Proceedings of the Fifth IASTED International Conference
POWER AND ENERGY SYSTEMS
June 15-17, 2005, Benalmádena, Spain

DISTRIBUTED-COMPUTING TECHNIQUE FOR MODELLING THE EQUIVALENT CIRCUIT OF PHOTOVOLTAIC MODULES

Leonardo Sandrolini
Department of Electrical Engineering
University of Bologna
Viale Risorgimento 2
I-40136 Bologna, Italy
email: leonardo.sandrolini@unibo.it

Ugo Reggiani and Marcello Artioli
Department of Electrical Engineering
University of Bologna
Viale Risorgimento 2
I-40136 Bologna, Italy
email: ugo.reggiani@mail.ing.unibo.it, marcello.artioli@mail.ing.unibo.it

ABSTRACT
This paper presents a distributed-computing technique based on a particle swarm optimization (PSO) algorithm for the parameter extraction of photovoltaic modules. The procedure exploits interlacing of the experimental data in order to decrease the computational load, and parallel computing of different and independent runs to collect a large amount of data in order to save computational time. The statistical instability typical of numerical data fitting algorithms can thus be efficiently controlled.

KEY WORDS
Renewable energy sources, photovoltaic module, equivalent circuit, distributed computing, particle swarm optimization.

1 Introduction

Among the renewable energy sources the interest in the photovoltaic one is rapidly increasing. The accurate modelling of photovoltaic (PV) modules has therefore become of primary concern in order to integrate these energy sources in more complex systems. Moreover, the insight provided by these models may be used as a quality control during the development stage to improve the devices [1]. The simulation of the behaviour of PV modules can be conveniently done by means of equivalent circuit models, that were originally developed for single cells only, and adapted to modules composed of several cells afterwards. The models make use of nonlinear lumped circuit elements whose parameters are determined from experimental data by means of iterative analytical [2]-[4] or numerical [1],[5]-[7] extraction techniques. The former techniques are attractive for their rapidity, as they require the knowledge of a few selected values only (e.g., short-circuit current, open-circuit voltage, current and voltage at the maximum power point, and slopes of the current-voltage characteristic at the axis intercepts). However, they are based on simplified formulae, and may be affected by the inaccuracy of measurements or calculations. The latter are based on algorithms that fit the calculated current-voltage characteristic to the experimental curve. An evident advantage is that a higher level of confidence of the extracted parameter may be obtained, as several points in the curve are used. As a consequent drawback, these techniques indeed require extensive computation and their accuracy depends on the fitting algorithm, the fitting criterion, i.e., the objective function to minimize, and the starting values of the parameters to extract. These are particularly important as an inappropriate choice of initial values, often obtained by analytical techniques, may result in a nonconvergence of the algorithm or lead to not always realistic parameters [1],[7],[8]. This paper is addressed to investigate the use of distributed (e.g., parallel and interlaced) computing with a fitting strategy based on a particle swarm optimization (PSO) algorithm as an effective tool to attack the parameter extraction problem. Rather than fixing initial values, the PV module parameters are searched within convenient search ranges and found analysing the possible solutions statistically. This results in a heavy computational load which can however be successfully subdivided into a number of computers to save computational time without any accuracy loss. With this technique, as many parameter extractions as the number of computers can be carried out in the same time, allowing one to control the statistical instability typical of numerical data fitting algorithms.

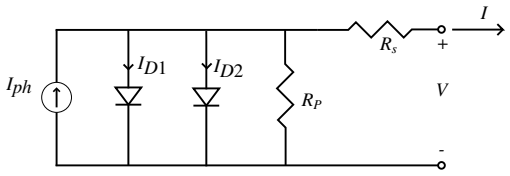

Figure 1. Two-diode equivalent circuit model for a photovoltaic module.

2 Problem definition

2.1 Equivalent circuit

The equivalent circuit considered in this paper to represent the physics of the PV module is the well-known two-diode model depicted in Fig. 1. With respect to the single-diode

model, this one gives a more accurate description of the PV module at low-irradiance conditions [2]. One of the two diodes represents the diffusion current in the p-n junction, whereas the other is added to take the space-charge recombination effect into account. With reference to Fig. 1, the current versus voltage characteristic $I-V$ is given by the nonlinear implicit expression

$$\begin{aligned} I - I_{ph} + I_{s1}\left[\exp\left(\frac{V+IR_S}{n_1 V_T}\right)-1\right] \\ +I_{s2}\left[\exp\left(\frac{V+IR_S}{n_2 V_T}\right)-1\right]+\frac{V+IR_S}{R_P} &= 0, \quad (1) \end{aligned}$$

where

I_{ph} is the photocurrent,

I_{s1}, I_{s2} are the diode saturation currents,

n_1, n_2 are the diode ideality factors,

R_S is the series resistance,

R_P is the parallel or shunt resistance,

V_T is the thermal voltage.

Equation (1) is fitted to the experimental data through an iterative fitting procedure, which is described in the next section.

2.2 Fitting procedure

The procedure implemented to fit the seven unknown PV module parameters $(I_{ph}, I_{s1}, I_{s2}, n_1, n_2, R_S, R_P)$ to the experimental data relies on the particle swarm optimization (PSO) strategy [9],[10]. This optimization algorithm can efficiently find global minima for nonlinear problems. The PSO algorithm belongs to the so-called population based search methods (e.g., genetic algorithms, evolutionary programming, evolution strategies, genetic programming, etc.), that can solve a wide variety of optimization problems. The PSO concept is quite simple, and the few parameters to adjust make a general version of the algorithm applicable to several problems with slight changes only. The swarm strategy simulates the social behaviour of flocking organisms and places its effectiveness in information sharing and successful identification of a leader. Each particle in the swarm is a potential solution of an optimization problem and it moves with an adaptable velocity that depends on its own flying experience (i.e., cognitive term) and on the experience of the other particles in the swarm (i.e., social term). In practice, the former accelerates the particle towards its best position, whereas the latter towards the overall best position. The initialization of the swarm and the velocities is usually performed randomly in the search space, and the particles (i.e., the sets of seven unknown parameters) vary their position until either a relatively unchanging state is found, or a prefixed number of iterations is reached.

In the proposed procedure, the search space for each PV module parameter is chosen conveniently only to limit the number of initial iterations. The parameters are then initialized randomly within their search spaces and therefore are not conditioned by a set of initial values. The adopted fitting criterion consists in minimising the area between the experimental curve and that predicted with the extracted parameters. An additional parameter, independent of the distribution of points, is considered in order to describe the fitting quality [11]. This parameter is the normalized area difference between the predicted and experimental curves. The main (computational) costs of this optimization procedure are proportional to the number of population particles, the (maximum) number of iterations needed to reach the prefixed convergence and the number of points that (1) has to fit to. An additional cost due to the stochastic nature of the PSO may be given by possible repetitions of the optimization procedure in order to obtain the convergence or the desired fitting quality. One of the aim of this paper is to suggest a strategy to lower the computational costs with no (or with a minimum) loss of fitting quality. The first suggestion is to decrease the number of points to fit without accuracy loss through an interlaced computing technique, while the second is to speed up the repetitions through a parallel computing technique.

3 Interlaced and parallel computing overview

3.1 Interlaced computing

The interlacing mechanism is used in TV broadcasting signals, where the image (formed by scan lines) is sent as the rapid sequence of two pictures, one formed with odd lines and the other with even lines, so that the eye sees the full image whereas only one half is actually sent. Such mechanism has already been successfully used in [12] to dramatically improve speed and robustness in a transient detection algorithm for electrical signals. The same principle may also be applied to any number of interlacing subsets. Fig. 2 shows a PV $I-V$ characteristic built with the original complete set of experimental data (the points to fit). There are four different symbol types because the curve is decomposed into four interlacing curves as depicted in Fig. 3, where each interlacing curve has its own symbol marker. In Figs. 2 and 3, only a few points are plotted to make the graphs more readable. Each interlacing curve has a quarter of the total point number of the original curve, and is not coincident with the others and the original one, although very similar to them. The advantage of applying the fitting procedure to four interlacing curves is that one obtains substantially the same fitting quality as if the original curve was used. Fig. 4 shows the fitting quality obtained by different runs of the proposed procedure grouped by an interlacing factor, that is, the *i*th group contains *i* runs, each carried out on one of the *i* interlacing curves. The extracted

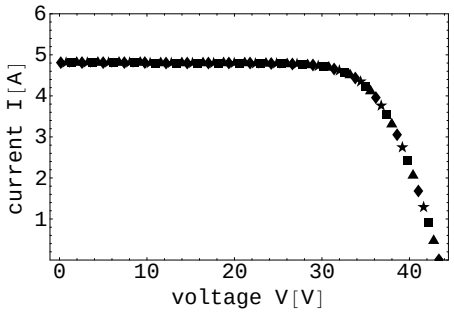

Figure 2. Original data set of the complete experimental *I-V* curve of the PV module.

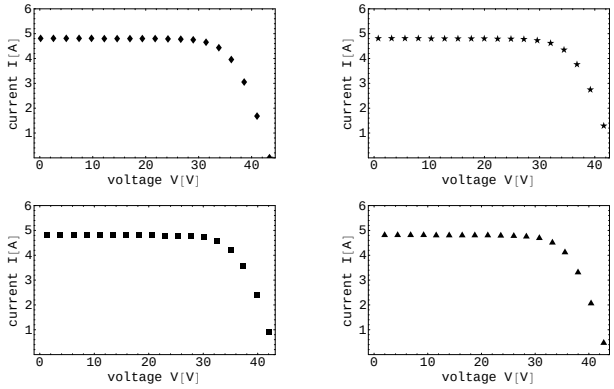

Figure 3. Four data subsets of the complete experimental *I-V* curve of the PV module.

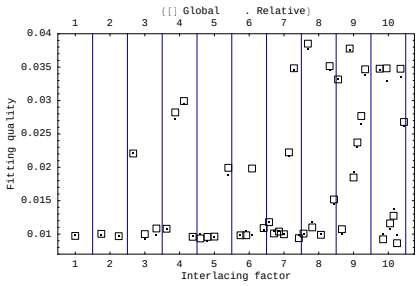

Figure 4. Global and relative fitting quality for different interlacing factors.

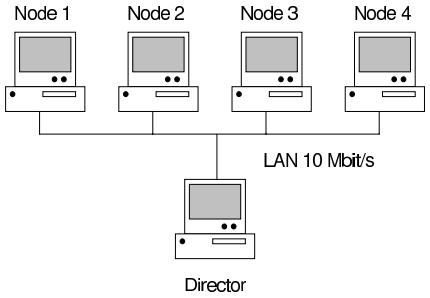

Figure 5. Cluster of computers used to implement the parallel computing strategy.

parameters are used to evaluate the fitting quality with respect to both the interlacing curve they are extracted from (called "relative" and indicated with a dot) and the original curve (called "global" and indicated with a square). The figure shows that even when the fit qualities are not coincident, they are in a good agreement. Of course, there is a trade off between the interlacing factor and the fitting quality in having a sufficient number of points to fit. In fact, for any number of points of the original curve, the higher the interlacing factor, the less the points in each subset/curve. In conclusion, the possibility of carrying out independent runs of the fitting procedure on interlacing curves leads naturally to the second suggestion: parallel computing.

3.2 Parallel computing

Recent advances in high-speed networks and in powerful microprocessors, availability of highly performing clustering software implementations enable cost-effective high-performance parallel computing on clustered low-cost workstations and PCs. Such clusters are very attractive because they rely on available on-the-shelf hardware and software. Many computing architectures have emerged as possible standards and the "Message Passing Interface" (MPI) model is considered one of the most mature method currently used [13]. MPI relies on task co-operation through explicit message exchanges carried out as point-to-point communication between two processes, which normally reside on two different processors, connected through a network. Calculation times have to take communication times into account as well. Generally, in designing parallel algorithms, the problem is first partitioned into many small tasks, then the interdependencies among the tasks and their data are analyzed to improve the communication between the tasks. Finally, the tasks are assigned to processors, following a particular load balancing and task scheduling strategy. Therefore, parallel computing is much more effective when a problem may be split into smaller ones which can be solved independently or with a low rate of dependency to reduce communication and waiting costs (when a processor has completed its task and waits for the output coming from another processor which is not ready yet).

4 Parameter extraction procedure

The stochastic nature of the optimization algorithm makes almost impossible to obtain repeated values for all the PV

module parameters. In fact, several sets of parameters that fit (1) to the experimental curve with different fitting qualities can be found. The optimal solution is obtained analysing statistically a large number of results, that are successfully collected and processed with the distributed-computing technique proposed. Several strategies may be adopted; after repeating the extraction procedure a number of times, the results are examined, and when the frequency distribution of a parameter can be approximated with a normal distribution, its search range is shrunk to the 95% confidence level of its mean. Confidence intervals can also be set for the means of parameters whose frequency distributions are unknown, provided that the skewness of these distributions is small and a sufficient number of samples is collected [14]. The large amount of data to collect and process results in a heavy computational load, that can however be efficiently subdivided into the computers of a cluster. In this paper, the cluster configuration used is really a very low-cost one: five old and dismissed Pentium133 PCs with 32 MByte of RAM and a 10 Mbit/s network card are connected as in Fig 5. One PC acts as director, whose primary job is to send data and instructions to the other computers of the cluster (nodes) and then to collect the results back from them. The MPI used is simple but very easy to setup and configure, as neither any special operating system nor any kind of specialized program library (a standard symbolic programming code is used) are required: only few lines of code are able to manage the program and data flow. The environment is not properly user friendly but it is robust and stable. Of course, the slow network connection and the not sophisticated software make the whole system not suitable for the so called "supercomputing", but allow one to re-use old hardware or to combine the current one in a very simple manner to increase computational power. This is particularly advantageous when the problem flow can be split into different ones with few intersections only, so that the bottle-neck of such a slow communication is avoided or reduced.

The experimental data set is that of a monocrystalline PV module (consisting of 72 silicon solar cells connected in series for a maximum output power of 150 W) for an irradiance level of 1000 W/m^2 and cell temperature of 25 °C. The strategy adopted consists in subdividing the experimental data set into four interlacing subsets, that are passed to different nodes and processed independently. Each node extracts the seven PV module parameters that fit the assigned subset of experimental data and evaluates the fitting quality according to [11]. The advantage of this parallel and interlacing computing technique resides in saving computational time indeed. In fact, each node processes a subset of the experimental data set only and four different sets of results can be obtained at the time cost of one run only.

5 Results

Tab. 1 shows the results obtained after a first run of 300 simulations, of which only 129 have a fitting quality less than 0.01. With the exception of the PV current I_{ph} only, the other parameters are highly variable. For these reasons, the criterion of choosing the mean value as representative of each parameter, although appropriate, is difficult to apply without further investigations. It can be noticed that the frequency distributions of the parameters n_1 and R_s can be approximated by a normal distribution; as an example, the histogram of R_s is depicted in Fig. 6, where the parameter spread range is split into equal-sized bins (classes) and the parameter occurrences that fall into each bin are counted. The skewness of the other five parameter distributions is small, therefore 95% confidence intervals can be assumed for all the parameter mean values. A possible way to proceed is to set these intervals as new search ranges and repeat the extraction procedure. After a second run of 336 simulations, a fitting quality less than 0.01 is obtained in all cases and all the parameters have distributions more similar to a normal distribution (see Figs. 7-13). Moreover, the parameter variation coefficients are now much smaller (see Tab. 2). The results of the second run allow one to choose for the equivalent circuit of Fig. 1 the mean value of each parameter. All the parameters but n_2 show feasible values; the second diode ideality factor seems indeed large also in comparison with n_1. Although the normal approach is to assume $n_2 = 2$ for one ideal monocrystalline solar cell, no clear indication has emerged for PV modules.

Table 1. Mean μ, standard deviation σ and variation coefficient σ/μ of the extracted circuit parameters after the first run (129 simulations with a fitting quality less than 0.01).

Parameter	μ	σ	σ/μ
I_{ph} (A)	4.852	0.032	0.66
I_{s1} (A)	5.54×10^{-7}	2.77×10^{-7}	50.03
I_{s2} (A)	5.17×10^{-4}	2.99×10^{-4}	57.88
n_1	104.39	6.62	6.35
n_2	386.76	107.72	27.85
R_S (Ω)	0.654	0.104	15.83
R_P (Ω)	8141.34	3924.03	48.20

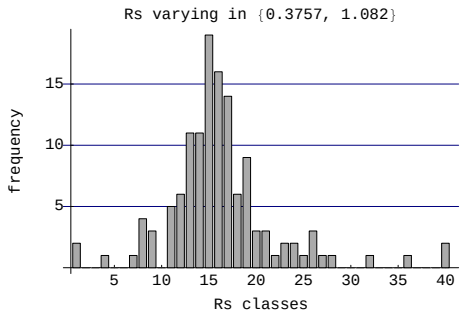

Figure 6. Frequency distribution of R_s after the first run (129 simulations with a fitting quality less than 0.01).

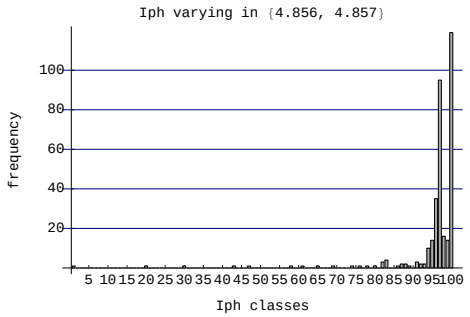

Figure 7. Frequency distribution of I_{ph} after the second run (336 simulations).

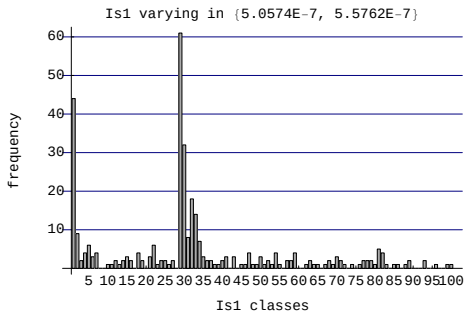

Figure 8. Frequency distribution of I_{s1} after the second run (336 simulations).

Further investigations are therefore required on this point. It must also be noted that the fitting procedure can be carried out imposing higher fitting qualities to the parameters; the value of 0.01 was only chosen for easiness. Better fitting qualities can help in locating the optimal solution; in fact, it was observed that the better the fitting quality, the less the standard deviation of the parameters.

6 Conclusions

In this paper a procedure for the equivalent circuit parameter extraction of photovoltaic modules by means of a distributed-computing technique based on a PSO algorithm is presented. With this technique the computational time can be saved without loosing accuracy by subdividing the calculation tasks into a number of computers. Furthermore, the statistical instability typical of numerical data fitting algorithms can be successfully handled with this approach. A modification in the procedure in order to take the dependence of some parameter on temperature and irradiance into account will be developed in future.

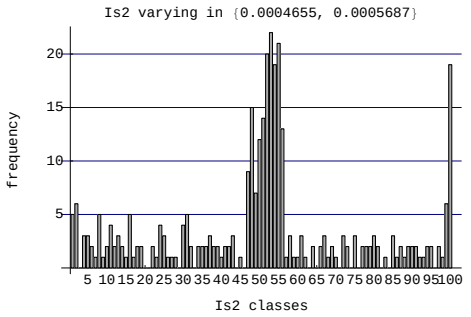

Figure 9. Frequency distribution of I_{s2} after the second run (336 simulations).

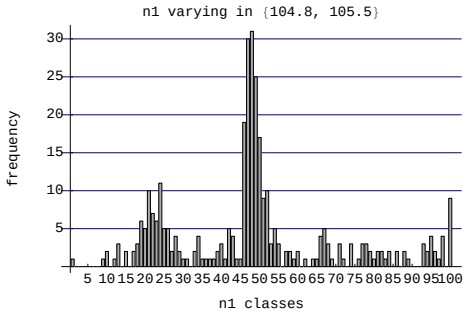

Figure 10. Frequency distribution of n_1 after the second run (336 simulations).

Acknowledgement

This work was supported in part by the Italian University Ministry (MIUR) under a Program for the Development of Research of National Interest (PRIN grant no. 2002098425_002).

References

[1] R. Gottschalg, M. Rommel, D. G. Infield, and M. J. Kearney, "The influence of the measurement environment on the accuracy of the extraction of the physical parameters of solar cells," *Measurement Science & Technology*, vol. 10(9), pp. 796–804, 1999.

[2] D. S. H. Chan and J. C. H. Phang, "Analytical methods for the extraction of solar-cell single- and double-diode model parameters from I-V characteristics," *IEEE Transactions on Electron Devices*, vol. ED-34(2) pt. 1, pp. 286–293, 1987.

[3] J. P. Charles, G. Bordure, A. Khoury, and P. Mialhe, "Consistency of the double exponential model with the physical mechanisms of conduction for a solar cell under illumination," *J. Phys. D-Appl. Phys.*, vol. 18, pp. 2261–2268, 1985.

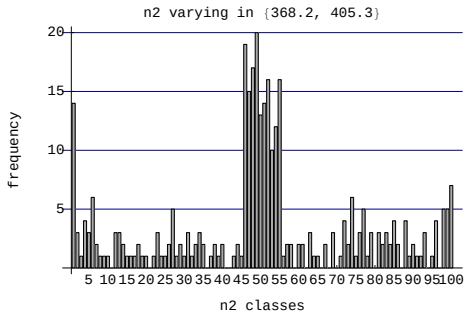

Figure 11. Frequency distribution of n_2 after the second run (336 simulations).

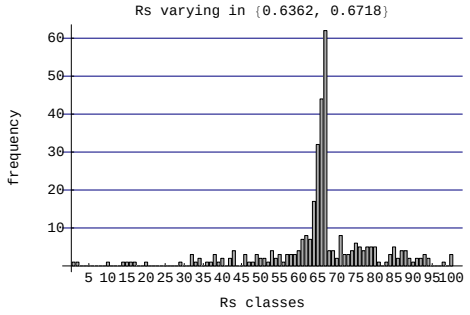

Figure 12. Frequency distribution of R_s after the second run (336 simulations).

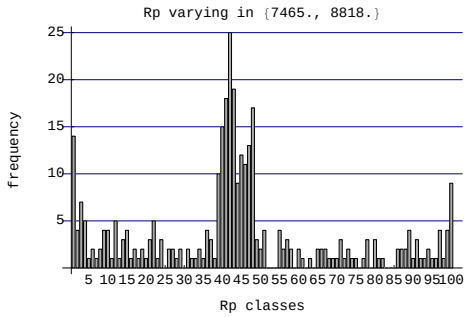

Figure 13. Frequency distribution of R_p after the second run (336 simulations).

Table 2. Mean μ, standard deviation σ and variation coefficient σ/μ of the extracted circuit parameters after the second run (336 simulations with a fitting quality less than 0.01).

Parameter	μ	σ	σ/μ
I_{ph} (A)	4.857	8.16×10^{-5}	1.7×10^{-5}
I_{s1} (A)	5.22×10^{-7}	1.22×10^{-8}	0.023
I_{s2} (A)	5.18×10^{-4}	2.65×10^{-5}	0.051
n_1	105.17	0.15	0.001
n_2	386.70	9.54	0.025
R_S (Ω)	0.659	0.005	0.008
R_P (Ω)	8051.7	348.7	0.043

[4] M. A. De Blas, J. L. Torres, E. Prieto, and A. Garcia, "Selecting a suitable model for characterizing photovoltaic devices," *Renewable Energy*, vol. 25(3), pp. 371–380, 2002.

[5] F. Nakanishi, T. Ikegami, K. Ebihara, S. Kuriyama, and Y. Shiota, "Modeling and operation of a 10 kW photovoltaic power generator using equivalent electric circuit method," in *Proc. 28th Photovoltaic Specialists Conference*, pp. 1703–1706, Sept. 15-22 2000.

[6] M. Chegaar, Z. Ouennoughi, and F. Guechi, "Extracting dc parameters of solar cells under illumination," *Vacuum*, vol. 75, pp. 367–372, 2004.

[7] J. A. Jervase, H. Bourdoucen, and A. Al-Lawati, "Solar cell parameter extraction using genetic algorithms," *Measurement Science & Technology*, vol. 12(11), pp. 1922–1925, 2001.

[8] K. Araki and M. Yamaguchi, "Novel equivalent circuit model and statistical analysis in parameters identification," *Solar Energy Materials and Solar Cells*, vol. 75, pp. 457–466, Feb 1 2003.

[9] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *IEEE Int. Conf. on Neural Networks*, (Perth, Australia), pp. 1942–1948, 27 Nov. - 1 Dec. 1995.

[10] R. C. Eberhart and Y. Shi, "Particle swarm optimization: Developments, applications and resources," in *2001 Congress on Evolutionary Computation*, (Seoul, Korea), pp. 81–86, May 27-30 2001.

[11] D. S. H. Chan, J. R. Phillips, and J. C. H. Phang, "A comparative study of extraction methods for solar cell model parameters," *Solid State Electronics*, vol. 29(3), pp. 329–337, 1986.

[12] M. Artioli, G. Pasini, L. Peretto, R. Sasdelli, and F. Filippetti, "Low-cost dsp-based equipment for the real-time detection of transients in power systems," *IEEE Transactions on Instrumentation and Measurement*, vol. 53(4), pp. 933–939, 2004.

[13] G. Fadlallah, M. Lavoie, and L. A. Dessaint, "Parallel computing environments and methods," in *Proc. Parallel Computing in Electrical Engineering - PARELEC 2000*, (Trois-Rivieres, Canada), pp. 2–7, 27-30 Aug. 2000.

[14] E. Kreyszig, *Advanced Engineering Mathematics*. New York, USA: John Wiley & Sons, Inc., 2nd ed., 1968.