Design of PV Model-based Solar Array Simulator Engine using Optimization Method

Young-Tae Seo, Thusitha Wellawatta, and Sung-Jin Choi, *Member, IEEE*
School of Electrical Engineering, University of Ulsan, Ulsan, South Korea
hot5752@naver.com, trwellawatta@gmail.com, sjchoi@ulsan.ac.kr

***Abstract* — Photovoltaic (PV) model can be a viable alternative to the conventional look-up-table as an accurate and versatile solar array simulator (SAS) engine. In PV model-based SAS, PV model has a critical role to generate appropriate I-V characteristic of the PV panel under rapidly varying temperature and irradiation, and thus calculation speed as well as accuracy are key performances. In this paper, a novel algorithm that is suitable for such a SAS engine is proposed. The suggested method adopts conjugate gradient optimization to extract PV model parameters from the changing conditions and to reconstruct the exact I-V curve very rapidly. For the verification, the proposed algorithm is compared with conventional ones which have been widely used in the PV model extraction. As a result, the proposed model shows superior calculation speed with good accuracy.**

Keywords—Solar Array Simulator, Optimization Method, Conjugate Gradient Method, PV Panel Modeling.

I. INTRODUCTION

In the development of the power conditioning system (PCS) for photovoltaic (PV) generations, solar array simulator (SAS) is very useful because real PV panels are difficult to handle and this not suitable for a sequence of repeated tests. Basic concept of the SAS is shown in Fig. 1 (a): In the SAS engine, PV characteristic can be generated either by a look-up-table (LUT) or by a PV model (PVM). In a LUT-based SAS, a lot of curve data are collected from the experiments on real PV panels and stored to LUT in advance. However, the limited information available in the table, most of the PV characteristics needs to be generated by interpolation of existing data and that may cause inaccuracy of the SAS. On the other hand, PVM can be readily constructed from datasheet supplied with PV panels and thus PVM-based SAS has a lot of advantage over the conventional scheme.

In PVM-based SAS, single-diode model shown in Fig. 2 is widely accepted as a PVM and two steps are performed inside the SAS engine. With an appropriate model extraction algorithm, the PV model parameters are adjusted according to varying conditions and thus speed of the algorithm is especially important to guarantee the fast response of the SAS.

In this paper, an effective I-V curve extraction method for SAS engine is proposed. This model adopts a conjugate gradient optimization method that uses gradient information of a model error function both to retain the accuracy and to improve the speed. To verify validity of the proposed model, comparisons with conventional methods [1],[2] are presented.

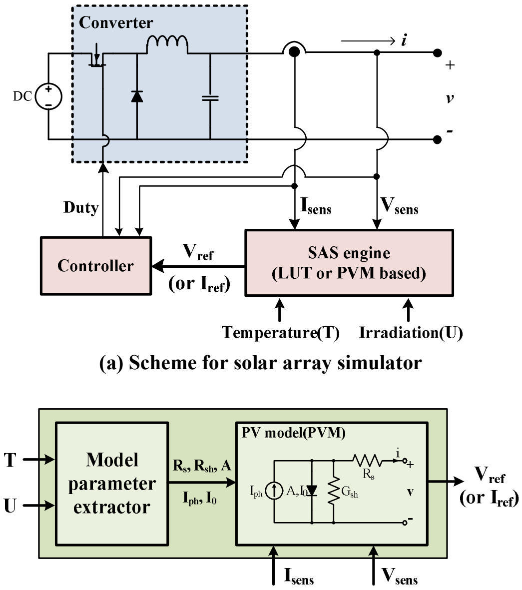


Fig. 1. Solar array simulator (SAS)

II. PROPOSED METHOD

1) Problem definition

I-V characteristics of PV panels changes under varying ambient temperature and irradiation levels. Single-diode PV model in Fig.2 describes the characteristics with an equation given by

$$i = I_{ph} - I_o(e^{\frac{v+iR_s}{N_S A V_T}} - 1) - (v + iR_s)G_{sh} \qquad (1)$$

The proposed method starts from an update process that has been introduced in [2] and [3] where such critical points are updated according to the external conditions. In other words, new datasheet values can be found as follows

$$I_{sc} = I_{sc,STC}\frac{U}{U_{STC}}[1 + k_i(T - T_{STC})] \quad (2)$$

$$V_{oc} = V_{oc,STC} + N_s A_{STC} V_T \ln(\frac{U}{U_{STC}}) + k_v(T - T_{STC}) \quad (3)$$

$$I_{mpp} = I_{mpp,STC}\frac{U}{U_{STC}}[1 + k_i(T - T_{STC})] \quad (4)$$

$$V_{mpp} = V_{mpp,STC} + N_s A_{STC} V_T \ln(\frac{U}{U_{STC}}) + k_v(T - T_{STC}) \quad (5)$$

From the modified datasheet values, model parameter can be extracted using optimization algorithm. The objective is to minimize the errors in two essential conditions that model should obey: one is the maximum power point (MPP) condition in I-V curve and the other is the null slope condition in P-V curve [3]. From the equation of single-diode model in eq. (1), eq. (6) describes the MPP condition

$$I_{ph} - I_o e^{\frac{V_{mpp}+I_{mpp}R_s}{N_s A V_T}} - (V_{mpp} + I_{mpp}R_s)G_{sh} - I_{mpp} = 0 \quad (6)$$

and eq. (7) represents the null-slope condition.

$$\left.\frac{dp}{dv}\right|_{@mpp} = I_{mpp} - V_{mpp}\frac{G_{sh}\left(\frac{(I_{sc}/G_{sh} - V_{oc} + I_{sc}R_s)}{N_s A V_T}e^{\frac{V_{mpp}+I_{mpp}R_s-V_{oc}}{N_s A V_T}} + 1\right)}{1 + R_s G_{sh}\left(\frac{(I_{sc}/G_{sh} - V_{oc} + I_{sc}R_s)}{N_s A V_T}e^{\frac{V_{mpp}+I_{mpp}R_s-V_{oc}}{N_s A V_T}} + 1\right)} = 0 \quad (7)$$

By the sum of square of the above two equations, objective function is defined as follows.

$$E(\boldsymbol{X}) = E(R_s, G_{sh}, A) \equiv f^2(R_s, G_{sh}, A) + g^2(R_s, G_{sh}, A) \quad (8)$$

Now, the three of the five PV model parameters- R_s, G_{sh}, and A – can be obtained by the following optimization problem:

$$\begin{aligned} &\text{Minimize } E(X) \\ &\text{Subject to: } 0 \le R_s \le \frac{V_{oc} - V_{mpp}}{I_{mpp}} \\ &\qquad 0 \le G_{sh} \le \frac{I_{sc} - I_{mpp}}{V_{mpp}} \\ &\qquad 0 \le A \le 2 \end{aligned} \quad (9)$$

The other two parameters, I_{ph} and I_o, can be obtained by simply solving the following simultaneous equations.

$$I_{ph} = I_o e^{\frac{qV_{oc}}{N_s AkT}} + V_{oc}G_{sh} \quad (10)$$

$$I_o = \left[I_{sc} - G_{sh}(V_{oc} - I_{sc}R_s)\right]e^{-\frac{qV_{oc}}{N_s AkT}} \quad (11)$$

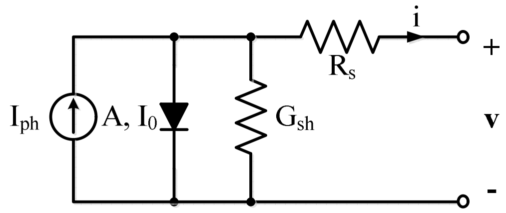


Fig. 2. Single-diode model for PV panel

Step 1: Measure ***Temperature (T) & Irradiation (U)***
Update ***Datasheet values using Eq. (2)~(5)***
Define $\boldsymbol{X_1} = (R_{s1}, G_{sh1}, A_1)^T$
Define $\boldsymbol{N_{max}}$ (maximum iteration)
Define $\boldsymbol{\varepsilon}$ (tolerance)

Step 2: *If* $i = 1$
$\boldsymbol{S_1} = -\nabla E(\boldsymbol{X_1})$
$\boldsymbol{X_2} = \boldsymbol{X_1} + \alpha_1 \boldsymbol{S_1}$
→ α_i is determined by minimizing $E(\boldsymbol{X_2})$
Go to step 4

Step 3: *for* $i = 2 : N_{max}$
$\beta_i = \frac{\nabla E(\boldsymbol{X_i})^T \nabla E(\boldsymbol{X_i})}{\nabla E(\boldsymbol{X_{i-1}})^T \nabla E(\boldsymbol{X_{i-1}})}$
$\boldsymbol{S_i} = -\nabla E(\boldsymbol{X_i}) + \beta_i \boldsymbol{S_{i-1}}$
$\boldsymbol{X_{i+1}} = \boldsymbol{X_i} + \alpha_i \boldsymbol{S_i}$
→ α_i is determined by minimizing $E(\boldsymbol{X_{i+1}})$
Go to step 4

Step 4: Calculate $\Delta E = E(\boldsymbol{X_{i+1}}) - E(\boldsymbol{X_i})$
Calculate $\Delta \boldsymbol{X} = \boldsymbol{X_{i+1}} - \boldsymbol{X_i}$

If $|\Delta E| \le \varepsilon$; ***Go to step 5***
If $\Delta \boldsymbol{X}^T \Delta \boldsymbol{X} \le \varepsilon$; ***Go to step 5***
If iteration = N_{max}; ***Go to step 5***
If $\nabla E(\boldsymbol{X_{i+1}})^T \nabla E(\boldsymbol{X_{i+1}}) \le \varepsilon$; ***Go to step 5***
Else ***Go to step 3***

Step 5: Extract $\boldsymbol{R_s}$, $\boldsymbol{G_{sh}}$, and $\boldsymbol{A}$ Calculate $\boldsymbol{I_{ph}}$, $\boldsymbol{I_o}$
Calculate ***I-V characteristic*** using Eq. (1)
Generate of $\boldsymbol{I_{ref}}$ (or $\boldsymbol{V_{ref}}$)

Fig. 3. Pseudo code of proposed method

2) *Model parameter extraction algorithm*

In the proposed algorithm, to find the optimal value of E(***X***), conjugate gradient method is adopted. In n-variable case, it spends only n-iteration steps in the best case and, consequently, its convergence speed becomes very fast [4].

Pseudo code of the SAS engine algorithm including the proposed algorithm is shown in Fig. 3. In the initialization step, parameter vector (***X***) is initialized as

$$\boldsymbol{X_1} = [\frac{V_{oc} - V_{mpp}}{2I_{mpp}}, \frac{I_{sc} - I_{mpp}}{2V_{mpp}}, 1] \quad (12)$$

and the maximum the allowable iteration number (N_{max}) and allowable tolerance for algorithm termination (ε) are defined. For the first iteration step, initial search vector, $\boldsymbol{S_1}$ is set to the gradient of $\boldsymbol{X_1}$ and the next parameter vector, $\boldsymbol{X_2}$ is found by solving 1-D sub-problem where golden section search is suitable for this purpose. In the subsequent steps, conjugate coefficient

$$\beta = \frac{\nabla E(\boldsymbol{X}_i)^T \nabla E(\boldsymbol{X}_i)}{\nabla E(\boldsymbol{X}_{i\text{-}1})^T \nabla E(\boldsymbol{X}_{i\text{-}1})} \tag{13}$$

is added to update the next search vector as follows.

$$\boldsymbol{S}_i = -\nabla E(\boldsymbol{X}_i) + \beta \cdot \boldsymbol{S}_{i\text{-}1} \tag{14}$$

Searching along the search direction, next solution ($\boldsymbol{X_{i+1}}$) that minimizes $E(\boldsymbol{X_{i+1}})$ is determined by solving another 1-D sub-problem.

$$\boldsymbol{X}_{i+1} = \boldsymbol{X}_i + \alpha \cdot \boldsymbol{S}_i \tag{15}$$

In every iteration step, termination conditions in Fig. 3 are tested. After finishing the optimization, every model parameter is identified by calculating I_o and I_{ph}, and the reference for PWM controller is generated.

III. PERFORMANCE VERIFICATION

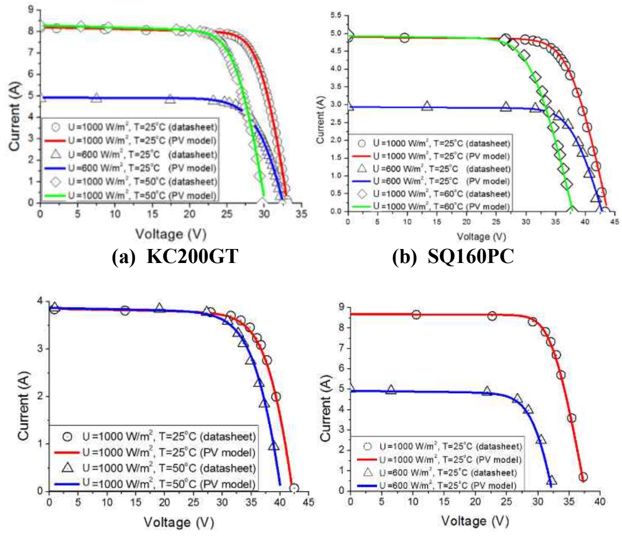
For checking the feasibility of the proposed SAS engine algorithm, Parameter extraction under the different temperature and irradiation condition has been performed by a MATLAB m-script. Fig. 4 shows the resulting I-V curves generated from the proposed PV engine, and the results are compared with a few measurement curves found in the panel datasheet.

To closely investigate the performance, four PV panels with different power ratings from various vendors are examined together. Besides the proposed method, conventional root finding approach in [1] and PSO method in [2] are also simulated.

In view of the model accuracy, I-V curve error is analyzed according to the following definition in compliant with EN50530 standard [5].

$$\varepsilon_I(\%) = \frac{1}{0.2V_{mpp}} \int_{V_{mpp}\pm 10\%} \left| \frac{i_s(v) - i_m(v)}{i_m(v)} \right| dv \times 100 \tag{16}$$

Table 1 shows accuracy results and the proposed model shows superior performance with PV panels to the conventional methods. Table 2 compares algorithm speed and the proposed model always shows the best performance both in the iteration step counts and the computation time. Consequently, it is concluded that the proposed algorithm is most suitable for PVM-based SAS engine.



(a) KC200GT **(b) SQ160PC** **(c) BP-MSX120** **(d) TMS245PC**

Fig. 4. I-V curve generation

IV. CONCLUSIONS

This paper presents a rapid engine algorithm for PVM-based SAS. The proposed method adopts a conjugate gradient optimization to extract the parameters under various operating conditions. Pseudo-codes are also presented for easy implementation. Verification process with different PV panels shows that this method reconstructs I-V curve with fast speed as well as good accuracy. With a subsequent hardware verification, it is expected that the proposed SAS engine will replace the conventional LUT-based SAS engine.

REFERENCES

[1] D. Sera, R. Teodorescu, and P. Rodriguez, "PV Panel Model Based on Datasheet Values," IEEE International Symposium on Industrial Electronics, pp. 2392-2396, 2007.

[2] J.J. Soon and K. S. Low, "Photovoltaic Model Identification Using Particle Swarm Optimization With Inverse Barrier Constraint," IEEE Transactions on Power Electronics, Vol 27, No. 9, Sept., 2012.

[3] Jun-Young Park and Sung-Jin Choi, "A New PSIM Model for PV Panels Employing Datasheet-based Parameter Tuning," The Transactions of Korea Institute of Power Electronics (KIPE), Vol. 20. No. 6, Dec., 2015.

[4] P. Venkataraman, *Applied Optimization with MATLAB Programming*, 2nd edition: Wiley, 2009.

[5] *IEC EN50530*, Standard for Overall Efficiency of Photovoltaic Inverters, CENELEC, Stassart 35, B-1050 Brussels.

Table 1. Model error

	MSX120	SQ160PC	KC200GT	TMS245PC
[1]	1.28	1.19	0.87	2.14
[2]	1.37	1.75	1.49	0.93
Proposed	1.16	1.02	1.36	1.07

Table 2. Calculation speed

	MSX120		SQ160PC		KC200GT		TMS245PC	
	Step	Time (sec.)	Step	Time (sec.)	Step	Time (sec.)	Step	Time (sec.)
[1]	67	9.55	83	11.78	33	5.15	44	6.82
[2]	10	0.44	10	0.36	10	0.91	10	0.33
Proposed	3	0.06	10	0.05	11	0.06	4	0.08