IEICE Proceeding Series

Exploring Maximum Power Point by Population-Based Optimization Algorithms

Masaya MURAOKA, Noriaki MIKAMI, Toshimichi SAITO

Vol. 1 pp. 618-621
Publication Date: 2014/03/17
Online ISSN: 2188-5079

2012 International Symposium on Nonlinear Theory and its Applications
NOLTA2012, Palma, Majorca, Spain, October 22-26, 2012

Exploring Maximum Power Point by Population-Based Optimization Algorithms

Masaya MURAOKA†, Noriaki MIKAMI† and Toshimichi SAITO†

†Graduate School of Engineering, Hosei University, Koganei, Tokyo, 184–8584 Japan
Email: masaya.muraoka.9y@stu.hosei.ac.jp, noriaki.mikami.9i@stu.hosei.ac.jp, tsaito@hosei.ac.jp

Abstract—This paper studies an application of the differential evolution algorithm to the maximum power point search in photovoltaic systems. Depending on the insolation, the power characteristic varies and complicated multi-peak may be generated. Applying a basic algorithm to a multi-peak objective function of terminal voltage vector, we have confirmed that the particles tend not to be trapped into local optima and global search can be possible. If parameters are selected suitably, our algorithm can find the maximum power points.

1. Introduction

The differential evolution (DE, [1] [2]) is a population-based optimization algorithm. In the DE algorithm, particles correspond to potential solutions. Following a difference equation, the particle location is updated. The DE is simple in concept and is easy to implement. The DE does not require differentiability of the objective functions and is applicable to various problems such as multi-peak problems. Engineering applications are many, including optimal design of signal processing circuits [3] [4].

In this paper, we consider an application of DE to photovoltaic (PV) systems. In order to increase the power efficiency, it is important to find the maximum power point (MPP) of PV arrays. It can contribute to design an efficient maximum power point tracker (MPPT, [5]-[7]).

Depending on the insolation, the power characteristic of the PV system varies and complicated multi-peak may be generated. In such a situation, local maximum power points (LPPs) may be generated and the LPPs make the MPPT difficult. In this paper, we construct a multi-peak objective function of two dimensional terminal voltage vectors (one-dimensional case is studied in [7]). Applying a rand type DE to this problem, we have confirmed that particles are scattered, tend not to be trapped into LPPs and global search can be possible. If parameters are selected suitably, the DE can find the MPP of multi-peak functions.

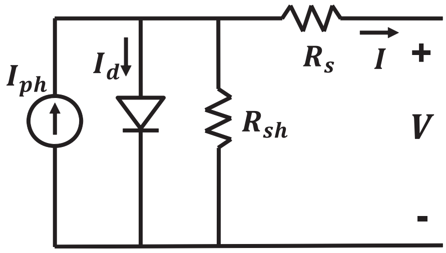

Figure 1: Equivalent circuit of a solar cell.

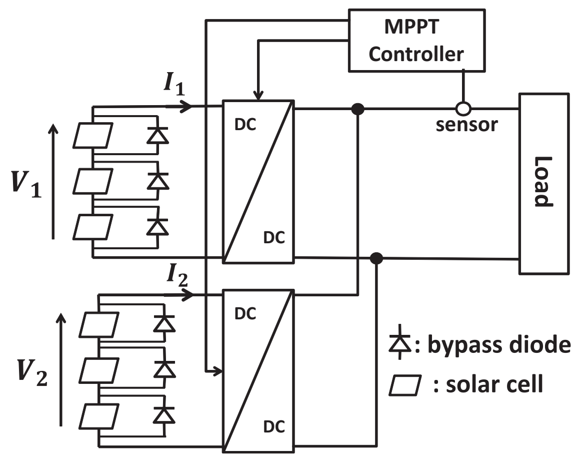

Figure 2: The objective system.

2. Objective Function

In this section, we introduce the objective function based on the equivalent circuit of a solar cell as shown in Fig. 1 [8] [9]. The voltage versus current characteristics is described by

$$I = I_{ph} - I_{rs}[\exp(\frac{qV}{kTA}) - 1] \tag{1}$$

where I is terminal current and V is terminal voltage. I_{ph} is photo-generated current, I_{rs} is the cell reverse saturation current, q is elementary charge, k is Boltzman constant, T is absolute temperature and A is diode ideality factor. For simplicity, series resistance R_s and shunt resistance R_{sh} are ignored. As the objective system, we consider multiple solar arrays system as shown in Fig. 2. It includes two sets of three solar cells connected in series. Each set is controlled by single centralized MPPT controller. The V_i versus I_i , i= 1 , 2 , characteristics is shown in Fig. 3. The objective function is the power generated by these solar arrays and depends on V_1 and V_2:

$$P = F(V_1, V_2)$$

Figure 4 shows contour map of this function where the maximum power point (MPP) is given at $V_1 = 1.23[V]$, $V_2 = 1.26[V]$, power=7.27 [W].

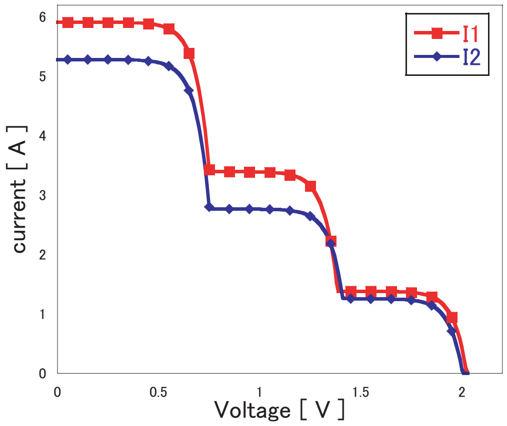

Figure 3: I-V characteristics of the objective function.

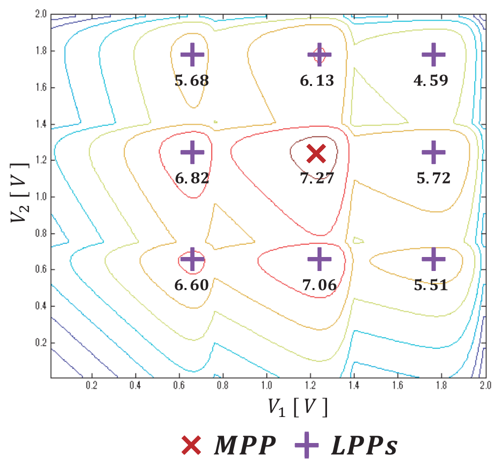

Figure 4: Contour map of the objective function.

3. Differential Evolution

In order to define the DE algorithm, we prepare N particles. Each particle is characterized by its position on the V_1 versus V_2 plane. The position $x_i \equiv (V_{1i}, V_{2i})$ is a potential solution. Since it is hard to set plural operating points, in the practical system, we use successive N operating points in time domain as the plural points in space domain virtually. Let X^t be the single particle (corresponding to single operating point) at time t and let x_i^n be the virtual i-th particle at iteration step n. Let $X^1 \equiv x_1^1$, $X^2 \equiv x_2^1, \cdots$, and let $X^N \equiv x_N^1$. In general, let the single particle at time $Nt+i$ be equivalent to the i-th virtual particle at iteration step n:

$$X^{Nt+i} \equiv x_i^n, \quad i \sim N$$

For convenience, we use the virtual particle x_i^n in the definition of the algorithm. The algorithm is defined by the following 5 steps.

Step 1 (Initialization): Let iteration step $n = 0$. The particles $\boldsymbol{x}_i^n$, $i = 1 \sim N$, are initialized randomly in the search space S_0. Let $X \equiv \{x_1^n, \cdots, x_N^n\}$.

Step 2 (Mutation): Three vectors $\boldsymbol{x}_{x1}^n$, $\boldsymbol{x}_{x2}^n$ and $\boldsymbol{x}_{x3}^n$ selected randomly from the set of particles X where $\boldsymbol{x}_{x1}^n \neq \boldsymbol{x}_{x2}^n \neq \boldsymbol{x}_{x3}^n$. A candidate vector $\boldsymbol{y}_i^n$ is made by

$$\boldsymbol{y}_i^n = \boldsymbol{x}_{x1}^n + B(\boldsymbol{x}_{x2}^n - \boldsymbol{x}_{x3}^n) \tag{2}$$

where B is the scaling parameter.

Step 3 (Crossover): Applying the crossover with probability P_c to the candidate vector $\boldsymbol{y}_i^n$ and the parent $\boldsymbol{x}_i^n$, we obtain an offspring $\boldsymbol{c}_i^n$ where $i = 1 \sim N$.

Step 4 (Survival): The parent $\boldsymbol{x}_i^n$ is compared with the offspring $\boldsymbol{c}_i^n$ and is updated as the following:

$$\begin{array}{l} \text{if } \ (f(\boldsymbol{x}_i^n) < f(\boldsymbol{x}_i^{new})) \ \text{ then } \boldsymbol{x}_i^n = \boldsymbol{x}_i^{new} \\ \text{if } \ (f(\boldsymbol{x}_i^n) > f(\boldsymbol{x}_i^{new})) \ \text{ then } \boldsymbol{x}_i^n = \boldsymbol{x}_i^n \end{array} \tag{3}$$

Step 5 (Termination Condition): Let $n = n + 1$, go to Step 2 and repeat until the maximum time limit n_{max}.

4. Numerical Experiment

In order to confirm the algorithm efficiency, we have performed basic numerical experiments for objective function as shown in Fig. 4. We have fixed the parameters: scaling parameter $B = 0.7$, crossover probability $P_c = 0.5$, number of particles $N \in \{5, 10, 20, 30, 50\}$ and maximum time limit $n_{max} = 100$. If the Gbest (the best value of the all particles) reach exceeds 7.25 [W] (99.7 % of MPP), we declare that the exploring is successful.

Figures 5 and 6 show exploring process in successful and unsuccessful runs, respectively. In the DE particles can be scattered and global search is possible. The particles tend not to be trapped into LPPs. Figure 7 shows Gbest for the number of particles N= 5, 10, 30 and 50.The result are summarized in Table. 1: the success rate SR and number of iterations #ITE are given in average after 1000 trials. As N increases inter-particle communication frequency increases and the search speed becomes faster. In Fig. 8, we have confirmed power sequence of single particle X^t corresponding to plural particles in Fig. 5. The power sequence is inconstant and the efficiency cannot be high.

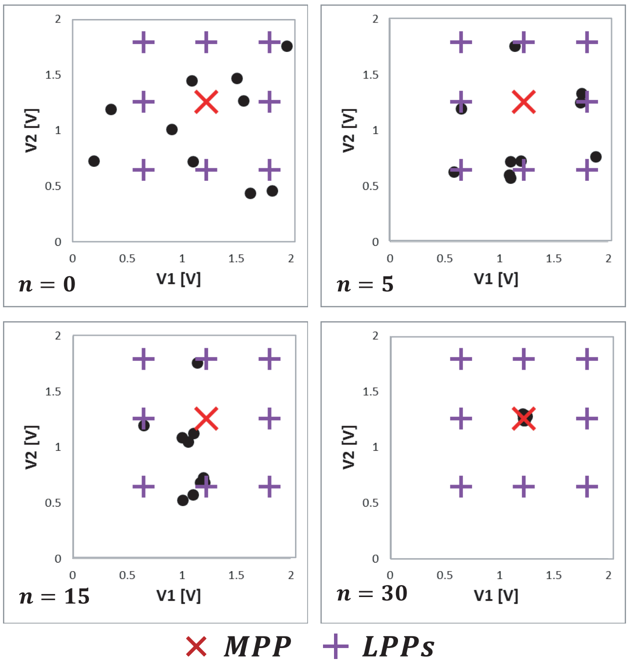

Figure 5: Exploring process in successful run.

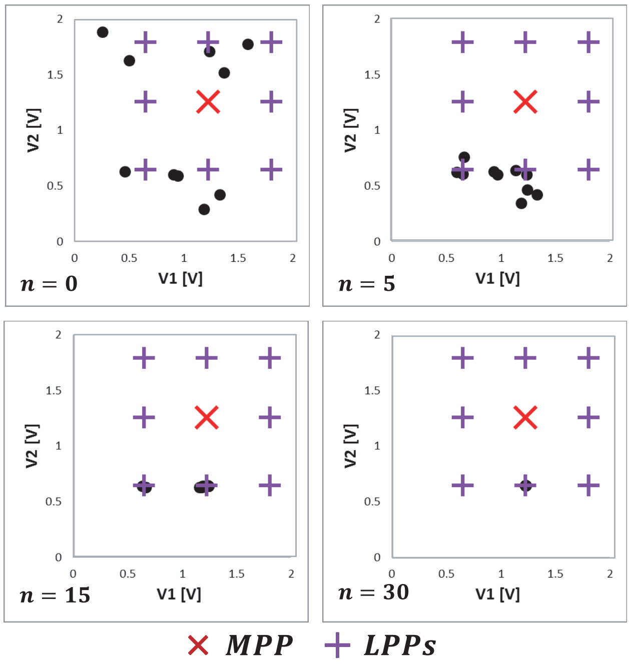

Figure 6: Exploring process in unsuccessful run.

Table 1: Success rate and iteration number.

#PCL	SR [%]	#ITE
5	82.0	16.4
10	99.3	13.3
20	100	10.8
30	100	9.41
50	100	7.84

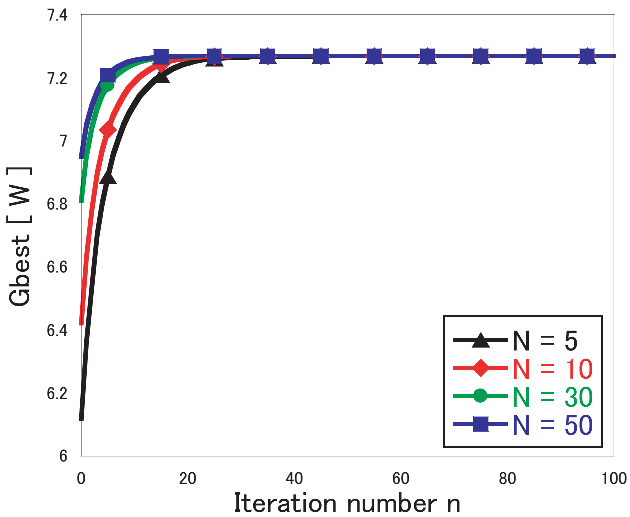

Figure 7: Characteristics of Gbest.

5. Conclusions

An application of DE to MPP search is studied in this paper. Performing basic numerical experiment, we have confirmed that the particles can be scattered. They tend not to be trapped into LPPs and global search can be possible. If the number of particle is suitable, we can find the MPP successfully. Future problems include consideration of an efficient MPPT method and comparison with other population-based optimization algorithms.

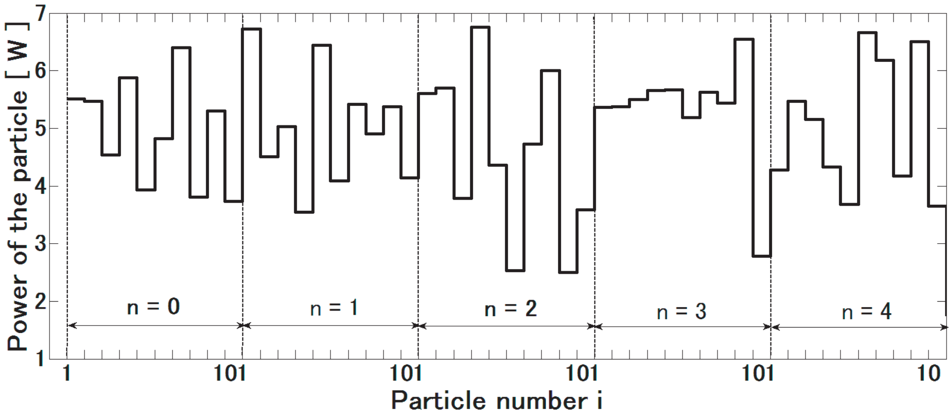

Figure 8: Power sequence of single particle X^t.

References

[1] R. Storn and K. Price, "Minimizing the real functions of the ICEC' 96 contest by differential evolution", in Proc , pp. 842-844, 1996.

[2] R. Storn and K. Price, "Differrential evolution - a simple and efficient heuristic for global optimization over continuous spaces", Journal of Global Optimaization, Vol. 11, pp. 341-359, 1997.

[3] Lampinen, H. and Vainio, O., "An optimization approach to designing OTAs for low-voltage sigma-delta modulators", Proc. of the 2001 WCCI, pp. 1665-1671, 2001.

[4] Luitel, B. and Venayagamoorthy, G.K., "Differential evolution particle swarm optimization for digital filter design", Proc. of the 2008 IEEE, pp. 3954-396, 2008.

[5] Masafumi Miyatake, Mummadi Veerachary, Fuhito Toriumi, Nobuhiko Fujii and Hideyoshi Ko, "Maximum Power Point Tracking of Multiple Photovoltaic Arrays: A Particle Swarm Optimization Approach" IEEE Transactions on Aerospace and Electronic Systems, Vol. 47, No. 1, pp. 367-380, 2011.

[6] Tat Luat Nguyen and Kay-Soon Low, "A Global Maximum Power Point Tracking Scheme Employing DIRECT Search Algorithm for Photovoltaic Systems", IEEE Trans. Industrial, Vol. 57, No.10, pp. 3456-3467, 2010.

[7] Hamed Taheri, Zainal Salam, Kashif Ishaque and Syafaruddin, "A Novel Maximum Power Point Tracking Control of Photovoltaic System Under Partial and Rapidly Fluctuating Shadow Conditions Using Differential Evolution", IEEE Symposium on Industrial Electronics and Applications, pp. 82-87, 2010.

[8] G. Vachtsevanos and K. Kalaitzakis, "A Hybrid Photovoltaic Simulator for Utility Interactive Studies", IEEE Transactions on Energy Conversion, Vol. EC-2, No. 2, pp. 227-231, 1987.

[9] Hengsi Qin and Jonathan W. Kimball, "Parameter Determination of Photovoltaic Cells from Field Testing Data using Particle Swarm Optimization", IEEE Power and Energy Conference at Illinois, pp. 1-4, 2011.