A Rapid I-V Curve Generation for PV Model-based Solar Array Simulators

Young-Tae Seo, Jun-Young Park, and Sung-Jin Choi, *Member, IEEE*
School of Electrical Engineering, University of Ulsan, Ulsan, South Korea
hot5752@naver.com, pjy10612@naver.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 its 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.**

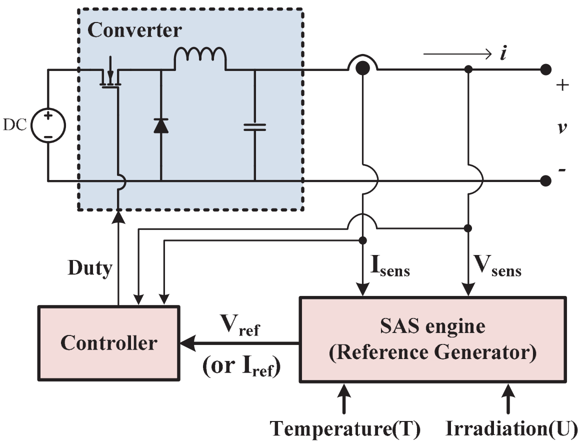
Keywords*— Solar Array Simulator, Conjugate Gradient Method, PV Panel Modeling.

I. Introduction

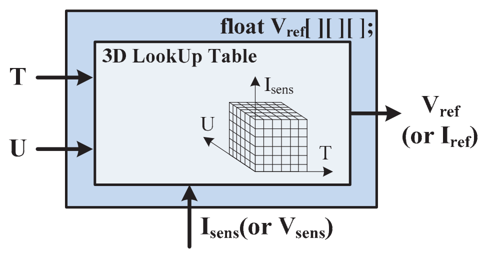
In the development of the power conditioning system (PCS) for photovoltaic (PV) generations, solar array simulator (SAS) is very useful because real PV panels are difficult to handle and are not suitable for a sequence of repeated tests. Basic concept of the SAS is shown in Fig. 1 (a): it consists of dc power source, dc/dc converter, PWM controller, and engine. The engine is usually implemented on a personal computer or an embedded system and emulates PV panel to generate the reference for the PWM controller according to different irradiation and temperature profiles. 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 3-dimensional LUT in advance as shown in Fig. 1(b). To reconstruct I-V curve by accessing the LUT generally shows fast response in a runtime environment. However, the finite number of index available in the LUT may require interpolation of existing data and that may cause inaccuracy of the SAS. That is, there is a trade-off between accuracy and the memory space involved. Moreover, changing panels requires re-generation of the table.

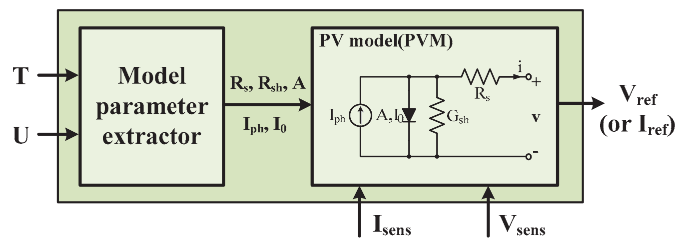
On the other hand, PVM has a lot of advantages over the LUT-based approaches in I-V curve generation. First, I-V curves can be readily constructed from the governing equation of PVM in real time, and thus it consumes less memory space. Second, there are many algorithm candidates to extract PVM



(a) Scheme for solar array simulator



(b) Look-up-table(LUT) based SAS engine



(c) PV model(PVM) based SAS engine

Fig. 1. Solar array simulator (SAS)

978-1-5090-0737-0/16/$31.00 ©2016 IEEE

parameters from three critical corner points – open-circuited, short-circuited, and the maximum power point in the PV panel datasheets [1-3]. In PVM-based SAS, single-diode model is widely accepted as a PVM and two steps are performed inside the SAS engine as shown in Fig. 1(c). PV model parameters are continuously extracted from the environmental profile and then reference value for the SAS control is generated according to the equations governing the PVM. With an appropriate model extraction algorithm, the PV model parameters should be adjusted according to varying conditions and thus speed of the algorithm is especially important to guarantee the fast response of the SAS.

Regarding such an algorithm, it is necessary to consider various parameter extraction methods have been studied in the literature. In [1], Newton-Raphson or bi-section root finding algorithm is used for this purpose. In spite of its good accuracy, sequential determination of the model parameters makes it operate relatively slowly. Consequently, it is difficult to adopt this algorithm in SAS engine that needs fast speed. Recently, a particle swarm optimization (PSO) is introduced for the PV model parameter determination [2]. However, it is based on random trial and error and thus its accuracy and calculation speed is inconsistent and heavily dependent on the particle size involved. Therefore, it is essential to investigate fast PV model construction algorithms suitable for PVM-based SAS engine.

In this paper, an effective I-V curve extraction method for SAS engine is proposed. This scheme 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 approach, comparisons with conventional methods are presented.

II. PROPOSED MEHTOD

A. 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

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

In order to explain the changing characteristics, the PV model parameters - R_s, G_{sh}, A, I_{ph}, and I_o – should be adjusted according to varying conditions. For this adjustment, it is desirable to utilize datasheet information which contains four critical points in the I-V curve - the short circuit current, I_{sc}, the open circuit voltage, V_{oc}, the maximum power current, I_{mpp}, and the maximum power voltage, V_{mpp}. However, most PV panel vendors provide that information measured only in standard test condition (STC) of 1000W/m^2 and 25^{o}C.

Therefore, the proposed method starts from an update process that has been introduced in [2] and [3] where such

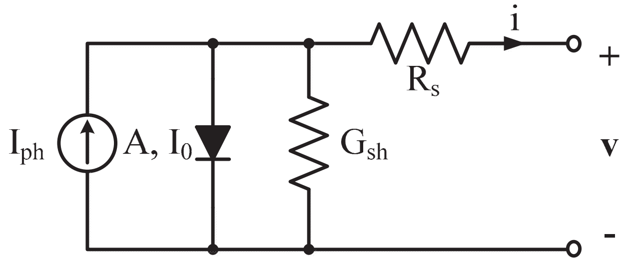


Fig. 2. Single-diode model for PV panel

critical points are updated according to the external conditions. In other words, modified datasheet values can be found as follows:

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

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

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

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

where each symbol with subscript 'STC' means that value in STC and k_i and k_v are the temperature coefficients for the current and voltage thermal drift, all of which are specified in datasheet.

From the modified datasheet values, model parameter can be extracted using optimization algorithm. First of all, from the open circuit and short circuit conditions, the following equitation can be obtained.

$$I_{ph} = I_o e^{\frac{qV_{oc}}{N_S AkT}} + V_{oc}G_{sh} \qquad (6)$$

$$I_o = \left[I_{sc} - G_{sh}(V_{oc} - I_{sc}R_s)\right]e^{-\frac{qV_{oc}}{N_S AkT}} \qquad (7)$$

Now, two additional conditions, the maximum power point (MPP) condition in I-V curve and the null slope condition in P-V curve, can be incorporated to construct a minimization problem. From the equation of single-diode model in eq. (1), eq. (8) describes the MPP condition.

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_1 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 the proposed method

$$I_{ph} - I_o e^{\frac{V_{mpp} + I_{mpp} R_s}{N_s A V_T}} - (V_{mpp} + I_{mpp} R_s) G_{sh} - I_{mpp} = 0 \qquad (8)$$

The left hand of eq. (8) is represented by a function of R_s, G_{sh}, and A, and is denoted as $f(R_s, G_{sh}, A)$. Meanwhile, the incremental output power can be decomposed into

$$\frac{dp}{dv} = i + v\frac{di}{dv}, \qquad (9)$$

and thus the null-slope condition is represented by

$$\left.\frac{dp}{dv}\right|_{@mpp} = I_{mpp} - V_{mpp} \frac{G_{sh}\left(\frac{(I_{sc}/G_{sh} - V_{oc} + I_{sc}R_s)}{N_s A V_T} e^{\frac{V_{mpp} + I_{mpp}R_s - V_{oc}}{N_s A V_T}} + 1\right)}{1 + R_s G_{sh}\left(\frac{(I_{sc}/G_{sh} - V_{oc} + I_{sc}R_s)}{N_s A V_T} e^{\frac{V_{mpp} + I_{mpp}R_s - V_{oc}}{N_s A V_T}} + 1\right)} = 0 \qquad (10)$$

where the left hand side is denoted as $g(R_s, G_{sh}, A)$. By the sum of square of the above two equations, error function is defined as follows.

$$E(\boldsymbol{X}) = E(R_s, G_{sh}, A) \equiv f^2(R_s, G_{sh}, A) + g^2(R_s, G_{sh}, A) \qquad (11)$$

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

$$\begin{aligned} &\textit{Minimize } E(\boldsymbol{X}) \\ &\textit{Subject to} : 0 \le R_s \le \frac{V_{oc} - V_{mpp}}{I_{mpp}} \\ &\qquad 0 \le G_{sh} \le \frac{I_{sc} - I_{mpp}}{V_{mpp}} \\ &\qquad 0 < A \le 2 \end{aligned} \qquad (12)$$

B. Model parameter extraction algorithm

In the proposed algorithm, to find the optimal value of E(X), conjugate gradient method is adopted. Because it utilizes gradient of the objective function to divide a multi-dimensional problem into one-dimensional (1-D) minimization problems, it has great advantage in multi-variate problem: for example, in n-variable case, it spends only n-iteration steps in the best case and, consequently, its convergence speed becomes very fast [4]. Therefore, it is suitable for extracting the parameters of the PV model that has more than three variables.

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

$$\boldsymbol{X_1} = \left[\frac{V_{oc} - V_{mpp}}{2I_{mpp}}, \frac{I_{sc} - I_{mpp}}{2V_{mpp}}, 1\right] \qquad (13)$$

and the maximum allowable iteration number (N_{max}) and allowable tolerance for algorithm termination (ε) are defined. For the first iteration step, initial search vector, S_1 is set to the gradient of the error function evaluated at X_1 and the next parameter vector, X_2 is set to a scaled linear composition of X_1 and S_1, where the appropriate scale factor, α, of the linear composition is found by solving 1-D sub-problem of minimizing $E(X_2)$. For such a sub-problem, golden section search algorithm has been adopted. In the subsequent steps, conjugate coefficient

$$\beta_i = \frac{\nabla E(\boldsymbol{X}_i)^T \nabla E(\boldsymbol{X}_i)}{\nabla E(\boldsymbol{X}_{i\text{-}1})^T \nabla E(\boldsymbol{X}_{i\text{-}1})} \tag{14}$$

is added to the update equation of the next search vector as follows.

$$\boldsymbol{S}_i = -\nabla E(\boldsymbol{X}_i) + \beta_i \boldsymbol{S}_{i\text{-}1} \tag{15}$$

Searching along the search direction, next solution (X_{i+1}),

$$\boldsymbol{X}_{i+1} = \boldsymbol{X}_i + \alpha_i \boldsymbol{S}_i \tag{16}$$

that minimizes $E(X_{i+1})$ is determined by optimizing the scale factor, α, repeatedly. In every iteration step, four termination conditions are tested. After finishing the optimization, every model parameter is identified by calculating I_o and I_{ph}.

III. Performance Verification

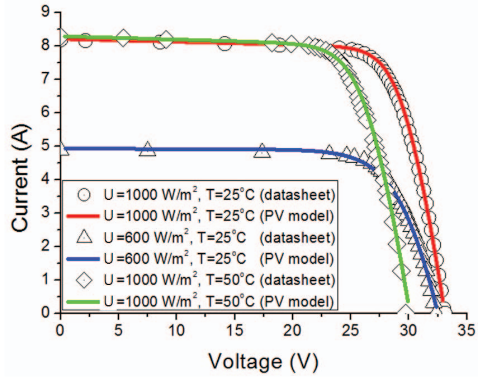
For checking the feasibility of the proposed PVM model construction algorithm, PV model parameters are extracted from various sample PV panels. To closely investigate the performance, four PV panels (MSX120, SQ160PC, KC200GT, TSM245PC) with different power ratings from various vendors are examined together. During the modeling process, critical operating points - V_{oc}, I_{sc}, I_{mpp}, V_{mpp} specified under STC, and the temperature coefficients k_i, and k_v – are used, and all of them can be easily accessible from the PV datasheet. The tolerance for the algorithm termination is set to $\varepsilon = 1\times10^{-8}$. It has been performed by a MATLAB m-script on Intel i5 760 2.80 GHz processor.

Fig. 4 shows the resulting I-V curves generated from the proposed PV engine, and the results are compared with the measurement curves found in the panel datasheet. It is clear that the model accuracy is very high even under the different temperature and irradiation condition. In Fig.4 (c) and (d), either temperature or irradiation change is considered because of lack of information in the datasheets.

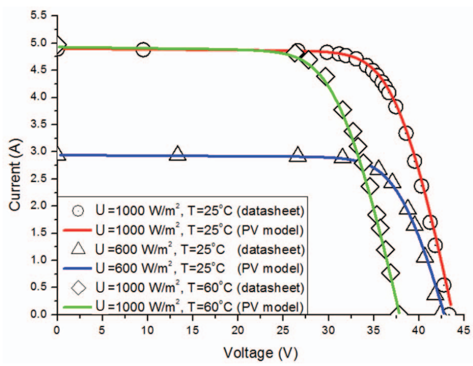
Besides the proposed method, conventional root finding approach in [1] and PSO method in [2] are also simulated in the same computing environment. In PSO method, number of the random samples and trials are set to 20 and 10, respectively. For a fair comparison, the same objective function in eq. (11) is used also for PSO method. In order to compare the model accuracy, current error in the I-V curve is analyzed according to the following definition in compliant with EN50530 standard [5].

$$\varepsilon_I(\%) = \frac{1}{0.2V_{mpp}} \int_{V_{mpp}\pm10\%} \left| \frac{i_s(v) - i_m(v)}{i_m(v)} \right| dv \times 100 \tag{17}$$

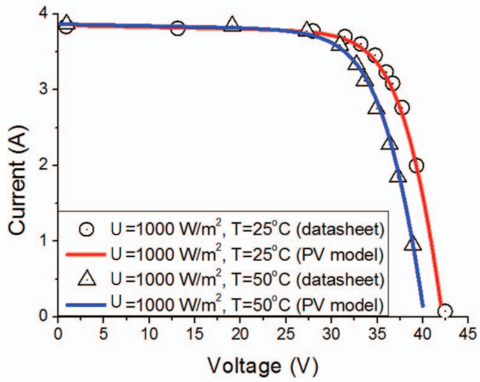
where the subscript 'm' stands for the measurement value obtained from datasheet curve and the subscript 's' denotes the value is the calculated, respectively. Table I and Fig. 5(a)(b)



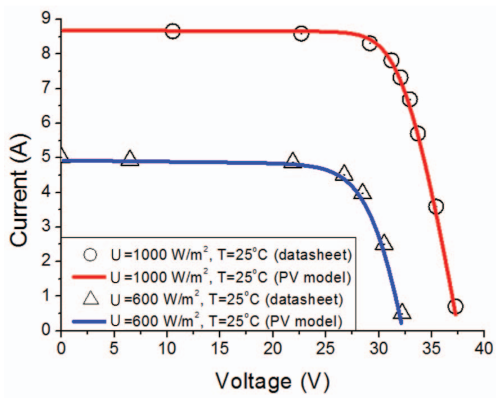
(a) KC200GT



(b) SQ160PC



(c) BP-MSX120



(d) TMS245PC

Fig. 4. I-V curve generation

TABLE I. CURRENT 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

show the accuracy comparison results. The proposed model shows superior performance with MSX120 and SQ160PC panels to the conventional methods. Despite of slightly higher error with KC200GT and TMS245PC, the proposed method is sufficiently competitive in model accuracy. It is also remarkable from the spider chart in Fig 5(b) that the accuracy from the proposed method is not highly dependent on the panel difference, whereas the other methods show some dependency on the panel characteristics. Table II and Fig. 5(c)(d) shows algorithm speed. 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.

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

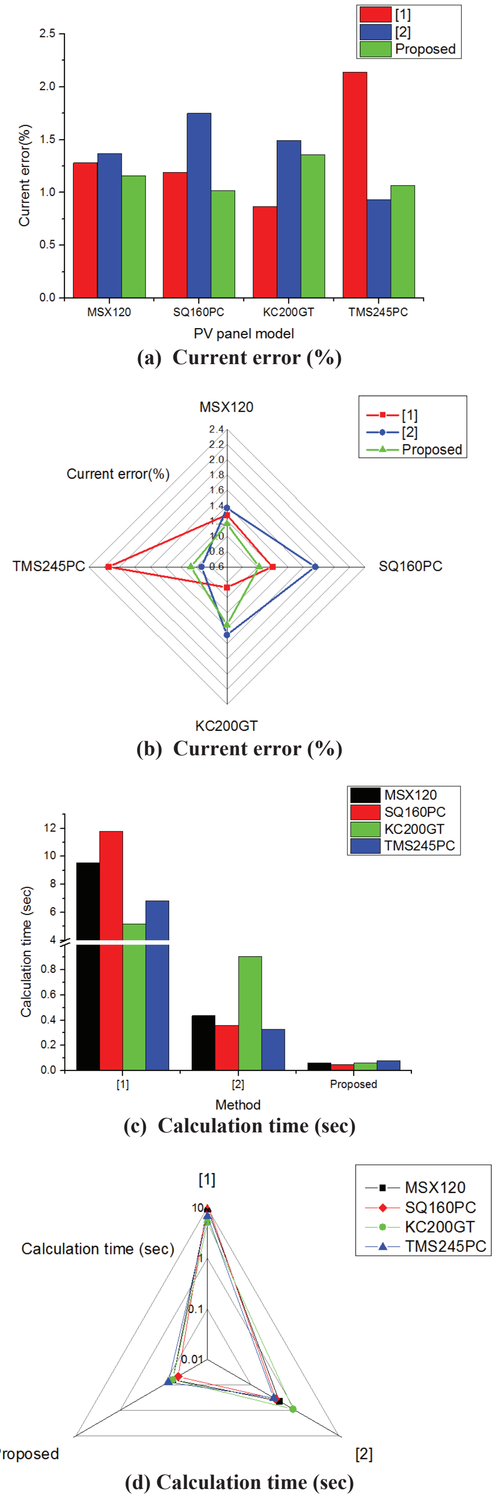
[1] D. Sera, R. Teodorescu, and P. Rodriguez, “PV Panel Model Based on Datasheet Values,” IEEE International Symposium on Industrial Electronics, pp. 2392-2396, 2007.

[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 ed., WILEY, 2009.

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



(a) Current error (%)

(b) Current error (%)

(c) Calculation time (sec)

(d) Calculation time (sec)

Fig. 5. Performance Comparisons

TABLE II. CALCULATION SPEED

	MSX120		SQ160PC		KC200GT		TMS245PC	
	Step	Time(sec.)	Step	Time(sec.)	Step	Time(sec.)	Step	Time(sec.)
[1]	67	9.547	83	11.78	33	5.148	44	6.817
[2]	10	0.437	10	0.359	10	0.905	10	0.328
Proposed	3	0.062	10	0.047	11	0.062	4	0.078