IET Renewable Power Generation

Research Article

IET Journals
The Institution of Engineering and Technology

Simple and fast computation photovoltaic emulator using shift controller

ISSN 1752-1416
Received on 31st December 2019
Revised 6th May 2020
Accepted on 18th May 2020
E-First on 22nd July 2020
doi: 10.1049/iet-rpg.2019.1504
www.ietdl.org

Razman Ayop[1] ✉, *Chee Wei Tan*[1], *Abba Lawan Bukar*[1,2]

[1]*School of Electrical Engineering, Universiti Teknologi Malaysia, 81310, UTM Johor Bahru, Malaysia*
[2]*Faculty of Electrical Engineering, University of Maiduguri, P.M.B 1069, Borno, Nigeria*
✉ *E-mail: razman.ayop@utm.my*

Abstract: Photovoltaic (PV) emulator (PVE) is a non-linear power supply that produces a similar output to that of the PV module. It requires a closed-loop controller such as the proportional–integral (PI) and fuzzy logic controllers to operate properly. The PI controller has a low-computational burden and is not robust since the transient response varies under different load conditions. Although the fuzzy logic controller is robust, it requires high-computation capability. This study proposed an alternative approach called the shift controller for the PVE to easily compute and maintain a fast transient response under various load conditions. The performance of the proposed shift controller for the PVE is compared with the performance of the PVEs using the PI and fuzzy logic controllers, respectively. The controllers are simulated using MATLAB/Simulink and implemented using dSPACE ds1104 for experimental validation. The results show that the proposed shift controller computes up to 18 times faster than the fuzzy logic controller. The proposed controller also responds faster to load change than the PI controller.

1 Introduction

The photovoltaic (PV) emulator (PVE) is a non-linear power supply that produces similar current–voltage (*I–V*) characteristic to that of the PV module or array. The use of PVE allows a flexible experimental condition for the PV energy generation system. The irradiance level is easily controlled using the graphical user interface, instead of relying on irradiance from the sun or an expensive controllable light when conducting the test on the PV system. The PVE is also commonly more efficient than the PV module, which allows the use of a lower rating power supply during the testing phase.

The PVE consists of the power converter and PV model interconnected by the control strategy [1, 2]. The function of the control strategy function is to determine the operating point of the PVE based on the load, irradiance, and temperature. The control strategy for the PVE is said to be robust if it can rapidly and accurately compute the operating point. The commonly used control strategies include the direct referencing method (DRM), hybrid-mode control, and resistance feedback method [3, 4]. The DRM uses a power converter to compute the operating point. It is easy to implement, but it takes a longer time to compute for the operating point and the output voltage oscillates under certain conditions. The hybrid-mode control method reduces the oscillation problem faced by DRM [5, 6]. However, the hybrid-mode control strategy is complex since it requires multiple PV models and controllers for the power converter. The resistance feedback method provides an accurate and reliable operating point for the PVE [7, 8]. Nonetheless, this control strategy requires a non-conventional PV model.

The conventional PV model used by various control strategies is the *I–V* PV model, which has voltage as the input and current as the output. On the other hand, the resistance feedback method requires current–resistance (*I–R*) PV model and has resistance as the input and current as the output [7, 9]. Although the characteristic equation between the *I–V* and *I–R* PV model is different, both of these equations are derived from the same PV equivalent circuit. Therefore, the results from these two PV models are the same.

There are two components involved in the power converter of the PVE, namely: the type of the power converter and the controller for the power converter. The buck controller is commonly used in the PVE application [3, 10, 11]. The flyback and forward converters are also implemented in the PVE for the galvanic isolation [12, 13]. The resonant converter allows the PVE to operate with the soft switching [14, 15]. Nonetheless, the operation is highly limited.

The controller for the power converter applied in the PVE affects the transient response. The PVE needs to have a fast transient response similar to the PV module [16]. Therefore, a good controller for the power converter is needed in the PVE application. The basic controller for PVE is to manually calculate the duty cycle based on the reference input [17]. However, this approach is prone to error. The proportional–integral (PI) controller is commonly used for the PVE since it is easily implemented and requires low computation [3, 7, 18]. This characteristic is important since the hardware platform also needs to compute the PV model to obtain the operating point for the PVE. The PI–derivative (PID) controller is also used in the PVE and has a low-computation burden almost similar to the PI controller [10, 19]. Nevertheless, both PI and PID controllers require complex mathematical modelling of the power converter and complex tuning method.

The other controllers used for the PVE are the linear quadratic regulator (LQR) [20] and model predictive control (MPC) [21]. The LQR and MPC have better performance than the PI controller [22]. Nonetheless, it involves complex mathematical derivation. The hybrid fuzzy controller is also used for the PVE. This controller combines with the PI [23] or PID [24] controllers to improve transient performance. The fuzzy controller is complicated since it involves a lot of mathematic equations during the operation, which burdens the hardware platform.

If a complex controller such as the fuzzy logic is used, the computation time becomes longer and this may result in unstable operation and inaccurate emulation. Despite the aforementioned advantages of the PI controller, the mathematical derivations of its transfer function for the power converter are quite complex. The techniques used to design the PI controller are also complicated. Besides, the PI controller is not robust. The transient response may become significantly slower when the load changes, which affects the performance of the PVE [7, 25]. In view of the foregoing analysis, it is, therefore, essential to design a controller with a low-computational requirement, with a fast transient response, highly

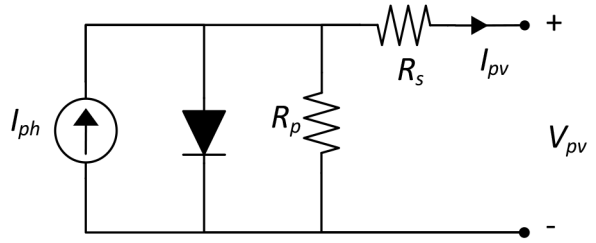


Fig. 1 *Single diode model*

Table 1 Parameters of the Ameresco Solar80J-B PV module [29]

Parameter	Value
open-circuit voltage, V_{oc}	44.4 V
short circuit current, I_{sc}	2.32 A
maximum power voltage, V_{mp}	35.2 V
maximum power current, I_{mp}	2.15 A
maximum power, P_{max}	75.71 W_p
temperature coefficient of I_{sc}, α	0.0024 A/°C
temperature coefficient of V_{oc}, β	−0.4 V/°C
number of cells in series, N_s	72
ideality factor, A_f	1.65
series resistance, R_s	1.00 Ω
parallel resistance, R_p	3500 Ω

robust, and requires no transfer function of the power converter to improve the quality of the PVE.

This study proposed a new controller called shift controller for the power converter. The proposed controller has a low-computational burden and fast response under various load conditions. The controller is implemented in the PVE and the results are compared with the PVEs based on the PI and fuzzy controllers. All PVEs use the same buck converter and PV model to produce an unbiased comparison. The buck converter uses a current-mode controlled closed-loop system, while the single diode model is used as the PV model. The control strategy used for the PVEs is the resistance feedback method due to its robustness. The PVEs are simulated using the MATLAB/Simulink and experimentally validated using dSPACE ds1104 rapid prototyping.

The study commences with the design of the components of the PVE, which includes the conventional PV model and the buck converter. The next section discusses the conventional controllers used as the benchmark for the proposed controller. These controllers are the PI and fuzzy logic controllers. The proposed shift controller is discussed in Section 4. The last section covers the results and discussions of the proposed controller and the comparison with the PI and fuzzy logic controllers. The analysed performances include computational time, accuracy, and transient responses.

2 Components of PVE

Basically, the PVE is composed of the PV model and power converter. The PV model and power converter are the single diode *I–R* PV model and buck converter, respectively.

2.1 I–R PV model

Since the resistance feedback control strategy is used, the *I–R* PV model is used instead of the conventional *I–V* PV model. The *I–R* PV model is based on the single diode model shown in Fig. 1. This model is suitable for the PVE since it is simple and accurate. The characteristic equation of the *I–R* PV model is shown in (1) [7, 8]. The diode current, I_d, photocurrent, I_{ph}, and saturation current, I_s, are calculated using (2)–(4), respectively [26–28]. The parameters of the model are based on the Ameresco Solar80J-B PV module, which is shown in Table 1 [29]

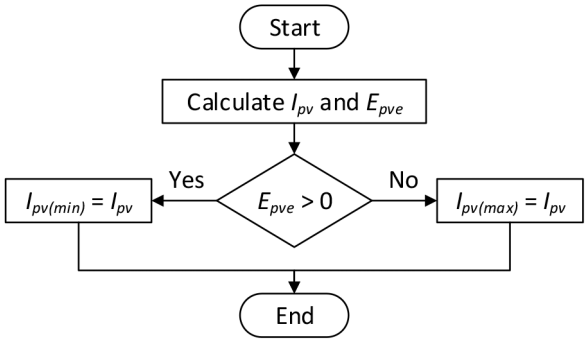


Fig. 2 *I–R PV model computation using the binary search method*

$$I_{pv} = I_{ph} - I_d - \frac{I_{pv}(R_{pv} - R_s)}{R_p} \tag{1}$$

$$I_d = I_s\left(\exp\left(\frac{I_{pv}q(R_{pv} - R_s)}{A_f kT}\right) - 1\right) \tag{2}$$

$$I_{ph} = \frac{G}{G_{stc}}(I_{sc} + \alpha(T - T_{stc})) \tag{3}$$

$$I_s = \frac{I_{sc} + \alpha(T - T_{stc})}{\exp\left(V_{oc} + \frac{\beta q(T - T_{stc})}{A_f kT}\right) - 1} \tag{4}$$

where I_{pv} denotes the PV current (A), q denotes the electron charge $(1.6 \times 10^{-19}\ \text{C})$, R_{pv} denotes the PV resistance (Ω), R_s denote the series resistance (Ω), A_f denotes the ideality factor, k denotes the Boltzmann constant $(1.38 \times 10^{-23}\ \text{J/K})$, T denotes the temperature of the p–n junction (K), R_p denotes the parallel resistance (Ω), G denotes the irradiance (W/m^2), STC denotes the standard test condition (1000 W/m^2 and 25°C), I_{sc} denotes the short circuit current (A), G_{stc} denotes the irradiance under STC (1000 W/m^2), α denotes the temperature coefficient of I_{sc}, T_{stc} denotes the temperature under STC (25°C), V_{oc} denotes the open circuit voltage (V) and β denotes the temperature coefficient of V_{oc}.

The *I–R* PV model is an implicit equation that requires a numerical method to solve. Therefore, the *I–R* PV model is computed using the binary search method [7, 8]. To implement the binary search method into the *I–R* PV model, the characteristic equation is modified from (1) to become (5) and (6). The computed I_{pv} is calculated using (7), which the maximum I_{pv}, $I_{pv(max)}$, and minimum I_{pv}, $I_{pv(min)}$, is updated in each iteration using the algorithm shown in Fig. 2

$$E_{pvm} = I_{ph} - I_{dm} - \frac{I_{pv}(R_{pv} - R_s)}{R_p} - I_{pv} \tag{5}$$

$$I_{dm} = I_s\left(e^{\frac{I_{pv}(R_{pv} - R_s)}{A_f V_T}} - 1\right) \tag{6}$$

$$I_{pv} = I_{pv(min)} + \left(\frac{I_{pv(max)} - I_{pv(min)}}{2}\right) \tag{7}$$

2.2 Buck converter

The PVE that produces the V_o and I_o without the ripple is preferred. Since the buck converter is a switched-mode power supply, this ripple cannot be avoided. Nonetheless, the advantages outweighed the disadvantages. The buck converter is highly efficient, easily controlled, requires a low number of components, and low in cost. These advantages make the buck converter commonly used in the PVE application.

Three components need to be determined for the buck converter. These components are shown in Fig. 3, which include the input voltage (V_i), inductance (L), and capacitance (C).

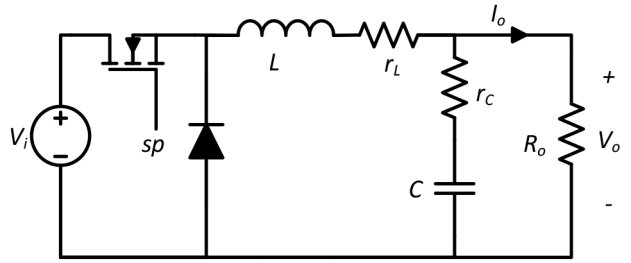


Fig. 3 *Circuit diagram of the buck converter with internal resistances*

Table 2 Specifications of the buck converter

Parameter	Value
input voltage, V_i	60 V
switching frequency, f_s	20 kHz
duty cycle, D	0.05–0.80
inductance, L	1.75 mH
internal resistance of inductor, r_L	0.83 Ω
capacitance, C	36 µF
internal resistance of capacitor, r_C	0.26 Ω
output resistance, R_o	5–90 Ω
MOSFET drain-source on resistance, $R_{ds(on)}$	0.28 Ω
diode forward voltage, V_f	0.44 V

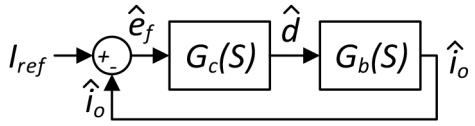


Fig. 4 *Block diagram of the closed-loop current-mode control buck converter system using the PI controller*

The V_i needs to be higher than V_{oc} and the V_i is calculated using (8) [8, 30]. The maximum R_o, $R_{o(max)}$, affects the L required by the PVE with the buck converter. A larger L is needed for a larger $R_{o(max)}$ to maintain continuous current mode. For the optimum performance of the PVE, the $R_{o(max)}$ is calculated during the operation at the maximum power point for the low G, as shown in (9). The L is also calculated during that condition, which is shown in (10) [8, 30]. The V_o ripple is large when the V_o is low. The V_o ripple is manipulated using the C, which is calculated using (11) [8, 30]

$$V_i = \frac{V_{oc}}{D_{max}} \tag{8}$$

$$R_{o\_max} = \frac{V_{mp\_G(min)}}{I_{mp\_G(min)}} \tag{9}$$

$$L = \frac{\left(1 - \frac{V_{mp\_G(min)}}{V_i}\right) R_{o\_max}}{2 f_s} \tag{10}$$

$$C = \frac{1 - D_{min}}{8 L \gamma_{V_o} f_s^2} \tag{11}$$

where D_{max} is the maximum D, $V_{mp\_G(min)}$ is the maximum power voltage at minimum G, $I_{mp\_G(min)}$ is the maximum power current at minimum G, f_s is the switching frequency, D_{min} is the minimum D, and γ_{V_o} is the output voltage ripple factor.

D_{max} is set to 0.8 and the calculated V_i is 55.5 V. The chosen V_i is 60 V to compensate for the non-ideality of the components in the buck converter. The calculated $R_{o(max)}$ is 74.9 Ω at 200 W/m^2. The chosen $R_{o(max)}$ is 90 Ω, which allows the PVE to emulate the constant voltage region at 200 W/m^2. The calculated L is 1.06 mH, in which the $V_{mp\_G(min)}$ obtained from the PV model is 31.7 V. A higher L is chosen to avoid discontinuous current mode operation, in which the chosen L is 1.75 mH. The calculated C is 17 µF and the C is increased to 36 µF to accommodate the internal resistance of the capacitor, r_C. The parameters of the buck converter are shown in Table 2.

3 PVE using conventional controller

Two types of conventional controllers are implemented in two separate PVEs. These PVEs are used as the benchmark to compare the performance of the proposed PVE. The conventional controllers are the PI and fuzzy controllers. Several criteria are listed for a fair comparison between the controllers:

- All controllers are designed at the load of 5 Ω.
- The controller is designed with the settling time, t_s, around 3 ms with no overshoot at 5 Ω.
- The designed controllers are fixed for all load ranges.

Since the buck converter performs faster at low load conditions, the design of all controllers is focused on 5 Ω load, which is the lowest limit set for the PVE. If the PI and fuzzy controllers are tuned to a specific load, the performance of these controllers is nearly similar. However, if a different load is tested with the same controllers, the performance of this controller becomes different. The ability of the controller to maintain the performance at various conditions is called robustness. To test the robustness of the controller, the performance at a specific load is established. All the controllers need to have the t_s around 3 ms with no overshoot when the load is 5 Ω. Then, the load is changed and the robustness of the controller is analysed.

3.1 PI controller

The transfer function for the buck converter, G_b, and the PI controller, G_c, are shown in (12) and (13), respectively [31]. The relationship between (12) and (13) is presented in Fig. 4. The proportional gain, K_p, and integral gain, K_i, are tuned using the 'PID tuner' function in MATLAB/Simulink. The K_p and K_i are 0.0063 and 85.26, respectively

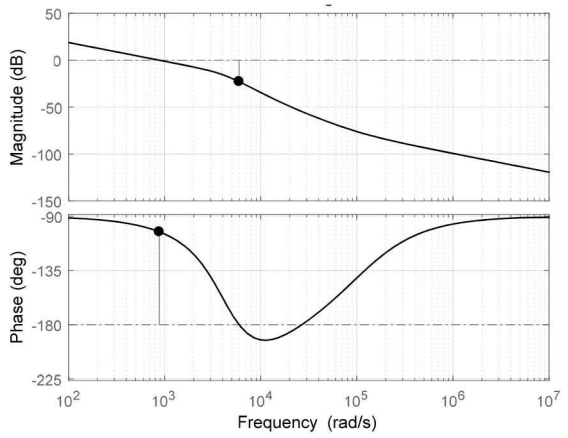
$$G_b(s) = \frac{\hat{d}(s)}{\hat{i}_o(s)} = \frac{V_i}{R_o} \frac{\frac{1}{LC}}{s^2 + \frac{1}{R_o C} s + \frac{1}{LC}} \tag{12}$$

$$G_c(s) = \frac{\hat{e}_f(s)}{\hat{d}(s)} = K_p + K_i / s \tag{13}$$

where $\hat{d}$ is the small signal of D, $\hat{i}_o$ is the small signal of I_o, and $\hat{e}_f$ is the small signal of E_f.

The stability of the PI controller is analysed using a bode plot. The analysis is conducted at the $R_{o(min)}$, which is 5 Ω. A good closed-loop system needs to have a large gain and phase margins. By referring to Fig. 5, the gain and phase margins are large. With the load of 5 Ω, the gain and phase margins are 22.8 dB (at 5950 rad/s) and 77.1° (at 879 rad/s), respectively. This PI controller produces 0% overshoot and t_s of 3.6 ms. This follows the specified criteria of the controller's design. Note that this result is based on the transfer function and not an actual circuit.

The PI controller was implemented in the PVE based on the resistance comparison method shown in Fig. 6. The V_o and I_o are measured and the R_o is digitally calculated by dividing the V_o over the I_o. The I_{ref} is calculated by the *I–R* PV model based on R_o, G, and T. The I_{ref} is compared with I_o to obtain E_f. The D is calculated by the PI controller based on E_f. The pulse-width modulation (PWM) produces the switching pulse, sp, according to D. The sp controls the buck converter and the process is repeated.



a

Performance and Robustness	
	Tuned
Rise time	0.00187 seconds
Settling time	0.00363 seconds
Overshoot	0 %
Peak	1
Gain margin	22.9 dB @ 5.98e+03 ra...
Phase margin	77.1 deg @ 879 rad/s
Closed-loop stability	Stable

b

Fig. 5 *The design of the PI controller*
(a) Bode plot of the closed-loop buck converter system at 5 Ω,
(b) Summary of the MATLAB PID tuner at 5 Ω

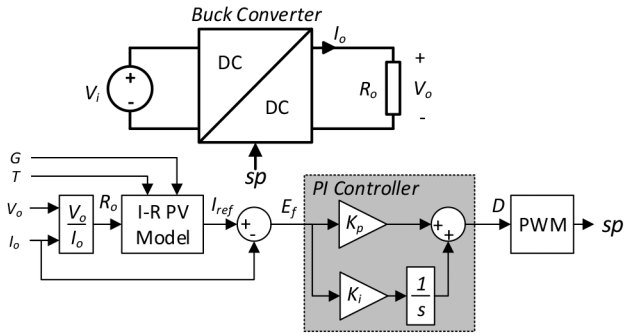


Fig. 6 *Block diagram of the PVE using a PI controller (PVEPI)*

3.2 Fuzzy controller

The fuzzy controller is designed according to the heuristic criteria. This is different from the PI controller based on the complex derivations and mathematical equations. The fuzzification is the interface that converts the controller input into the information that is used by the fuzzy interference shown in Fig. 7 [32–34]. The fuzzy interference emulates decision making based on the rules. The defuzzification converts the conclusions of fuzzy interference and produces the output.

The fuzzy controller for the buck converter is based on these criteria [34]:

- The change of D, dD, needs to be large if I_o is far away from I_{ref}.
- The dD needs to be small if I_o is close to I_{ref}.
- The D needs to be fixed to prevent overshoot if I_o reaches I_{ref} and I_o approaches I_{ref} rapidly.
- A small dD is needed to prevent I_o from moving away from I_{ref} if I_o reaches I_{ref} and I_o still changing.
- The D needs to be fixed if I_o reaches I_{ref} and I_o reaches a steady state.
- The sign of dD needs to be negative if I_o is above I_{ref}.

Based on the above criteria, the membership functions and fuzzy matrix rules produced are shown in Fig. 8 and Table 3, respectively

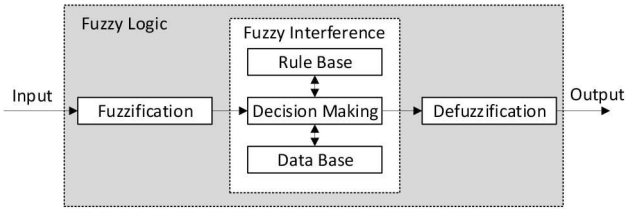


Fig. 7 *Block diagram of the fuzzy logic*

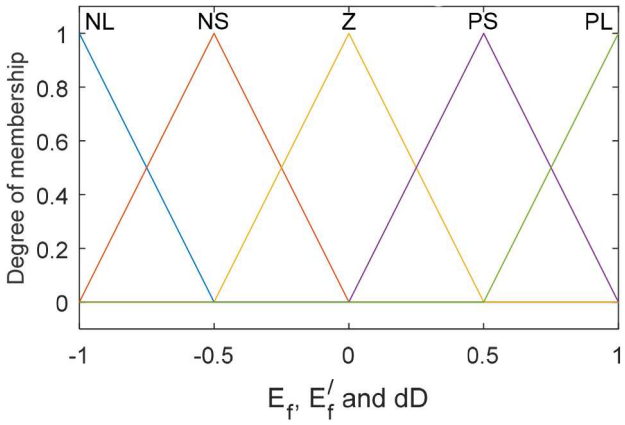


Fig. 8 *Triangular membership function for E_f, E_f', and dD*

Table 3 Fuzzy rule matrix [34]

		E_f'				
		NL	**NS**	**Z**	**PS**	**PL**
E_f	**NL**	NL	NL	NL	NS	Z
	NS	NL	NL	NS	Z	PS
	Z	NL	NS	Z	PS	NL
	PS	NS	Z	PS	PL	PL
	PL	Z	PS	NL	PL	PL

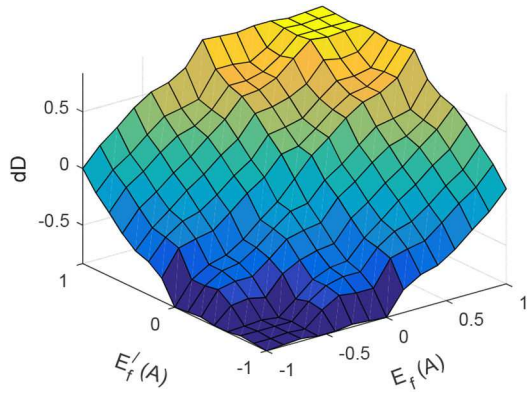


Fig. 9 *3D surface of the fuzzy logic corresponding to the membership functions and fuzzy rule matrix*

[34]. The membership functions for E_f, the derivative of E_f, E_f', and the change of D (dD) consist of five parts, which are the negative large, negative small (NS), zero (Z), positive small (PS), and positive large (PL). The E_f and E_f' are the input for the fuzzy logic and the dD is the output for the fuzzy logic.

The 3D surface generated from the membership functions and fuzzy rule matrix is shown in Fig. 9. The surface shows that dD is large when E_f is large and dD is small when E_f is small. The dD is zero when E_f is zero or if $E_f'$is large. The dD is small if E_f is already zero, but E_f' is still changing. The dD is zero if E_f and E_f' are zero. These characteristics correspond to the criteria set to control the buck converter.

The fuzzy logic is implemented in the PVE according to Fig. 10 [35]. The V_o and I_o are measured and R_o is digitally calculated by

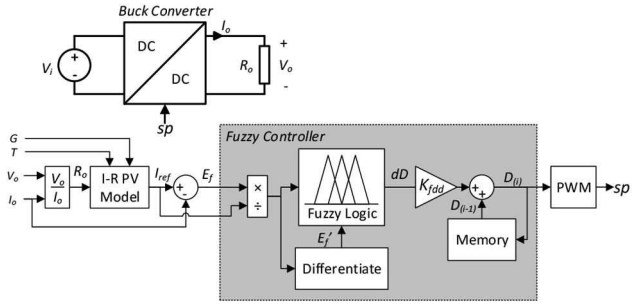


Fig. 10 *Block diagram of the PVE using a fuzzy controller (PVEF)*

Table 4 Common relationship between error and duty cycle

E_f	NL	NS	Z	PS	PL
dD	NL	NS	Z	PS	PL

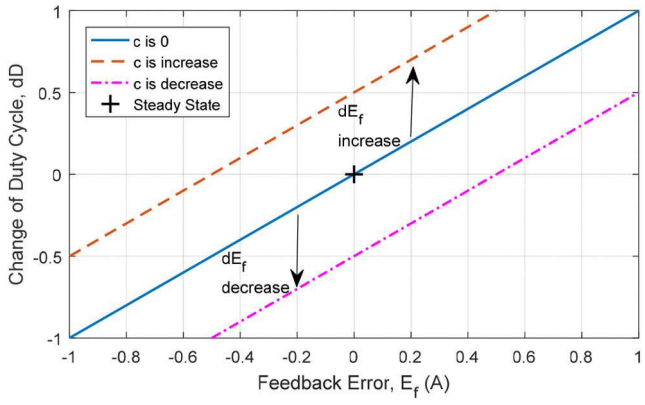


Fig. 11 *Concept of the shift controller*

dividing V_o over I_o. The I_{ref} is calculated by the *I–R* PV model based on R_o, G, and T. The I_{ref} is compared with I_o to obtain E_f. The E_f is divided by I_{ref} to standardise the input for the fuzzy logic. The derivative of E_f is calculated. The E_f and E_f' are fed into the designed fuzzy logic to produce dD. The dD is multiplied with the fuzzy dD gain, K_{fdd}, and it is added with the previous D, $D_{(i-1)}$, to obtain the current D, $D_{(i)}$. The K_{fdd} is adjusted to 0.0122 to ensure t_s is 3 ms at 5 Ω load, which follow the controller's design criteria. The PWM produces sp according to $D_{(i)}$. The sp controls the buck converter and the process is repeated again.

4 PVE using the proposed shift controller

The concept for the shift controller is based on the characteristic of the conventional fuzzy logic controller and it is derived using a single linear equation. This controller is then implemented into the PVE using the resistance feedback method, which is a similar method used for the PVEs using a conventional controller.

4.1 Proposed shift controller

Based on the PI and fuzzy logic controllers, a controller for the buck converter commonly uses the E_f to determine D. To obtain the current D, $D_{(i)}$, the dD is added to the previous D, $D_{(i-1)}$, as shown in (15)

$$D_{(i)} = D_{(i-1)} + \mathrm{d}D \tag{14}$$

The larger the absolute E_f, the larger the absolute dD. For the negative feedback system of the buck converter, if E_f is positive, dD produced by the controller is positive. While if E_f is negative, the dD produced by the controller is negative. The dD produced is zero when E_f is zero and the system achieves the steady-state. This relationship is tabulated in Table 4.

The relationship shown in Table 4 also can be represented in the form of a linear equation in Fig. 11, in which E_f is at the x-axis and the dD is at the y-axis. The gradient of the line, m, is the compensation gain and the intercept on the y-axis, c, is equal to zero. The c is equal to zero because when E_f is Z, dD is Z.

Although this approach allows the output of the buck converter to follow the reference input, the transient response produced is slow. The transient response becomes faster if m is increased. Nonetheless, the system oscillates and becomes unstable when m is too large.

To overcome this problem, the absolute dD is temporarily increased during the transient response, which is a similar concept to the fuzzy logic controller. This is observed in Table 3, in which dD is temporarily increased when E_f' increases.

The temporary increase in dD can be imitated in a linear equation by shifting the line shown in Fig. 11. If E_f is positive, the system is in the transient mode. The c is increased or shifted-up to increase dD, which produces a faster transient response. If E_f is negative, the system is also in the transient mode. The c is decreased or shifted down to increase the dD to the negative direction, which also results in faster transient response. When E_f is zero, the system achieves steady state.

The c needs to move to the zero or original state to avoid oscillation or unstable output. The shifting of the c depends on the transient state. The transient state of the system is measured by calculating the change of E_f, dE_f. A larger absolute dE_f shows the system is in the early portion of the transient state and a small absolute dE_f shows the system is in the final portion of the transient state.

Based on this concept, the controller is realised using the linear equation shown in (15). This equation is modified according to Fig. 11 and the equation becomes (16), in which m is assumed equal to one for derivation simplification. The dE_f is the difference between the current dE_f, d$E_{f(i)}$, and the previous dE_f, d$E_{f(i-1)}$, which is substituted into (16) to become (17). The current dD, d$D_{(i)}$ is simplified and the final equation is shown in (18)

$$y = mx + c \tag{15}$$

$$\mathrm{d}D = E_f + \mathrm{d}E_f \tag{16}$$

$$\mathrm{d}D_{(i)} = E_{f(i)} + \left(E_{f(i)} - E_{f(i-1)}\right) \tag{17}$$

$$\mathrm{d}D_{(i)} = 2E_{f(i)} - E_{f(i-1)} \tag{18}$$

The $D_{(i)}$ is calculated by adding $D_{(i-1)}$ to the product of the d$D_{(i)}$ and the dD gain, K_{dd}, as shown in er. The K_{dd} is tuned manually using the trial and error method. The K_{dd} is 0.01 that produces t_s around 3 ms at 5 Ω load, which follows the controller's design criteria. The shift controller presented in (19) is a simple equation that requires low computation. As a result, the sampling time required by the hardware platform is lower, allowing a higher frequency operation of the buck converter and faster transient response. Since PVE operating points change based on load, G, and T, the d$D_{(i)}$ needs to adapt with reference input, Ref. To do this, d$D_{(i)}$ is divided by Ref

$$D_{(i)} = D_{(i-1)} + \frac{K_{dd}}{Ref}\left(2E_{f(i)} - E_{f(i-1)}\right) \tag{19}$$

4.2 Implementation of shift controller into PVE

The V_o and I_o were measured and the R_o was digitally calculated by dividing V_o by I_o. The I_{ref} was calculated by the *I–R* PV model based on the R_o, G, and T. The I_{ref} was compared with I_o to obtain $E_{f(i)}$. The $D_{(i)}$ was calculated using the proposed shift controller, which is from (19). The Ref is obtained from I_{ref}. The $E_{f(i-1)}$ and $D_{(i-1)}$ are obtained using the 'memory' block in MATLAB/Simulink. The 'memory' block is affected by the sample time. Therefore, the sample time for this block is fixed. The $D_{(i)}$ was fed

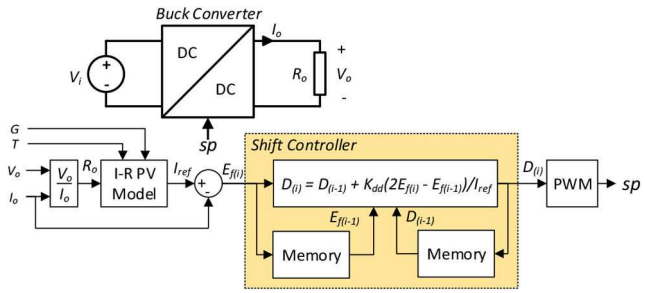


Fig. 12 *Block diagram of the proposed PVE using shift controller (PVES)*

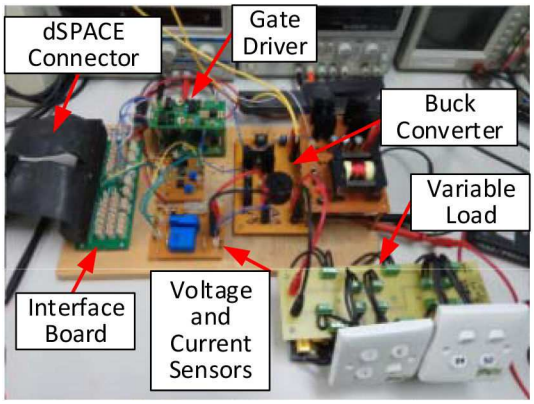


Fig. 13 *Experimental validation setup*

into the PWM to produce sp. The sp controlled the buck converter and the process was repeated (see Fig. 12).

5 Results and discussions

The simulations and experimental validations are conducted to examine the capability of the proposed shift controller in improving the performance of the PVE. The performance examination includes the computation of the controller, its accuracy, and transient response of the PVE.

The experimental validation setup is shown in Fig. 13. The V_o and I_o are measured using the LEM-LV25 and LEM-LY05, respectively. The sensors are connected to the interface board, in which the measured V_o and I_o are transmitted to dSPACE ds1104 via dSPACE connector. The PVE's controller produces the sp. The sp is amplified by the gate driver from 5 V to 15 V. The amplified sp is connected at the gate of the buck converter's metal oxide semiconductor field-effect transistor (MOSFET).

5.1 Computation of controller

A small t_{sam} is important to control the PVE properly. As the t_{sam} reduces, a higher f_s is able to be used. This reduces the L and C needed by the buck converter to maintain continuous current mode operation and the desired output voltage ripple factor, respectively. As a result, faster transient response is achieved for the PVE. The t_{sam} used in the simulation is fixed to 50 µs. However, in the experimental validation, the t_{sam} depends on the complexity of the controller. A complex controller requires a longer time to compute, thus has a higher t_{sam}. In the experimental validation, the t_{sam} is measured using dSPACE ds1104.

By referring to Table 5, the PVE using the PI controller (PVEPI) requires 31 µs to compute the PV model and the PI controller using dSPACE ds1104. The fast computation is due to the simplicity of the PI controller. This is because the PI controller only consists of two gains and an integrator. Nonetheless, the PVEF requires 518 µs to compute the controller, which is up to 16 times slower than the PVEPI. The high t_{sam} is caused by the complex fuzzy logic used in the PVE's controller. The fuzzy logic consists of 15 membership functions that need to obey 25 rules, which involve a lot of mathematic equations. Since the proposed

Table 5 Sample time needed by dSPACE ds1104 to compute the PVEs with the various power converter's controllers

Type of PVE	Sample Time, t_{sam}
PVEPI	31 µs
PVEF	518 µs
PVES	29 µs

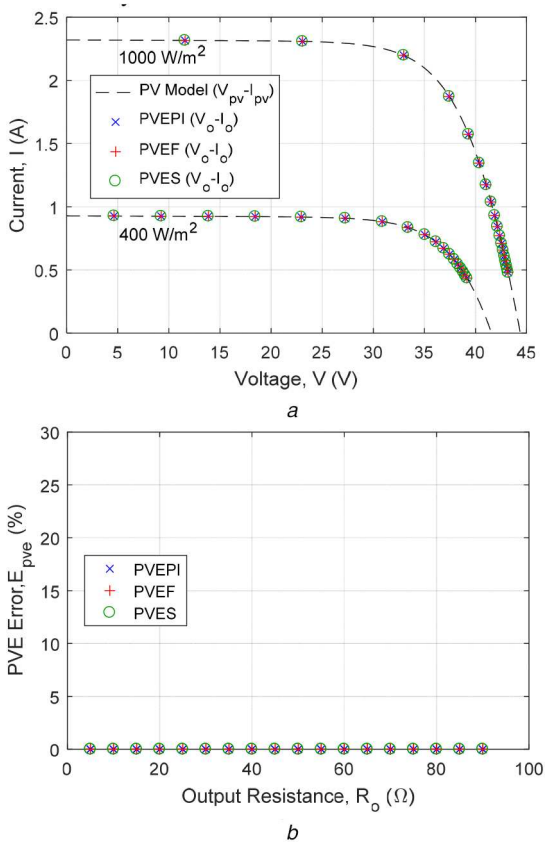


Fig. 14 *The accuracy test based on the simulation*
(a) Simulated V_o and I_o mapped on the *I–V* characteristic curve,
(b) Corresponding PVE error compared to the PV model

shift controller is based on a simple basic linear equation, the computation time of the PVE using the shift controller (PVES) only requires 29 µs to compute. The computation time of the proposed PVES is faster than the computation time of the PVEPI and PVEF. This shows that the PVES does not burden the hardware platform, which allows the PVE to operate at a high f_s and produces a fast transient response.

5.2 Accuracy of PVE

The accuracy of the V_o and I_o is the steady-state response analysis and it is an important aspect in analysing the performance of the PVE. A highly accurate PVE is able to follow the PV model accurately and has zero steady-state error. The simulation results in Fig. 14*a* show that all PVEs are able to follow the *I–V* characteristic curve of the PV model at 400 and 1000 W/m^2. The load is ranged from 10 to 90 Ω with the step-changed of 5 Ω. The PVE error, E_{pve}, is calculated using (20), which allows a more detailed view on the accuracy of the PVE. The higher the E_{pve}, the lower the accuracy of the PVE

$$E_{pve} = |I_o - I_{pv}|/I_{pv} \times 100\% \quad (20)$$

The E_{pve} is calculated at various conditions and the results are shown in Fig. 14*b*. The maximum E_{pve} for the PVEPI, PVEF, and PVES are under 1×10^{-5}%. The E_{pve} is low because there is no

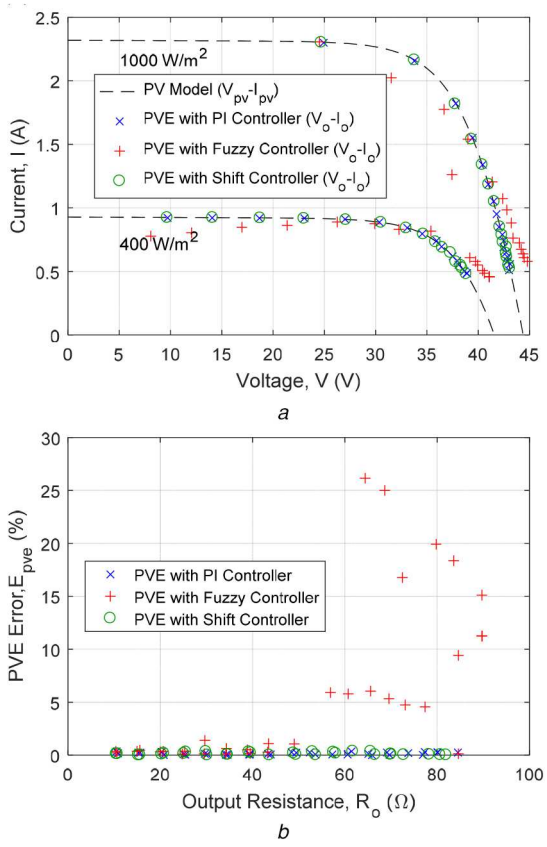


Fig. 15 *The accuracy test based on the experiment*
(a) Experimental V_o and I_o mapped on the *I–V* characteristic curve,
(b) Corresponding PVE error compared to the PV model

Table 6 T_s and the time increase of various controllers

Type	PVEPI	PVEF	PVES
load 5 Ω	3.5 ms	3.0 ms	3.5 ms
load 90 Ω	69.0 ms	10.9 ms	14.1 ms
time Increase, t_{inc}	19.7	3.6	4.0

sensors error. The computation time for the simulation is low (<1 μs), which allows the controllers to operate properly and accurately. If these results are compared with the other simulated PVE, the error is significantly lower [13, 36]. This is because the PVEPI, PVEF, and PVES use the resistance feedback control strategy, which is more accurate and stable than the conventional DRM [1].

The experimental results for all PVEs are shown in Fig. 15. Based on Fig. 15*a*, the PVEPI and PVES are able to follow the *I–V* characteristic curves of the PV model. However, the PVEF is unable to follow these curves. This is due to the very high sampling time required by the fuzzy controller, which is 518 μs. The high sampling time highly affects the ability of the PVEF to produce an accurate V_o and I_o. The maximum E_{pve} for the PVEF is 26.2%, as shown in Fig. 15*b*. This shows that the low sampling time is important to ensure the PVE is accurate. The maximum E_{pve} for PVEPI and PVES are 0.4 and 0.4%, respectively. The experimental maximum E_{pve} is higher than the simulation maximum E_{pve} because of the error caused by the sensors.

5.3 Transient response of the PVE

The transient response is also an important aspect in analysing the performance of the PVE. The PVE needs to be fast since the PV module has a fast transient response. The transient response analysis is conducted using the irradiance and load tests. The performance of the PVEs is measured using the t_s of the I_o, which is the time taken for the I_o to reach within 2% of the final value.

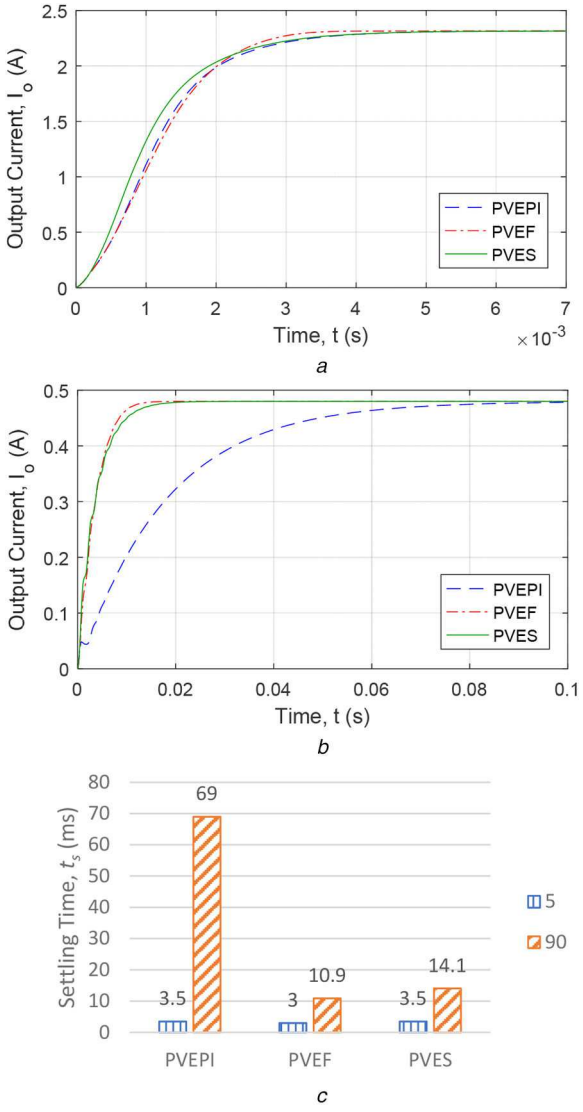


Fig. 16 *Simulated waveform of I_o based on various controllers with the load of*
(a) 5 Ω,
(b) 90 Ω at 1000 W/m^2,
(c) Corresponding t_S for 5 and 90 Ω loads

5.3.1 Robustness test: The robustness of the controller is the ability of the controller to maintain performance at various loads. The performance of the PVEs is measured using the t_s of the I_o. By referring to Figs. 16*a* and *c*, the PVEPI, PVEF, and PVES performance are almost similar, which are 3.5, 3.0, and 3.5 ms, respectively. The reason is that all of these controllers are designed according to the controller's design criteria, which ensure a fair comparison between the controllers.

A robust controller only shows a small-time increase and the results are tabulated in Table 6. Based on Figs. 16*b* and *c*, the t_s for the PVEPI, PVEF, and PVES are 69.0, 10.9, and 14.1 ms, respectively.

To quantify the robustness of the controllers, the time increase, t_{inc}, is introduced and expressed in (21). A smaller t_{inc} represents a more robust controller. By referring to Table 6, the t_{inc} for the PVEPI, PVEF, and PVES are 69.0 times, 3.6 times, and 4.0 times, respectively. This shows that the PVEF and PVES are more robust than the PVEPI.

$$t_{inc} = \frac{t_s \text{ for } 5\ \Omega}{t_s \text{ for } 90\ \Omega} \tag{21}$$

5.3.2 Irradiance test: The irradiance test determines the response of the PVE during the *G* changes. The *G* is step-changed from 400

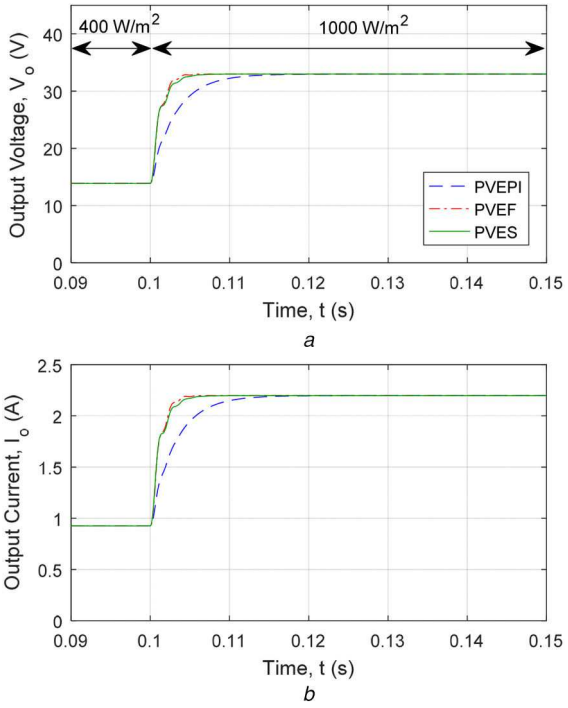


Fig. 17 *Simulated waveforms of the PVE based on the various controllers under the step-changed of the irradiance from 400 to 1000 W/m^2 with the load of 15 Ω*

(a) The waveforms of V_O,

(b) The waveforms of I_O

to 1000 W/m^2 at 0.1 s. During this test, the load and T are kept at 15 Ω and 25°C, respectively. The V_o and I_o of all PVEs are measured and the simulation results are shown in Fig. 17.

The t_s for the PVEPI, PVEF, and PVES are 10.4, 4.2 and 4.2 ms, respectively. The PVEPI response is slower than the PVEF and PVES. The PI controller is designed for a wide range of loads (from 5 to 90 Ω). This is because the load for the PVE changes based on various conditions. Since the stability margins at a low load are smaller than the stability margins at high load, the design of the PI controller is based on 5 Ω. As a result, the transient response becomes slow at 15 Ω. This is different for the PVEF and PVES, in which these controllers are robust against the load change. As a result, the transient response is kept fast even when the load is 15 Ω.

The irradiance test is conducted experimentally similar to the condition in the simulation. The PVEPI requires 10.3 ms to achieve steady-state, as shown in Fig. 18, which is only 0.1 ms different from the simulation. While the PVES is able to achieve steady-state in only 4.0 ms, which is only 0.2 ms faster than the simulation result. The results obtained from the experiment correspond to the simulation results, in which the proposed PVES is faster than the PVEPI.

The experimental result obtained from the PVEF is different from the simulation result. The simulated t_s for the PVEF is only 4.2 ms. While the experimental t_s for the PVEF is not obtained since the I_o does not follow the I_{ref} when the irradiance changes. This fail emulation process is caused by the very large t_{sam} of the PVEF (518 µs). The hardware platform does not have enough data samples for the fuzzy controller to control the PVEF during the transient state. As a result, the fuzzy controller cannot produce proper D based on the E_f. There are two approaches to overcome this problem. The first approach is by reducing the f_s. When the f_s is reduced, the L and C increase to maintain continuous current mode operation and a similar γ_{V_o}, respectively. As a result, the performance of the PVEF becomes slower. The second approach is by replacing the dSPACE ds1104 with a much faster hardware platform. The problem faced by the current hardware platform is

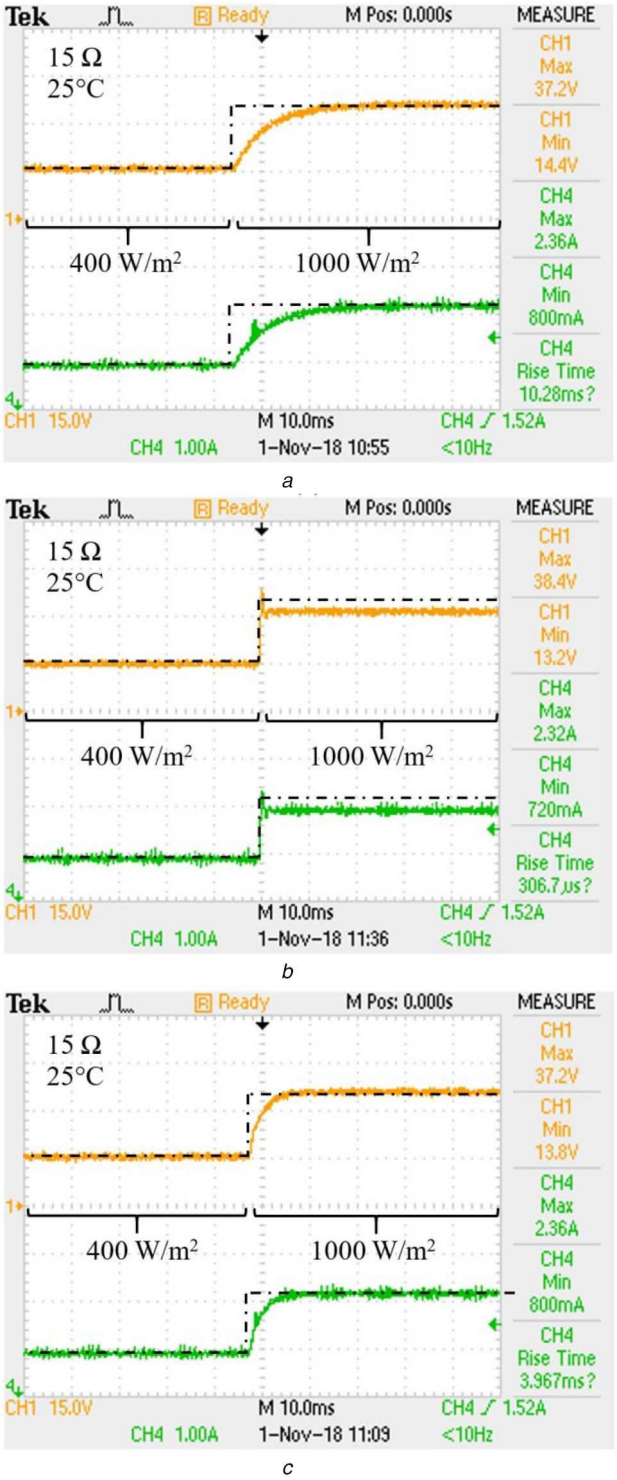


Fig. 18 *Experimental V_o and I_o of the PVE based on the various controllers under the step-changes of the irradiance from 400 to 1000 W/m^2 with a load of 15 Ω*

(a) PVEPI,

(b) PVEF,

(c) PVES

the large sample time. By using a faster hardware platform, the experimental results similar to the simulation results are obtained.

5.3.3 Load test: The load test determines the response of the PVE during the load changes. During this test, the G and T are kept constant at 1000 W/m^2 and 25°C, respectively. The load is step-changed from 10 to 60 Ω at 0.1 s. The simulation t_s for the PVEPI, PVEF, and PVES are 36.6, 6.6, and 10.2 ms, respectively. The

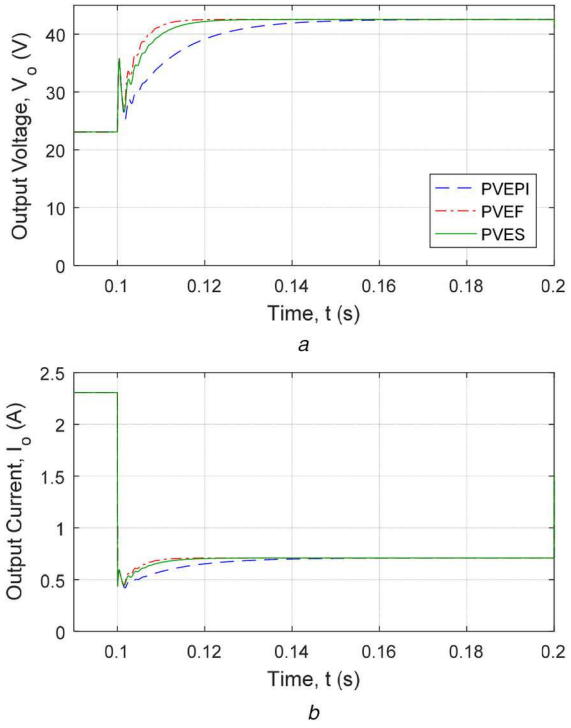


Fig. 19 *Simulated waveforms of the PVE based on the various controllers under the step-changed of the load from 10 to 60 Ω at 1000 W/m²*
(a) The waveforms of V_o,
(b) The waveforms of I_o

results presented in Fig. 19 show that the PVEPI response slower than the PVEF and PVES.

While the proposed PVES is slightly slower than the PVEF.

The experimental results for the load test are shown in Fig. 20. The experimental t_s for the PVEPI is 35.9 ms, which is 0.7 ms faster than the simulation t_s. The experimental t_s for the PVES is 15.7 ms, which is 5.5 ms lower than the simulation t_s. The experimental results also prove that the proposed PVES has a faster response during the load changes than the conventional PVEPI. A similar problem is faced by the PVEF during the load change. Due to the large t_{sam}, the fuzzy controller in the PVEF is unable to operate properly and the I_o does not follow the I_{ref}.

6 Conclusion

The PVE requires a closed-loop controller to produce a correct output voltage and current based on the input reference produces by the control strategy. The PI controller requires a low-computational capability but has a poor performance. The fuzzy logic controller has outperformed the PI controller, yet it requires a high-computation capability. The proposed shift controller is compared with the conventional PI and fuzzy logic controllers by applying it to the PVE, and the results show that the proposed PVES computes faster than the PVEPI and PVEF, respectively. Accordingly, the implemented PVES hardware is able to produce a more accurate emulation than the experimental PVEF. The PVES also features a faster transient response than the PVEPI under the test of irradiance and load change. To conclude, the proposed shift controller implemented in the PVES is simple to implement, computes at a fast rate, highly accurate, and has a fast transient response.

7 Acknowledgments

The authors would like to express gratitude to Universiti Teknologi Malaysia (UTM) for providing comprehensive library facilities and funding. Funding provided by Universiti Teknologi Malaysia Encouragement Research Grant under vote Q.J130000.2651.18J39. Lastly, thanks to colleagues who have either directly or indirectly contributed to the completion of this work.

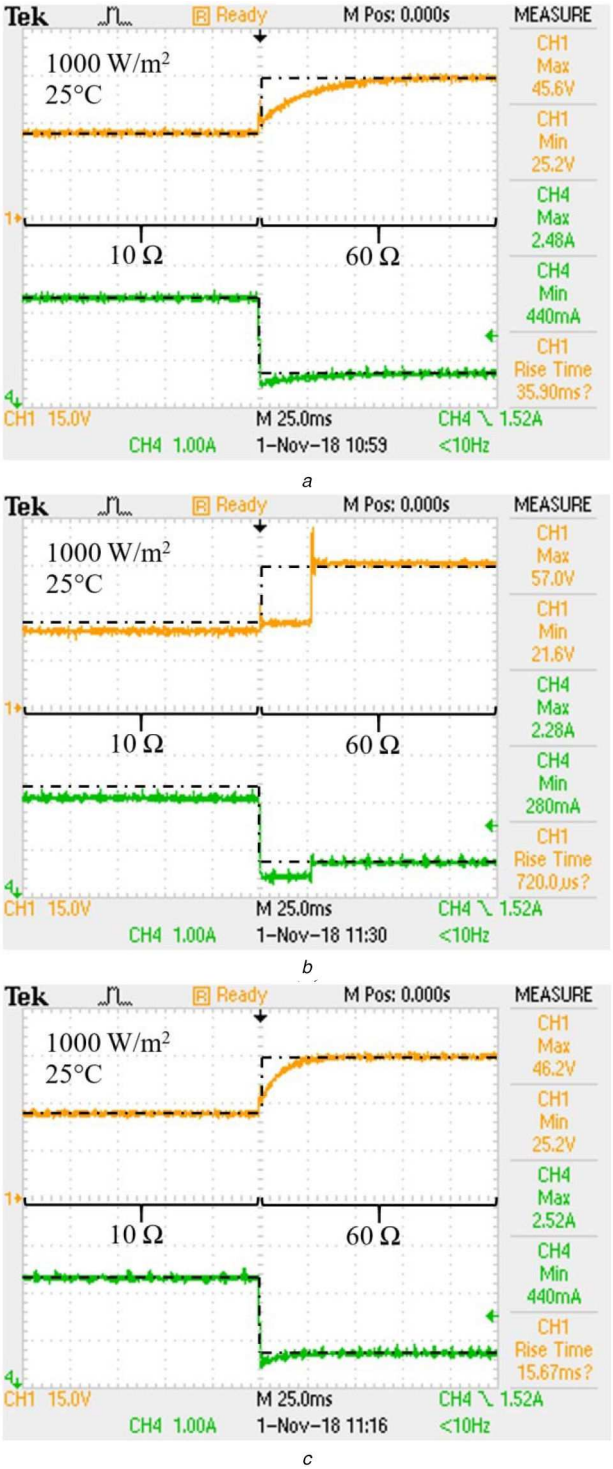


Fig. 20 *Experimental V_o and I_o of the PVE based on the various controllers under the step-changes of the load from 10 to 60 Ω at 1000 W/m²*
(a) PVEPI,
(b) PVEF,
(c) PVES

8 References

[1] Ayop, R., Tan, C.W.: 'A comprehensive review on photovoltaic emulator', *Renew. Sustain. Energy Rev.*, 2017, **80**, pp. 430–452

[2] Shahabuddin, M., Riyaz, A., Asim, M., *et al.*: 'Performance based analysis of solar PV emulators: a review'. 2018 Int. Conf. on Computational and Characterization Techniques in Engineering & Sciences (CCTES), Integral University, Uttar Pradesh. India, 2018, pp. 94–99

[3] Sampaio, L.P., Silva, S.A.O.D.: 'Graphic computational platform integrated with an electronic emulator dedicated to photovoltaic systems teaching', *IET Power Electron.*, 2017, **10**, pp. 1982–1992

[4] Cupertino, A.F., Pereira, H.A., Mendes, V.F.: 'Modeling, design and control of a solar array simulator based on two-stage converters', *J. Control, Autom. Electr. Syst.*, 2017, **58**, pp. 585–596
[5] Mai, T.D., De Breucker, S., Baert, K., *et al.*: 'Reconfigurable emulator for photovoltaic modules under static partial shading conditions', *Sol. Energy*, 2017, **141**, pp. 256–265
[6] Jin, S., Zhang, D., Wang, C.: 'UI-RI Hybrid lookup table method with high linearity and high-speed convergence performance for FPGA-based space solar array simulator', *IEEE Trans. Power Electron.*, 2018, **33**, pp. 7178–7192
[7] Ayop, R., Tan, C.W.: 'A novel photovoltaic emulator based on current-resistor model using binary search computation', *Sol. Energy*, 2018, **160**, pp. 186–199
[8] Ayop, R., Tan, C.W.: 'Rapid prototyping of photovoltaic emulator using buck converter based on fast convergence resistance feedback method', *IEEE Trans. Power Electron.*, 2019, **34**, pp. 8715–8723
[9] Ayop, R., Tan, C.W., Lau, K.Y.: 'Computation of current-resistance photovoltaic model using reverse triangular number for photovoltaic emulator application', *Indonesian J. Electr. Eng. Inf.*, 2019, **7**, pp. 314–322
[10] Shinde, U.K., Kadwane, S.G., Keshri, R.K., *et al.*: 'Dual mode controller-based solar photovoltaic simulator for true PV characteristics', *Can. J. Electr. Comput. Eng.*, 2017, **40**, pp. 237–245
[11] González-Medina, R.P., Garcerá, I., Figueres, G.: 'A low-cost photovoltaic emulator for static and dynamic evaluation of photovoltaic power converters and facilities', *Prog. Photovolt., Res. Appl.*, 2012, **22**, pp. 227–241
[12] Barrera, L.M., Osorio, R.A., Trujillo, C.L.: 'Design and implementation of electronic equipment that emulates photovoltaic panels'. 2015 IEEE 42nd Photovoltaic Specialist Conf. (PVSC), New Orleans, USA, 2015, pp. 1–5
[13] Atoche, A.C., Castillo, J.V., Ortegón-Aguilar, J., *et al.*: 'A high-accuracy photovoltaic emulator system using ARM processors', *Sol. Energy*, 2015, **120**, pp. 389–398
[14] Cruz, R.D., Rajesh, M.: 'Half bridge LLC resonant DC–DC converter for solar array simulator application'. 2015 Int. Conf. on Technological Advancements in Power and Energy (TAP Energy), Kerala, India, 2015, pp. 138–143
[15] Chang, C.-H., Chang, E.-C., Cheng, H.-L.: 'A high-efficiency solar array simulator implemented by an LLC resonant DC–DC converter', *IEEE Trans. Power Electron.*, 2013, **28**, pp. 3039–3046
[16] Nguyen-Duy, K., Knott, A., Andersen, M.A.E.: 'High dynamic performance nonlinear source emulator', *IEEE Trans. Power Electron.*, 2016, **31**, pp. 2562–2574
[17] Chen, C.-C., Chang, H.-C., Kuo, C.-C., *et al.*: 'Programmable energy source emulator for photovoltaic panels considering partial shadow effect', *Energy*, 2013, **54**, pp. 174–183
[18] Balakishan, C.H., Sandeep, N.: 'Development of a microcontroller based PV emulator with current controlled DC–DC buck converter', *Int. J. Renew. Energy Res.*, 2014, **4**, p. 1
[19] Viglus, F.J., Casaro, M.M.: 'Photovoltaic array emulation using a three-phase DC–DC converter with galvanic isolation'. 2016 12th IEEE Int. Conf. on Industry Applications (INDUSCON), Curitiba, Brazil, 2016, pp. 1–8
[20] Alaou, M., Maker, H., Mouhsen, A., *et al.*: 'Emulation Of different photovoltaic materials and technologies using PV array emulator with linear quadratic regulator', *Int. J. Sci. Technol. Res.*, 2019, **8**, pp. 745–749
[21] Remache, S.E.I., Cherif, A.Y., Barra, K.: 'Optimal cascaded predictive control for photovoltaic systems: application based on predictive emulator', *IET Renew. Power Gener.*, 2019, **13**, pp. 2740–2751
[22] Lindiya, S.A., Vijayarekha, K., Palani, S.: 'Deterministic LQR controller for dc–dc buck converter'. 2016 Biennial Int. Conf. on Power and Energy Systems: Towards Sustainable Energy (PESTSE), University of Bengaluru, India, 2016, pp. 1–6
[23] Zhang, J., Wang, S., Wang, Z., *et al.*: 'Design and realization of a digital PV simulator with a push-pull forward circuit', *J. Power Electron.*, 2014, **14**, pp. 444–457
[24] Shao, W., Meng, Z.Q., Zhou, H.A., *et al.*: 'A photovoltaic array simulator based on current feedback fuzzy PID control', *J. Intell. Fuzzy Syst.*, 2015, **29**, pp. 2555–2564
[25] Chariag, D., Sbita, L.: 'Design and simulation of photovoltaic emulator'. 2017 Int. Conf. on Green Energy Conversion Systems (GECS), Hammamet, Tunisia, 2017, pp. 1–6
[26] Sarvi, M., Ahmadi, S., Abdi, S.: 'A PSO-based maximum power point tracking for photovoltaic systems under environmental and partially shaded conditions', *Prog. Photovolt., Res. Appl.*, 2015, **23**, pp. 201–214
[27] Moussa, I., Bouallegue, A., Khedher, A.: 'Development of a low cost PV simulator based on FPGA technology'. 2017 Int. Conf. on Green Energy Conversion Systems (GECS), Hammamet, Tunisia, 2017, pp. 1–6
[28] Erkaya, Y., Moses, P., Flory, I., *et al.*: 'Steady-state performance optimization of a 500 kHz photovoltaic module emulator'. 2017 IEEE 44th Photovoltaic Specialist Conf. (PVSC), Washington D.C., USA, 2017, pp. 1–4
[29] Solar, A.: '*Ameresco solar 80 W (24 V) photovoltaic modules – 80J-B (24 V)*' (Ameresco Inc., 2014)
[30] Hart, D.W.: '*Power electronics*' (Tata McGraw-Hill Education, Valparaiso University, Indiana, 2011)
[31] Ayop, R., Tan, C. W., Lim, C.S.: 'The resistance comparison method using integral controller for photovoltaic emulator', *Int. J. Power Electron. Drive Syst.*, 2018, **9**, pp. 820–828
[32] Ugale, C.P., Dixit, V.V.: 'Buck-boost converter using fuzzy logic for low voltage solar energy harvesting application'. 2017 11th Int. Conf. on Intelligent Systems and Control (ISCO), Coimbatore, India, 2017, pp. 413–417
[33] Lorenzo, J., Espiritu, J.C., Mediavillo, J., *et al.*: 'Development and implementation of fuzzy logic using microcontroller for buck and boost DC-to-DC converter'. IOP Conf. Series: Earth and Environmental Science, Chengdu, China, 2017
[34] Al-Nussairi, M.K., Bayindir, R., Hossain, E.: 'Fuzzy logic controller for Dc-Dc buck converter with constant power load'. 2017 IEEE 6th Int. Conf. on Renewable Energy Research and Applications (ICRERA), San Diego, USA, 2017, pp. 1175–1179
[35] Ganeswari, J.A., Kiranmayi, R.: 'Performance improvement for DC buck converter with fuzzy controller', *Int. J. Pure. Appl. Math.*, 2018, **118**, pp. 27–32
[36] Moussa, I., Khedher, A.: 'Photovoltaic emulator based on PV simulator RT implementation using XSG tools for an FPGA control: theory and experimentation', *Int. Trans. Electr. Energy Syst.*, 2019, **29**, p. e12024, https://onlinelibrary.wiley.com/doi/abs/10.1002/2050-7038.12024