Engineering, Prince Sultan University, Riyadh 12435, Saudi Arabia; shankarums@gmail.com

* Correspondence: abdelilah.chalh@usmba.ac.ma

Received: 9 February 2020; Accepted: 10 March 2020; Published: 15 April 2020



Abstract: The real implementation of the maximum power point tracking (MPPT) controllers for the photovoltaic (PV) systems is still a big challenge for researchers working in this field. Often, they use simulation tools to assess the performance of their MPPT algorithms before actual implementation. In this context, this paper aims to propose a trusted simulation of a PV system designed under Proteus software. The proposed PV simulator can be used to verify and evaluate the performance of MPPT algorithms with a closer approximation to the real implementation. The main advantage of this model that it contains a real microcontroller, as can be found in reality, so that same code for the MPPT algorithm used in the simulation will be used in real implementation. In contrast, when using (Powersim Software) PSIM or Matlab/Simulink, the code of the algorithm must be rewritten once the real experiment begins, because these tools don't provide a microcontroller or an electronic board in which our algorithm can be implemented and tested in the same way as the real experiment. After this section, a modified Hill-Climbing (HC) algorithm is introduced. The proposed algorithm can avoid the drift problem posed by conventional HC under a fast variation in insolation. The simulation results show that this method presents good performance in terms of efficiency (99.21%) and response time (10 ms), which improved by 1.2% and 70 ms respectively compared to the conventional HC algorithm.

Keywords: Proteus; MPPT algorithm; PV Simulator; Drift problem

1. Introduction

Today, solar energy currently has taken a large part of the market compared with other sources of renewable energy [1–3]. This development has pushed many researchers to search for important solutions to increase the PV energy extracted from the PV panels. Among these solutions, we find MPPT techniques [4–7]. These techniques are used to control a DC-DC converter to extract the maximum power from the PV system. The role of the DC-DC converter is to provide impedance matching between the PV array and load. In addition, many works have investigated the effect of partial shading conditions (PSC) on the extracted PV energy, always with the aim of improving the PV system efficiency. As a result, several GMPPT techniques have been proposed in the literature to make PV systems working at the actual best efficiency under PSC [8–13]. On the other hand, to validate the

performance of these techniques, it is necessary to test them before implementation. For this reason, researchers often use simulation tools such as Matlab-Simulink or PSIM to model PV systems and then test the performance of their MPPT algorithms [14–16]. However, the main disadvantage of these tools is that they do not contain microcontrollers or embedded boards (such as Digital Signal Processor (DSP), Field-Programmable Gate Array FPGA, Arduino, Programmable Interface Controllers (PIC) . . .) in which the MPPT algorithm can be implemented and tested as it is done in an actual prototype. Therefore, once we start the actual implementation of the MPPT algorithm for a real PV system, several problems may occur due to the disparities that can appear between the software development and the requirements during the development process. Moreover, as all the components used (DC/DC converter, sensors, actuators and microcontroller) are real, unlike those designed in the simulation tool, it is difficult to know the component responsible in the event of a bug. That can increase the time spent in debugging runtime errors. Against, these problems can be avoided by using the Proteus tool as an alternative for modelization and simulation.

Proteus is an electronic circuit design software that includes diagrammatic capture, PROSPICE simulation, and printed circuit board (PCB) layout modules. It provides in its libraries embedded boards such as PIC and Arduino, where we can implement our controller by uploading the hex code to the microcontroller as it actually happens in reality. In addition, this tool has been developed to contain components or models that are close to reality, which can give results that are more accurate. Figure 1 shows a comparison between the steps of developing a circuit board in Proteus and in traditional design tools. From this figure, it can be noticed that when using traditional design tools (such as Matlab/Simulink or PSIM), the software development and testing system cannot begin until a PCB or physical prototype is available. Thereafter, if something is wrong with the hardware design, the entire process must be repeated. However, using Proteus, software development can begin as soon as the schematic is drawn, and the combination of hardware and software can be thoroughly tested before physical prototyping. In addition, the same algorithm's code implemented in the microcontroller on Proteus will be used in the real experiment. Whereas, using PSIM or Matlab/Simulink, we have to rewrite the code of the algorithm once we begin the real experiment. For these reasons, researchers and engineers can really benefit from Proteus to test and evaluate the performances of their MPPT algorithm before the real implementation.

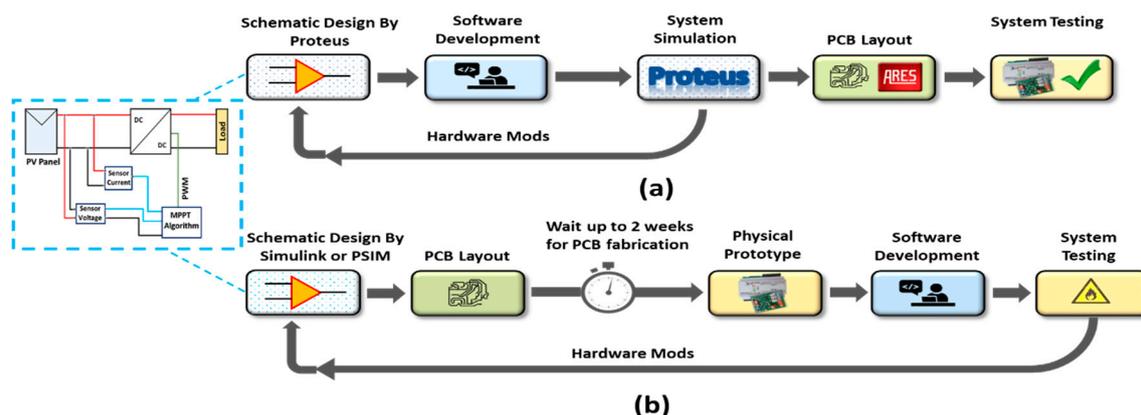


Figure 1. Steps to develop a circuit board in Proteus tool: (a) compared to those for developing it in traditional design tools, (b) such as Matlab/Simulink or PSIM.

However, to the authors' best knowledge, there is no simulation found in the literature of an entire PV system under Proteus software. While recently, some works have addressed the simulation of PV panel in Proteus as the first attempts in this subject. A single-diode Proteus model of PV module has been proposed and validated in our previous work [17–20]. Authors in [21] have been proposed a two-diode model of PV panel in Proteus. Against this background, this paper aims to present an entire

design of a PV system based on Proteus, which can be used as a simulator to test the performances of MPPT algorithms. This simulator is composed of a PV panel model, DC-DC converter, voltage and current sensors, graph analysis, a Liquid-Crystal Display (LCD) screen, and the Arduino UNO board in which the MPPT algorithm is implemented. The design aspects of the PV system under Proteus are extensively described in this paper. Moreover, a modified Hill-Climbing (HC) MPPT algorithm is proposed in this paper to avoid the drift problem posed by the conventional HC algorithm when under a fast variation in insolation.

The remainder of this paper is organized as follows. Section 2 presents the PV system design in Proteus and describes the analysis of the effect of the drift problem under a fast variation in insolation with the conventional and improved HC MPPT method. Section 3 discusses the simulation results obtained. Finally, Section 4 concludes with a summary and discussion of future works.

2. Materials and Methods

2.1. Description of a PV System

2.1.1. Description of the Entire PV System

Figure 2 shows the schematic diagram of the entire PV system on Proteus. The latter is composed of a PV panel, a DC-DC converter, an MPPT controller, and a load. The PV panel is connected to the load via the boost converter. The MPPT controller implemented in the Arduino board uses the voltage and PV current data to control the boost converter to reach the MPP. The code of this controller is developed firstly in the Arduino software (IDE), and then its hexadecimal code is uploaded to the board on Proteus.

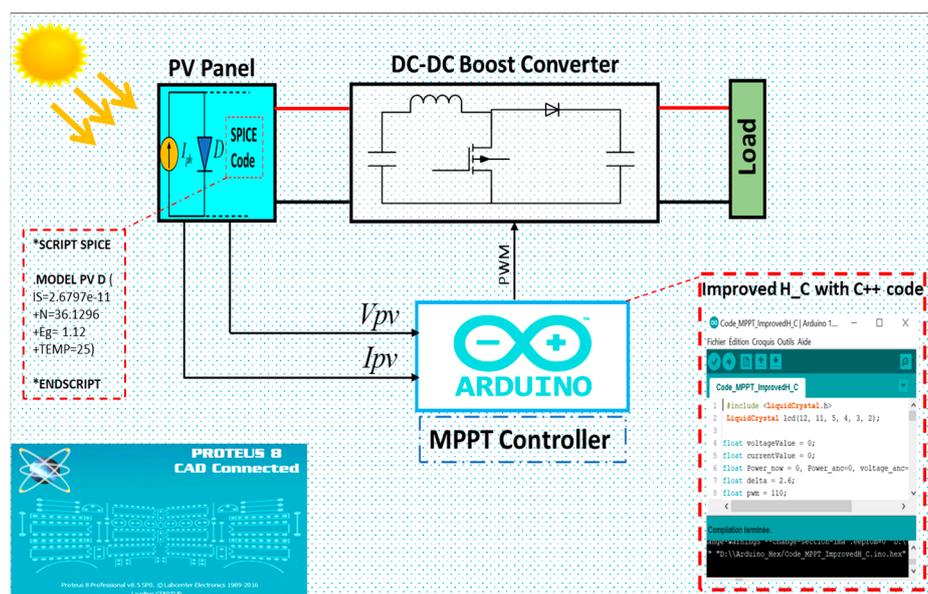


Figure 2. Schematic diagram of the entire photovoltaic (PV) system.

2.1.2. Proteus PV Panel Model

The equivalent circuit for the most used solar cell model consists of a diode and a current source connected in parallel, as well as a shunt resistor R_{sh} and a series resistor R_s , as shown in Figure 3. Based on this circuit diagram, the output current is given by the following expression [22]:

$$I = I_{ph} - I_{s,0} \left(\exp \left(\frac{V + IR_s}{aV_t} \right) - 1 \right) - \frac{V + IR_s}{R_{sh}} \quad (1)$$

where: I_{ph} : Represents the photocurrent of the solar cell (A); $I_{s,0}$: Represents the reverse saturation current of the solar cell (A); V : Represents the output cell voltage of the solar cell (V); a : Represents the diode ideality factor of the solar cell; V_t : Represents the thermal voltage of the solar cell.

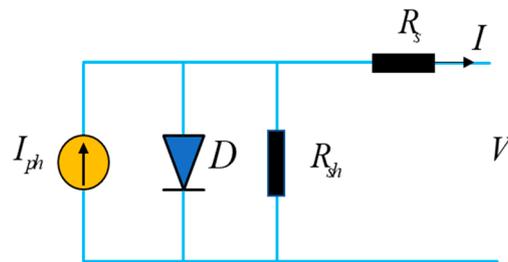


Figure 3. Equivalent circuit for single diode model of solar cell.

In this study, the TDC-M20-36 panel is used, and Table 1 presents its specifications. Figure 4 presents the Proteus PV panel model with the Spice code. In this latter, the parameters of the used diode (saturation current, number of cells, ideality factor, and bandgap energy) according to the TDC-M20-36 panel’s specification, must be introduced. Note that presented the PV panel model is already validated through simulation/ experimental results in our previous paper [17].

Table 1. Characteristics of the used PV model at Standard Test Conditions (STC).

TDC-M20-36	
Pmax of PV panel	20 W
Vmpp at Pmax of PV panel	18.76 V
Impp at Pmax of the PV panel	1.07 A
Current Isc at Short-circuit (SC)	1.17 A
Voltage Voc at Open-circuit (OC)	22.70 V
Temperature coefficient Kv at OC	−0.35%/°C
Temperature coefficient Ki at SC	−0.043%/°C
Number of cells	36

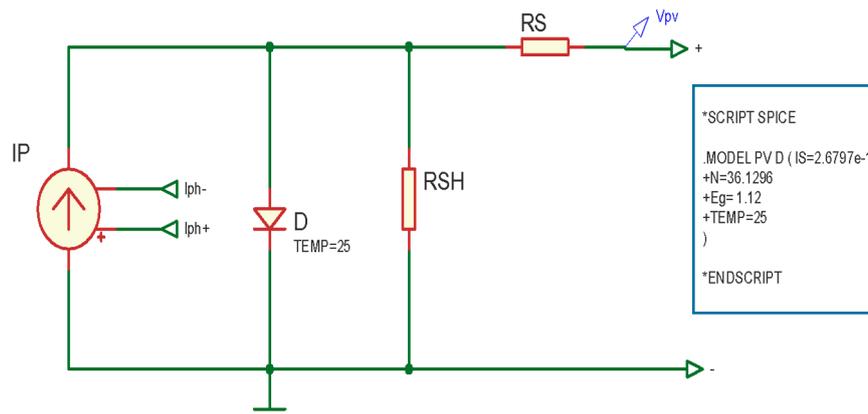


Figure 4. The equivalent circuit of panel model under Proteus.

2.1.3. DC-DC Boost Converter

The DC-DC converter is a principle element of the PV systems with an MPPT controller, which provides impedance matching between the PV array and load. Figure 5 presents the circuit diagram of the used DC-DC converter. Table 2 presents its principal parameters, which are the inductor

(L), input capacitor (C_{in}), output capacitor (C_{out}), and the switching frequency (f_s). In addition, Equations (2) and (3) define the relationship between its inputs and outputs.

$$V_0 = \frac{V}{1 - \alpha} \quad (2)$$

$$I_0 = I(1 - \alpha) \quad (3)$$

$$R_{eq} = \eta(1 - \alpha)^2 R_{load} \quad (4)$$

where: I_0 : Represent the output current of the Boost converter (A); I : Represent the output current of the PV model (A); V_0 : Represent the output voltage of the boost converter (V); V : Represent the output voltage of the PV model (V); R_{eq} : The equivalent resistance as seen by the PV module (Ω); R_{load} : The load resistance of the PV system (Ω); η : The efficiency of the boost converter (%); α : The duty cycle.

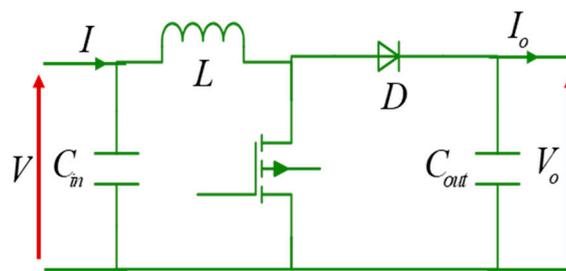


Figure 5. Circuit diagram of the DC-DC boost converter.

Table 2. The values used of the parameters of the DC-DC boost converter in our work.

Parameters	Value
L	20 mH
C_{in}	220 μ F
C_{out}	470 μ F
f_s	1 kHz

2.1.4. MPPT algorithm

The MPPT technique is a controller that varies the duty cycle of the DC-DC converter in order to extract the maximum power from PV panels. In this paper, a classical algorithm named the HC algorithm is used to make the MPPT control of the PV system. But, as reported in the literature, this algorithm suffers from drift problem when the fast variation in insolation [23,24]. Therefore, a modified HC algorithm is proposed to solve this problem.

Conventional HC MPPT algorithm

The conventional HC algorithm is based on the variation of PV power (dP) and PV voltage (dV) by considering the P-V characteristics curve. However, this MPPT algorithm suffers from drift problems that occur with the fast variation in insolation (rapid increase in insolation). Figure 6 presents the PV curves under a rapid increase in insolation. From this figure, it can be noted that when an increase in insolation at point 3, the operating point is set to a new point 4 ($dP = P_4(kTa) - P_3(kTa) > 0$ and $dV = V_4(kTa) - V_3(kTa) > 0$). Moreover, as shown in the flowchart of the conventional HC MPPT algorithm presented in Figure 7, the duty cycle decreases with increasing insolation, which involves moving point 4 to point 5 of the maximum power point (MPP) in the new curve, called the drift problem. Similarly, thing occurs for an increase of insolation on other points. So, A drift problem occurs when there is a rapid change in insolation due to cloudy days. This problem is due to the incorrect decision of the conventional HC MPPT algorithm when $dP/dP > 0$.

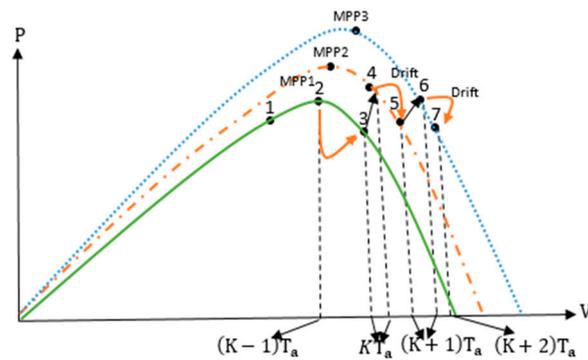


Figure 6. The P-V curves of the PV module under a rapid increase in insolation.

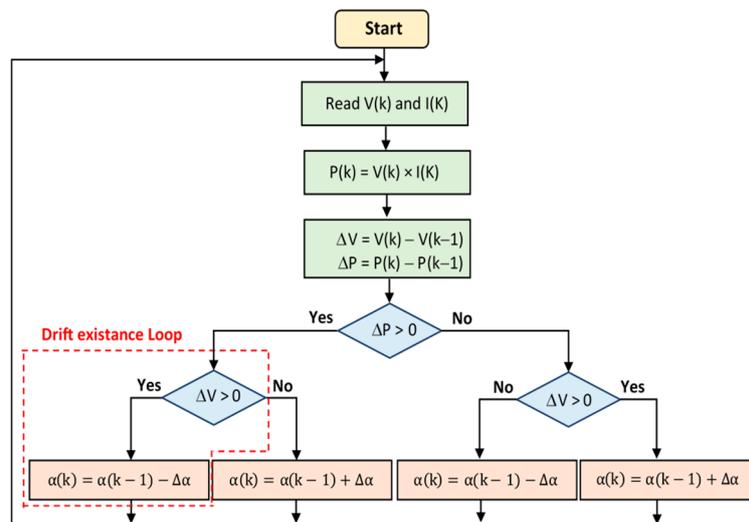


Figure 7. The flowchart of the conventional HC MPPT algorithm.

Improved HC MPPT algorithm

After detecting the loop in which the conventional HC algorithm makes a bad decision, a simple solution is proposed to overcome the obvious problem posed by HC algorithm.

The relationship between the PV current and voltage can be written as follows:

$$I = \frac{V}{R_{eq}} \tag{5}$$

With the value of the equivalent resistance is defined by Equation (4).

This relationship can also be expressed based on the single-diode model of the PV module as follows:

$$I = I_{ph} - I_0 \left(\exp\left(\frac{q(V + IR_s)}{\partial KTN_s}\right) - 1 \right) - \frac{(V + IR_s)}{R_{sh}} \tag{6}$$

By substituting Equation (5) into Equation (6), and by considering Taylor’s series expansion up to first order, Equation (7) can be expressed as follows:

$$\frac{V}{\eta(1 - \alpha)^2 R_{load}} = I_{ph} - \frac{I_0}{\partial KTN_s} \left(qV + \frac{R_s}{\eta(1 - \alpha)^2} V \right) - \frac{V}{R_{sh}} - \frac{R_s}{R_{sh}} \frac{V}{\eta(1 - \alpha)^2 R_{load}} \tag{7}$$

By simplifying Equation (7), the V can be expressed in terms of I_{ph} for an insolation G and the slope of the load line as follows:

$$V/G = \frac{I_{ph}/G}{\frac{1}{\eta R_{load}(1-\alpha)^2} \left(1 + \frac{R_s}{R_{sh}}\right) + \frac{I_0}{aKT N_s} \left(q + \frac{R_s}{\eta R(1-\alpha)^2}\right) + \frac{1}{R_{sh}}} \tag{8}$$

By substituting (8) into (5) yields

$$I/G = \frac{1}{\eta R_{load}(1-\alpha)^2} \frac{I_{ph}/G}{\frac{1}{\eta R_{load}(1-\alpha)^2} \left(1 + \frac{R_s}{R_{sh}}\right) + \frac{I_0}{aKT N_s} \left(q + \frac{R_s}{\eta R(1-\alpha)^2}\right) + \frac{1}{R_{sh}}} \tag{9}$$

At an insolation of G , I_{ph} can be expressed in terms of $I_{sc,n}$ as follows:

$$I_{ph}/G = (I_{sc,n} + K_1 \Delta T) \frac{G}{G_n} \tag{10}$$

With $I_{sc,n}$ is the short-circuit current at nominal conditions, K_1 is the short circuit current/temperature coefficient, and $\Delta T = T - T_n$ (T and T_n are the present and nominal temperatures respectively). From Equations (8) and (9), we get the Equation (10). In addition, Equations (11) and (12) can be obtained to derive the expressions V and I according to insolation.

$$\frac{dV}{dG} = \frac{(I_{sc,n} + K_1 \Delta T) \frac{1}{G_n} + K_1 \frac{G}{G_n} \frac{dT}{dG}}{\frac{1}{\eta R_{load}(1-\alpha)^2} \left(1 + \frac{R_s}{R_{sh}}\right) + \frac{I_0}{aKT N_s} \left(q + \frac{R_s}{\eta R_{load}(1-\alpha)^2}\right) + \frac{1}{R_{sh}}} > 0 \tag{11}$$

$$\frac{dI}{dG} = \frac{1}{\eta R_{load}(1-\alpha)^2} \frac{(I_{sc,n} + K_1 \Delta T) \frac{1}{G_n} + K_1 \frac{G}{G_n} \frac{dT}{dG}}{\frac{1}{\eta R_{load}(1-\alpha)^2} \left(1 + \frac{R_s}{R_{sh}}\right) + \frac{I_0}{aKT N_s} \left(q + \frac{R_s}{\eta R_{load}(1-\alpha)^2}\right) + \frac{1}{R_{sh}}} > 0 \tag{12}$$

The conditions $\frac{dV}{dG} > 0$ and $\frac{dI}{dG} > 0$ are validated in the case of a fast increase in insolation. Because in equations (11) and (12) the numerator and the denominator are positive values. Hence, from the I-V characteristics curve of the PV module, which are shown in Figure 8, it can be noticed that $dV > 0$ and $dI > 0$ in the case of a fast increase in insolation. Therefore, to avoid the drift problem, we rely on the information of dV and dI to make a good decision.

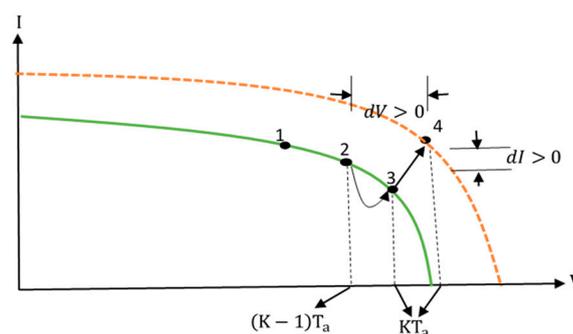


Figure 8. The I-V curves for analysis of the drift problem in case of a rapid increase in insolation with the observation of change in current (dI).

As shown in Figure 8, for two irradiation difference values, the dV and dI indicators are always positive when we have a fast increase in insolation. Therefore, to resolve the drift problem, the modified HC MPPT algorithm take a decision present in the Flowchart, as shown in Figure 9.

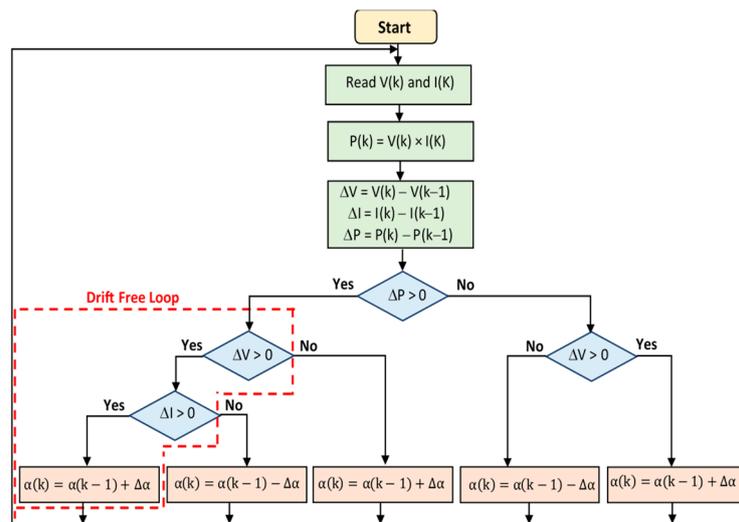


Figure 9. The flowchart of the modified HC MPPT algorithm.

2.2. PV System Design in Proteus Software

Figure 10 presents the PV system design, made in Proteus. As shown, this PV system can be divided into eight blocks as follows:

- Block (1): presents the subcircuit of the PV panel model.
- Block (2): presents the boost converter.
- Block (3): presents the embedded board (Arduino Uno).
- Block (4): presents the LCD screen, which is used to display the values of PV voltage, current and power.
- Block (5): presents the driver (TC4420), which is used to control the metal–oxide–semiconductor field-effect transistor (MOSFET) transistor of the Boost converter.
- Block (6): presents the current sensor (INA169) used for measuring the PV current. The modelization of this sensor is based on the INA168, which is available in the Proteus Tool. In order to model this sensor, you will need to follow the next steps:
 - ✓ Launch the Proteus tool application.
 - ✓ Open the Pick Devices.
 - ✓ Select the INA168 component.
 - ✓ Add two resistances R_5 (0.1 Ω) and R_6 (50 K Ω). Where R_s is a shunt resistor placed in series between the output of the PV module and the Boost converter, and R_L is a load resistor connected between the Pin 1 of the INA168 and the ground.
 - ✓ Set the power-supply voltage to 6 V in order to adapt it with that of INA169.

Then, the PV panel output current can be defined by the Equation (13) [25].

$$(I_S = \frac{V_{out1} 1K\Omega}{R_5 R_6}) \quad (13)$$

- Block (7): presents the module of the voltage sensor (B25 Voltage Sensor Module) used for measuring the PV voltage [26]. It is basically a voltage divider using two series resistances. The PV panel output voltage is defined by Equation (14):

$$(V_{out2} = \frac{R_3}{R_3 + R_4} V_{in}) \quad (14)$$

where ($R_3 = 25K\Omega$) and ($R_4 = 100K\Omega$)

- Block (8): presents the graph analysis, it is used to display the simulation results.

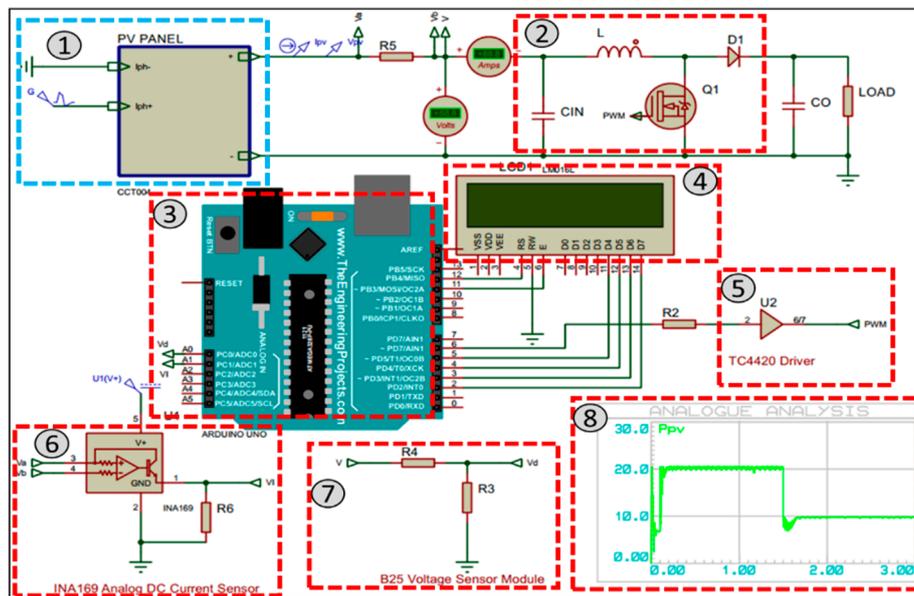


Figure 10. Schematic of PV system design in Proteus software.

3. Results and Discussion

In order to examine the effectiveness of the conventional and modified HC MPPT algorithms to solve the drift problem, a test case of a fast variation in insolation from 500 W/m^2 to 750 W/m^2 and from 750 W/m^2 to 1000 W/m^2 was performed. The corresponding results are shown in Figure 11. From this figure, it can be seen that the conventional HC algorithm cannot avoid the drift problem when the insolation is increased, while the modified HC algorithm tracks the MPP without the drift problem. In addition, Table 3 presents a comparison between conventional and modified HC MPPT under different uniform irradiance. The test case of a fast variation in insolation from 500 W/m^2 to 750 W/m^2 presents regarding efficiency (99.15%) and response time (10 ms). Further, the test case from 750 W/m^2 to 1000 W/m^2 in insolation presents regarding efficiency (99.21%) and response time (10 ms). In these two test cases, the efficiency and response time are improved by 1.2% and 70 ms respectively as compared to conventional HC. Therefore, the modified HC algorithm can improve the efficiency of the PV system by gaining additional power when drift occur, compared to the conventional HC algorithm. Hence, the present method allows obtaining a significant energy gain throughout the life cycle of the PV panel.

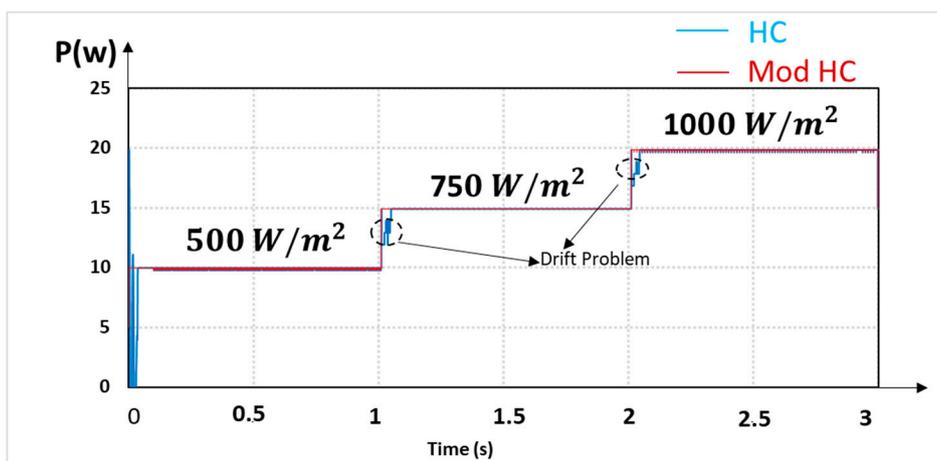


Figure 11. Output power with conventional and modified HC MPPT algorithm under a fast variation in insolation.

Table 3. Comparison between conventional and modified HC MPPT under different uniform irradiance.

MPPT Algorithms	G = 500 W/m ²		G = 750 W/m ²		G = 1000 W/m ²	
	Efficiency	Response Time	Efficiency	Response Time	Efficiency	Response Time
Conventional HC	98.39%	10 ms	98.55%	80 ms	98.85%	80 ms
Modified HC	99.11%	5 ms	99.15%	10 ms	99.21%	10 ms

4. Conclusions

In this paper, a simple PV system developed under Proteus software is proposed. This system can be used as a simulator to test the performances of MPPT algorithms before real implementation. The present simulator is composed of a PV panel model, DC-DC converter, voltage and current sensors, graph analysis, LCD screen, and an Arduino UNO board. It is demonstrated in this work that the choice of the Proteus software to test MPPT algorithms improves the PV system realization time. Because the same C MPPT code used in the simulation will be used for the real physical prototype. Further, the simulation results of the conventional and Modified HC MPPT algorithms under a fast variation in insolation shown that the modified method can avoid the drift problem. As well as addition, the efficiency and response time are improved by 1.2% and 70 ms respectively as compared to conventional HC.

As a perspective, future studies will be focused on the following two aspects: (i) Design of a simple method to find the GMPP under PSC with simple instructions; And (ii) the implementation of this method using the proposed PV Simulator.

Author Contributions: A.C., S.M. and A.E.G. performed the study of a simple PV simulator. In addition, A.C. performed the simulation examination. A.E.H., U.S. and A.D. revised the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the College of Engineering, Prince Sultan University, Riyadh, Saudi Arabia.

Acknowledgments: The authors like to express their sincere gratitude to the Renewable Energy research lab, College of Engineering, Prince Sultan University, Riyadh, Saudi Arabia for providing technical support.

Conflicts of Interest: The authors declare that they have no competing interests.

Abbreviations

PV	photovoltaic
CAs	conventional algorithms
MPPT	maximum power point tracking
HC	Hill-Climbing
GMPP	global maximum power point
PCB	printed circuit board

Nomenclatures

a	The diode ideality factor of the solar cell
I_{ph}	The photocurrent of the solar cell [A]
I_0	The output current of the Boost converter [A]
I_{mpp}	The current at MPP [A]
G	The solar irradiance level [W/m ²]
G_n	The solar irradiance nominal [W/m ²]
K	The constant of Boltzmann [J. K ⁻¹]
N_s	The number of cells connected in series
R_{load}	The load resistance [Ω]
R_{eq}	The equivalent input resistance of the converter Boost [Ω]
R_s	The series resistance of the solar cell [Ω]
R_{sh}	The shunt resistance of the solar cell [Ω]
T	The junction temperature [K]
T_n	The nominal temperatures [K]
V	The PV panel output voltage [V]
V_{mpp}	The voltage at MPP [V]
V_0	The Boost output voltage [V]
η	The efficiency of the DC-DC converter Boost [%]
α	The duty cycle.

References

1. Kousksou, T.; Allouhi, A.; Belattar, M.; Jamil, A.; El Rhafiki, T.; Arid, A.; Zeraouli, Y. Renewable energy potential and national policy directions for sustainable development in Morocco. *Renew. Sustain. Energy Rev.* **2015**, *47*, 46–57. [[CrossRef](#)]
2. Kabir, E.; Kumar, P.; Kumar, S.; Adelodun, A.A.; Kim, K.H. Solar energy: Potential and future prospects. *Renew. Sustain. Energy Rev.* **2018**, *82*, 894–900. [[CrossRef](#)]
3. Subramaniam, U.; Vavilapalli, S.; Padmanaban, S.; Blaabjerg, F.; Holm-Nielsen, J.B.; Almakhlis, D. A Hybrid PV-Battery System for ON-Grid and OFF-Grid Applications—Controller-In-Loop Simulation Validation. *Energies* **2020**, *13*, 755. [[CrossRef](#)]
4. Subudhi, B.; Pradhan, R. A comparative study on maximum power point tracking techniques for photovoltaic power systems. *IEEE Trans. Sustain. Energy* **2013**, *4*, 89–98. [[CrossRef](#)]
5. Ferrero Bermejo, J.; Gómez Fernández, J.F.; Pino, R.; Crespo Márquez, A.; Guillén López, A.J. Review and Comparison of Intelligent Optimization Modelling Techniques for Energy Forecasting and Condition-Based Maintenance in PV Plants. *Energies* **2019**, *12*, 4163. [[CrossRef](#)]
6. Motahhir, S.; El Hammoumi, A.; El Ghzizal, A. The most used MPPT algorithms: Review and the suitable low-cost embedded board for each algorithm. *J. Clean. Prod.* **2019**, *246*, 118983. [[CrossRef](#)]
7. Lee, H.S.; Yun, J.J. Advanced MPPT Algorithm for Distributed Photovoltaic Systems. *Energies* **2019**, *12*, 3576. [[CrossRef](#)]
8. Pathy, S.; Subramani, C.; Sridhar, R.; Thentral, T.; Padmanaban, S. Nature-inspired MPPT algorithms for partially shaded PV systems: A comparative study. *Energies* **2019**, *12*, 1451. [[CrossRef](#)]
9. Bahrami, M.; Gavagsaz-Ghoachani, R.; Zandi, M.; Phattanasak, M.; Maranzana, G.; Nahid-Mobarakeh, B.; Pierfederici, S.; Meibody-Tabar, F. Hybrid maximum power point tracking algorithm with improved dynamic performance. *Renew. Energy* **2019**, *130*, 982–991. [[CrossRef](#)]

10. Huang, C.; Wang, L.; Yeung, R.S.C.; Zhang, Z.; Chung, H.S.H.; Bensoussan, A. A prediction model-guided Jaya algorithm for the PV system maximum power point tracking. *IEEE Trans. Sustain. Energy* **2017**, *9*, 45–55. [[CrossRef](#)]
11. Bingöl, O.; Özkaya, B. Analysis and comparison of different PV array configurations under partial shading conditions. *Solar Energy* **2018**, *160*, 336–343. [[CrossRef](#)]
12. Belhachat, F.; Larbes, C. Comprehensive review on global maximum power point tracking techniques for PV systems subjected to partial shading conditions. *Solar Energy* **2019**, *183*, 476–500. [[CrossRef](#)]
13. Huang, C.; Wang, L.; Zhang, Z.; Yeung, R.S.C.; Bensoussan, A.; Chung, H.S.H. A Novel Spline Model Guided Maximum Power Point Tracking Method for Photovoltaic Systems. *IEEE Trans. Sustain. Energy* **2019**. [[CrossRef](#)]
14. Ishaque, K.; Salam, Z. A comprehensive MATLAB Simulink PV system simulator with partial shading capability based on two-diode model. *Solar Energy* **2011**, *9*, 2217–2227. [[CrossRef](#)]
15. Bagyaveereswaran, V.; Umashankar, S.; Arulmozhivarman, P. Simulation Tool for Tuning and Performance Analysis of Robust, Tracking, Disturbance Rejection and Aggressiveness Controller. *Algorithms* **2019**, *12*, 144. [[CrossRef](#)]
16. Motahhir, S.; El Ghzizal, A.; Sebti, S.; Derouich, A. Proposal and Implementation of a novel perturb and observe algorithm using embedded software. In Proceedings of the 2015 3rd International Renewable and Sustainable Energy Conference (IRSEC), Marrakech, Morocco, 10–13 December 2015; pp. 1–5.
17. Motahhir, S.; Chalh, A.; El Ghzizal, A.; Sebti, S.; Derouich, A. Modeling of Photovoltaic Panel by using Proteus. *J. Eng. Sci. Technol. Rev.* **2017**, *10*, 8–13. [[CrossRef](#)]
18. Motahhir, S.; Chalh, A.; El Ghzizal, A.; Derouich, A. Development of a low-cost PV system using an improved INC algorithm and a PV panel Proteus model. *J. Clean. Prod.* **2018**, *204*, 355–365. [[CrossRef](#)]
19. Motahhir, S. Data for: Development of a low-cost PV system using an improved INC algorithm and a PV panel Proteus model. *Mendeley Data* **2020**. [[CrossRef](#)]
20. Motahhir, S. PV Panel in Proteus. *Mendeley Data* **2017**. [[CrossRef](#)]
21. Yaqoob, S.J.; Obed, A.A. Modeling, Simulation and Implementation of PV System by Proteus Based on Two-diode Model. *J. Tech.* **2019**, *1*, 39–51.
22. Chaibi, Y.; Salhi, M.; El-Jouni, A.; Essadki, A. A new method to extract the equivalent circuit parameters of a photovoltaic panel. *Solar Energy* **2018**, *163*, 376–386. [[CrossRef](#)]
23. Killi, M.; Samanta, S. Modified perturb and observe MPPT algorithm for drift avoidance in photovoltaic systems. *IEEE Trans. Ind. Electron.* **2015**, *62*, 5549–5559. [[CrossRef](#)]
24. Xiao, W.; Dunford, W.G. A modified adaptive hill climbing MPPT method for photovoltaic power systems. In Proceedings of the 2004 IEEE 35th Annual Power Electronics Specialists Conference (IEEE Cat. No. 04CH37551), Aachen, Germany, 20–25 June 2004; Volume 3, pp. 1957–1963.
25. INA1x9 Datasheet [WWW Document]. 2017. Available online: www.ti.com (accessed on 8 January 2018).
26. Voltage Sensor Module-Arduino Compatible [WWW Document], n.d. Available online: <https://www.emartee.com/product/42082/VoltageSensor> (accessed on 8 January 2018).

