Photovoltaic System Allocation Using Discrete Particle Swarm Optimization with Multi-level Quantization

Hwachang Song†, Ryan Diolata* and Young Hoon Joo*

Abstract – This paper presents a methodology for photovoltaic (PV) system allocation in distribution systems using a discrete particle swarm optimization (DPSO). The PV allocation problem is in the category of mixed integer nonlinear programming and its formulation may include multi-valued discrete variables. Thus, the PSO requires a scheme to deal with multi-valued discrete variables. This paper introduces a novel multi-level quantization scheme using a sigmoid function for discrete particle swarm optimization. The technique is employed to a standard PSO architecture; the same velocity update equation as in continuous versions of PSO is used but the particle's positions are updated in an alternative manner. The set of multi-level quantization is defined as integer multiples of powers-of-two terms to efficiently approximate the sigmoid function in transforming a particle's position into discrete values. A comparison with a genetic algorithm (GA) is performed to verify the quality of the solutions obtained.

Keywords: Multi-level quantization, Optimal allocation, Discrete particle swarm optimization, Photovoltaic systems

1. Introduction

Recent advances in photovoltaic (PV) technologies have brought very promising opportunities for the utilization of renewable solar energy systems. In [1], IEA reported that in 2007 alone about 2.26 GW of PV capacity were installed (an increase of more than 50% over the previous year) which brought the total installed capacity to 7.8 GW. Of the total capacity installed in 2007, about 95% (2.16GW) were installed as grid connected systems. The expected insertion and increasing penetration of grid connected PV systems will modify the way in which the entire system is planned and operated. This has raised many challenges for utility grid planners and operators.

The installation of PV systems at non-optimal places have several uncertainties and this may cause operational problems such as the violation of voltage limits and increase in power losses, resulting in an increased operational cost. It is therefore essential for the utility operators to investigate the technical and economic impacts of installing PV systems in their grid. In [2], Paatero et al. investigated the effect of large scale implementation of distributed PV generation in distribution networks. They modeled three different network topologies and studied the effect on voltage and power losses at different penetration levels. Hernandez et al. in [3] presented a systematic approach for optimal location and sizing of PV systems in distribution feeders. They followed a multi-objective optimization approach which considers both the technical and economical aspects.

The optimal placement and sizing of PV systems in distribution networks is a complex optimization problem. It involves both discrete and continuous variables. Owing to the discrete and discontinuous nature of the problem, classical techniques are rendered unsuitable and the use of a global search technique is warranted. Recently, metaheuristics optimization methods have been widely applied, especially in hard optimization problems involving continuous and discrete variables.

In past years, particle swarm optimization (PSO) has been successfully applied in many research and engineering areas. It is demonstrated that PSO provides better results in a faster, cheaper way compared with other global optimization methods such as a genetic algorithm (GA) [4-5]. The PSO was originally developed for nonlinear continuous optimization problems [6]; in the real world, however, many optimization problems are defined in discrete value spaces where the domain of the solution space is finite. It can be argued that with any problems, discrete or continuous, variables can be transformed into its equivalent binary representation [7].

In [8], the fathers of the PSO algorithm, Kennedy and Eberhart, introduced the discrete binary PSO (DBPSO) when they reworked the original version in order to operate on discrete binary variables. There has been some other exploration of PSO techniques for discrete optimization. Yang et al. [9] and Khanesar et al. [10] developed an algorithm based on DBPSO which uses a different method to update velocity. Pampara et al. [11] solved the binary optimization problem using angle modulation with only four parameters in continuous PSO, which allowed for faster optimization of several problems. Inspired by natural evolution, Sadri and Suen [12] extended the DBPSO by using birth and death operations to model a dynamic swarm. Lee et al. [13] modified the original DBPSO by adopting the

† Corresponding Author : Dept. of Electrical Engineering, Seoul National University of Technology, Korea. (hcsong@snut.ac.kr)
* School of Electronic and Information Engineering, Kunsan National University, Korea. ({ryan_diolata, yhjoo}@kunsan.ac.kr)
Received 11 February 2009; Accepted 8 May 2009

concepts of genotype-phenotype representation and the mutation operator of genetic algorithms. Rastegar et al. [14] introduced another discrete binary PSO based on learning automata. Multi-valued discrete particle swarm has also been proposed in [7, 16] and tested on benchmark problems. The algorithms reported above are limited only to binary problems and are suitable for variables that can take only three states 1, 0 and -1. Moreover, several concerns over DBPSO are as follows: the range of the discrete variable often does not match the upper limit of the binary equivalent representation, the hamming distance between two discrete values undergoes a nonlinear transformation when an equivalent binary version is used, and the binary representation increases the dimensions of the particle [7]. Parameters and the memory of DBPSO are some other concerns as discussed by Khanesar et al. [10].

In this paper, a methodology for the optimal allocation of photovoltaic systems in distribution systems is presented, using a discrete PSO algorithm. The discrete PSO is modified from a standard PSO with linearly decreasing inertia weight (PSO-LDIW) by the inclusion of a novel multi-level quantization scheme. In order not to compromise the robustness of the PSO algorithm, the same velocity update equation that preserves the social and cognitive components is used but the particle's position is updated in an alternative way. The variables are not converted into equivalent binary representations. Instead, in this paper, a multi-level quantization scheme is adopted which mainly includes the approximation of a sigmoid function to transform a particle's position into discrete values. A comparison with a genetic algorithm (GA) is performed to verify the quality of the solutions obtained by the discrete PSO-LDIW. The versatility of this technique was shown by applying into other continuous PSO versions, such as repulsive particle swarm optimization (RPSO), and a PSO with a constriction factor approach (PSO-CFA).

2. PV Allocation Problem Formulation

Assuming that the candidate buses are given for the installation of PV systems and that the maximum capacity of PV systems to be installed at each bus is provided, represented as ($\Delta P \cdot u_{M1}$, $\Delta P \cdot u_{M2}$, ..., $\Delta P \cdot u_{Mc}$), where c denotes the total number of candidate buses, and ΔP is the capacity of a PV system considered and u_{Mm} is the maximum integer variable corresponding to the maximum capacity at bus m. In this problem, it is important to determine the appropriate number of units and the best locations in the given network.

To evaluate the operational improvement of the PV installation, active power losses can be taken as the performance index for the PV allocation planning problem. The losses in the network branches can be calculated as the difference of the injected power between the sending end and the receiving end buses. The system loss then can be described as follows:

$$f_1 = \sum_{m=1}^{b} P_{loss}^{m} = \sum_{m=1}^{b} \left(P_{s,m} - P_{r,m} \right) \tag{1}$$

where $P_{s,m}$ and $P_{r,m}$ denotes the injected active power from the sending and receiving end, respectively, and b stands for the total number of branches. The objective function is minimizing f_1.

The constraints considered in this paper are as follows:

- Equality constraints for nonlinear power flow equations: The power balance equation is included as an equality constraint to ensure the balance between supply and demand. These constraints for bus m are presented as follows:

$$0 = P_{Gm} + \Delta P_{pv} u_m - P_{Dm} - V_m \sum_{j=N} V_j \cdot \left(G_{mj} \cos\theta_{mj} + B_{mj} \sin\theta_{mj} \right) \tag{2}$$

$$0 = Q_{Gm} - Q_{Dm} - V_m \sum_{j=N} V_j \cdot \left(G_{mj} \cos\theta_{mj} - B_{mj} \sin\theta_{mj} \right) \tag{3}$$

- Inequality constraint on voltage limits: Voltage levels at the distribution buses should be within the established limits to maintain power quality. This constraint is described as follows:

$$V_m^{\min} \le V_m \le V_m^{\max} \tag{4}$$

- Inequality constraint associated by the PV systems penetration level: The maximum PV penetration level should be smaller or equal to the specified amount. The maximum PV penetration level can be set as a percentage of the total load. In this paper the maximum PV penetration level is set to 80% of the total real power load of the network. This constraint is now described as follows:

$$\begin{aligned} P_{gen}^{pv} &\le P_{spec}^{pv} \\ P_{gen}^{pv} &= \sum_{m \in S_{pv}} \Delta P_{pv} u_m \end{aligned} \tag{5}$$

where the following notations are made: P_{Gm} is real power generated at bus m; P_{Dm} is real power load at bus m; Q_{Gm} is reactive power generated at bus m; Q_{Dm} is the reactive power load at bus m; G_{mj} and B_{mj} are the conductance and susceptance for the (m, j) component from Y bus matrix; θ_{mj} is the voltage angle difference between bus m and j; V_m and V_j are voltage magnitudes of bus m and j, respectively; $V_m^{\max}$ and $V_m^{\min}$ are the upper and lower limits for bus m, respectively; P_{gen}^{pv} and P_{spec}^{pv} are the total injected power and the specified penetration level of PV units, respectively; $\Delta P_{pv} u_m$ is the power injected of PV at bus m with a unit number u_m.

Equations (2) and (3) are the equality constraints for the active and reactive power balance at each bus. In equation (2), u_m is an integer variable for the number of PV units that are installed at bus m, and $u_m = 0, \ldots, u_{Mm}$. Voltage deviation at each bus is restricted in the upper and lower limit and described as an inequality constraint in (4). Another constraint included is the inequality constraint described in (5) that limits the total penetration of PV power by the specified P_{spec}^{pv}. To force these constraints within the limits, these two inequality constraints are converted into the objective function as quadratic penalty terms for the application of simulation based optimization techniques such as a PSO. Then, the augmented objective function can now be described as:

$$\min F = f_1 + \alpha_1 f_2 + \alpha_2 f_3$$

$$f_2 \equiv \begin{cases} 0 \ , \ P_{gen}^{pv} \le P_{spec}^{pv} \\ 1 \ , \ P_{gen}^{pv} > P_{spec}^{pv} \end{cases} \tag{6}$$

$$f_3 \equiv \sum_{m \in uv} \left(V_m^{\max} - V_m\right)^2 + \sum_{m \in lv} \left(V_m^{\min} - V_m\right)^2$$

where α_1 and α_2 are the penalty factors against the solution's violating the two inequality constraints; uv and lv are the sets of buses whose voltage magnitudes are violating the upper and lower voltage limits, respectively. If the set of buses exceeds beyond statutory voltage limits, in this paper, they are penalized by a very high value of $\alpha_2 (10^6)$. The parameter α_1 (10^6) is the value introduced if the PV penetration level exceeds beyond the maximum penetration value. For this purpose, the injection violation index, f_2 is employed. Normally, f_2 is set to 0 but it is set to 1 if $P_{gen}^{pv} > P_{spec}^{pv}$. This optimization problem is severely nonlinear due to the equality constraints of (2) and (3) and the original objective function f_1, which is said to be have many local minima. Besides, the main variables of the problem are integer as the penetration level of the PV system at a candidate bus m is expressed by $\Delta P \cdot u_{Mm}$.

3. Discrete Particle Swarm Optimization Algorithms

3.1 Continuous PSO versions

A problem is given in the PSO architecture and a way to evaluate a proposed solution to it exists in the form of a fitness function. A communication structure or social network is also defined, assigning neighbors for each individual to interact with. Then a population of individuals defined as random guesses at the problem solutions are initialized. These individuals are candidate solutions, also known as particles. As a single particle by itself is unable to accomplish anything due to the fact power is an interactive collaboration, an iterative process to improve these candidate solutions needs to be set in motion. The particles iteratively evaluate the fitness of the candidate solutions and remember the location where they had their best success. The individual's best position corresponding to their best solution is called the particle best or local best. Each particle makes this information available to their neighbors. Each particle has a memory and remembers the following information: the particle's best position, "*pbest*", where the particle itself attained its best success, and the global best position, "*gbest*", where its neighborhood or any particle in the swarm attained the best success globally.

In every iteration, each particle moves from the current position to the next by adjusting its own position and velocity based on these two best positions. The particle position and velocity update equations in the simplest form that govern the PSO is given by equations (7) and (8) below:

$$V_i^{k+1} = w \cdot V_i^k + C_1 \cdot r_1 (p_{pbest}^k - X_i^k) + C_2 \cdot r_2 \cdot (p_{gbest} - X_i^k) \tag{7}$$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \tag{8}$$

$$w = w_2 + (w_2 - w_1) \cdot \left(\frac{(t_{\max} - t)}{(t_{\max})} \right) \tag{9}$$

where X_i^{k+1} represents the current position of the particle, X_i^k is the previous particle's position, V_i^{k+1} is the current velocity of the particle, V_i^k is the previous velocity of the particle. C_1 and C_2 are the acceleration coefficients while r_1, and r_2 are random numbers uniformly distributed in the interval [0, 1]. P_{pbest}^k is the personal best position of the particle, p_{gbest} is the global best position of the swarm. w_1 and w_2 stands for the initial and final value of inertia weight respectively. t_{max} is the maximum number of iterations and t as the current iteration number.

The role of inertia weight w is considered to be very important in PSO convergence behavior [18]. The inertia weight is employed to control the momentum of the particle from the previous history of velocity on the current velocity. The larger inertia weight facilitates global exploration, while a smaller inertia weight tends to facilitate local exploration. To ensure the search balance and to accelerate convergence, a time-varying inertia weight, w, is utilized and varies from 0.9 at the beginning to 0.4 toward the end of optimization [18].

The early versions of PSO [6,17-19] were meant to handle nonlinear continuous optimization problems. Many numerical experiments, for example [4-5], proves that a PSO can obtain high accuracy global optimal or quasi-optimal solutions for continuous variable multimodal functions and is superior to other meta-heuristic techniques like simulated annealing (SA) and genetic algorithms (GA). It is said that a PSO can be easily expanded to treat problems with discrete variables [15, 20].

3.2 Previous discrete PSO versions

The discrete binary PSO (DBPSO) version was first reported by Kennedy and Eberhart in [8]. The main concept of the discrete binary PSO version is the same as the continuous one. The discrete binary PSO uses the same velocity update equation as in the continuous version, but the values for the solution, X, (particle's position) are transformed into discrete binary value. In the discrete binary PSO, the velocity functions as a probability that a bit (position) takes on zero or one. The transformation of the velocity in the interval [0, 1] is accomplished using the sigmoid function defined as:

$$sig(V_i^{k+1}) = \frac{1}{1 + \exp\left(-V_i^{k+1}\right)} \tag{10}$$

The position is then updated and defined by the following rule [24];

$$X_i^{k+1} = \begin{cases} 0 \ \ if \ ran() \ge sig(V_i^{k+1}) \\ 1 \ \ if \ ran() < sig(V_i^{k+1}) \end{cases} \tag{11}$$

where $sig(V_i^{k+1})$ represents the sigmoid limiting transformation and *ran( )* is a random number generated with a uniform distribution in the interval [0, 1].

It is noted that the inertia weight parameter in discrete binary PSO has a value of 1 and hence this parameter has no use for DBPSO convergence. The velocity term is also limited to $|V_i^{k+1}| \leq V_{\max}$, where $V_{\max}$ is a value typically close to 6.0 to correspond to a maximum probability of 0.9975 that a bit is flipped into 1, and a minimum probability of 0.0025 that the bit remains 0.

There has been some other exploration of PSO techniques for discrete optimization. Authors El-Dib et al. [9] had applied the PSO technique for the VAR planning problem in power systems. In their scheme, sigmoid function was also used for logical transformation sig(V_i^{k+1}) to accomplish the particle's motion. The change in position is defined by the following rule:

$$V_i^{k+1} = \begin{cases} 1 \ if \ sig(V_i^{k+1}) > \rho \\ 0 \ if -\rho \leq sig(V_i^{k+1}) \leq \rho \\ -1 \ if \ sig(V_i^{k+1}) < -\rho \end{cases} \quad (12)$$

where ρ is the user defined threshold value.

Using the concept of phasor, Moradi and Firuzabad [15] proposed a trinary version of discrete PSO for switch placement in distribution systems. In this algorithm, the phase of V_i is calculated and then it is mapped into one of the three states. The three states are represented by unity vectors with the angle of 0°, -120° and 120°. The angles of the vectors are transformed into interval [0, 1] such as 1/6, 3/6, and 5/6. At each step, the phase difference of V_i and the three numbers are determined. The number, d_k, is transfor- med using:

$$T(d_k) = \frac{\alpha \cdot \exp(\tan(\pi(0.5 - d_k)))}{1 + \alpha \cdot \exp\exp(\tan(pi(0.5 - d_k)))} \quad (13)$$

where d_k is one of the distances (d_1, d_2, d_3), and α is a constant number for tuning up the convergence rate.

Using this transformation, when the distance approaches zero, the transformed number is limited toward 1 or the probability that the state to be selected is about 1 and when the distance is about 0.34 the probability is 0. In fact, the transformation equation in (13) is a modified sigmoid transformation. This algorithm is more likely to be of the one proposed in [5] and which is suitable only for three states.

The algorithms mentioned above are suitable only for variables that can take three states 1, 0 and -1 and limited only to discrete problems with binary valued solution elements. In the real world, many problems are multi-valued discrete problems. This algorithm may not be suitable for problem types which include multi-valued discrete variables. If these algorithms are applied to the problem, the movement of particles may only take a one step forward or one step backward scheme. The limited movement (one step forward - one step backward) scheme of particles may slow down the search process or may even bring the algorithm trap to a local minimum. The proposed multi-value discrete transformation for the PSO is discussed in the next section.

4. Multi-valued Quantization Scheme for a Discrete PSO

4.1 Motivations

The discrete binary PSO has been able to optimize various discrete-valued optimization problems. However, there are several drawbacks associated with this algorithm: the range of the discrete variable often does not match the limit of the equivalent binary representation; the hamming distance between two discrete values undergoes a nonlinear transformation when an equivalent binary representation is used, which often adds complexity to the search process; and the binary representation increases the dimension of the particle [7]. Changes to the concept of particle trajectory and velocity in the discrete binary PSO introduce another concern. The parameter inertia weight is insignificant and in some cases has difficulty tuning because the values of $w<1$ prevents the convergence of the DBPSO and for values $-1<w<1$, V_i^{k+1} becomes 0 over time [10].

Another parameter V_{max} offers a different concept for the DBPSO. In the continuous PSO version, a large value for V_{max} encourages particle exploration, while in the DBPSO a small value for V_{max} promotes particle exploration even if a good solution is found. If we examine equation (11) it can be observed that the next value for the bit is dependent on the current value of that bit and the value is updated using only the velocity vector. Thus, particle memory is not good as it was in the continuous PSO version. To resolve some of the problems associated with a DBPSO, we have defined the multilevel quantization that approximates the sigmoid function to transform a particle's position into multilevel discrete values. The new algorithm can optimize multi-valued discrete optimization problems.

4.2 Multi-level quantization

Assuming the discrete variables in a multi-valued discrete problem has range [0, N], where N implies the number of N-array number system. To transform the position of particles into a multi-value discrete number, first, we map the values of velocity (V_i^{k+1}), with range [-V_{max}, V_{max}], to a hypercube with range [-1, 1] using the logistic function equation in (14). Then, the set of array of the multilevel discrete values is defined by the set of the quantization level. The set of multilevel quantization is defined as integer multiples of power of two that can be expressed as the sums and differences of power-of-two terms [21].

$$S_i = sig(V_i^{k+1}) = -1 + \frac{2}{1+\exp\left(-V_i^{k+1}/\varsigma\right)} \tag{14}$$

where ς is the steepness of the sigmoid function.

The important issue is how to set the ranges to represent the coefficients of l level of discrete value numbers. First, we will identify the range of values that the particle position will not move $\left(X_i^{k+1}(S_i)=0\right)$ for the next step. To set for this range, the power-of-two spaces' is used. If the value of S_i is within the range of [-2^l, 2^l], the particle takes no movement on the next iteration. For the succeeding steps, the coefficients (n) of discrete values at l-th level are defined by the ranges expressed as the sums and differences of powers-of-two terms. The multilevel approximation of the sigmoid function is shown in Fig. 1.

The steepness function ς can be adjusted to vary the values of ranges of S_i, and in this study ς is set to 1. The new position update of the particle is defined by the following rule:

$$X_i^{k+1}(S_i)=\begin{cases} X_i^k+n, & -1+2^{-n}<S_i \\ \vdots & \\ X_i^k+l, & -1+2^{-l}<S_i\le 1-2^{-(l+1)} \\ \vdots & \\ X_i^k & -2^{-1}\le S_i\le 2^{-1} \\ \vdots & \\ X_i^k-l, & -1+2^{-(l+1)}-<S_i\le -1+2^{-l} \\ \vdots & \\ X_i^k-n, & S_i<-1+2^{-n} \end{cases} \tag{15}$$

where n is the discrete maximum value of velocity for the i-th variable, X_i, of a particle.

4.3 Particle velocity

In the DBPSO, the velocity term indicates the probabilities of solution elements that assume the value of 0 or 1 and the parameter inertia weight has no function or contribution for the algorithm's convergence. In our algorithm, the significance of the velocity variable and its parameters, the meaning of velocity clamping and the inertia weight, follows exactly the standard continuous PSO version. In fact, inertia weight has some valuable information about the previously found directions.

As in the continuous version, a large value of velocity (V_i^{k+1}) results in a random search, where the particle will take a large step size in just a single iteration. For example, if the value of V_i^{k+1} is large the particle will encourage more exploration while in the DBPSO a smaller value may promote more exploration even if a good solution is found. Also, a large amount of maximum velocity, V_{max}, encourages more exploration and a lesser value causes the particle to move less. When a particle is approaching a region of good solution, this means that a particle's velocity is approaching zero and the particle's step in discrete value is also less.

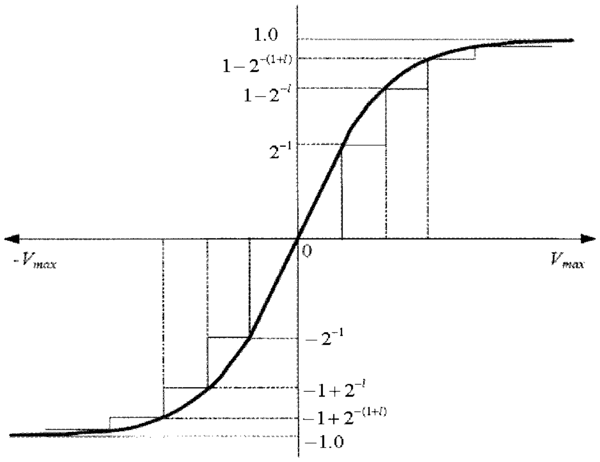


Fig. 1. Multi-level quantization with a sigmoid function

Pertaining to the position updating rule of the DBPSO, the movement of a particle at the current step is independently performed from that at the previous step. When the particle's velocity approaches 0, then, the search at the next iteration tends to be changed into a pure random search. This might cause a loss of information on the previously found good positions. In our algorithm, the position update uses the previous position information, and the social and cognitive component is preserved. Furthermore, the inertia term is used to maintain the previous direction of the particle to the personal best or global best positions.

5. Simulation Results and Discussions

5.1 Numerical setup

To verify the feasibility of the proposed technique, the PSO-LDIW combined with the proposed multilevel quantization is applied to the PV allocation optimization problem. The PV allocation problem is formulated as an integer nonlinear optimization problem. An IEEE-37 bus distribution system was utilized as the test system. The topology and complete data of the test network can be found in [19]. The system's total real and reactive power loads are 814 kW and 406 kVAr, respectively, with real and reactive power generation of 838.179 kW and 422.421 kVAr respectively at the slack bus. The system's total real power loss before PV integration is 0.125881 pu. In this simulation, two cases employing different problem dimensions where simulated. The cases are described as follows:

- Case 1: This case is a strategic allocation of PV systems. The candidate buses are selected based on the initial evaluation of power flow from which buses with the lowest voltage magnitudes are chosen as candidates for installation. The goal of this case is not just to minimize the losses but also to provide voltage support.

- Case 2: In this case all buses except bus 1 are chosen as candidates for installation. The aim of this case is to relieve the loads in distribution lines and to intensively reduce line losses.

In both cases, the position (bus location) and sizes of the PV units are chosen as decision variables. Each PV unit is assumed to constantly deliver 10 kW of peak power of which the power factor is assumed to be unity. In both cases the maximum penetration level is set as 80% of the total real power load. The algorithm's parameter settings are shown in Table 1. In this study, four levels of quantization are applied and hence the maximum discrete value, n, in (15) is 3. The proposed multilevel quantization is also applied to the RPSO [23-24] and PSO-CFA [19] to show that the proposed technique can be applied to any continuous PSO algorithm.

Table 1. PSOs and GA parameters

	C_1	C_2	C_3	K	W_{max}	W_{max}	P_{mut}	P_{cross}
LDIW	2.0	2.0	-	-	0.9	0.9	-	-
CFA	2.8	1.3	-	0.729	-	-	-	-
RPSO	1.5	-1.5	0.5	-	0.7	0.01	-	-
GA	-	-	-	-	-	-	0.01	0.8

To show the quality of the solutions obtained by the PSO algorithms, a comparison using a GA is made. GAs have been reported to produce a good solution for discrete variable problems. In this paper, the GA uses the following scheme: real coding process, proportional selection, elitist model, two-point crossover, uniform mutation with 0.01 probability, and a crossover probability of 80%. A GA with two-point crossover can better handle problems with long blocks of parent vector swapped in their entirety into the child [8].

5.2 Case 1

Case 1 is a problem with a lower dimension wherein only eight candidate buses and 10 different sizes (discrete values) of PV units available as choices for each bus. The performance of the heuristic algorithm must be concluded only after many trials. To verify efficiency, 50 independent runs are simulated for each algorithm. For each separate run, the particles are initialized in random positions. A maximum number of iteration ($iter_{max}$ = 100) is used as the stopping criterion for each algorithm.

The standard deviation, average, worst and best values of the final solutions of different algorithms for case 1 are shown in Table 2. The PSO-LDIW combined with the proposed multilevel quantization technique has obtained higher quality solutions compared to the GA and the two other PSO algorithms. For all categories, including standard deviation, averages and worst and best values, the PSO-LDIW employed in conjunction with multilevel quantization achieves the lowest values. Note that the variation on the objective function, especially on power loss, is minimal for the following reasons: there are only a few selected candidate buses for installation, and the distances between each of the candidate buses are relatively small.

Table 2. Performances of the four algorithms for Case 1

	Objective function value			
	Average	Worst	Best	Std. dev.
GA	0.0513785	0.0519725	0.0511500	0.000169
PSO-LDIW	**0.0511366**	**0.0511993**	**0.0511099**	**0.000025**
PSO-CFA	0.0511673	0.0514618	0.0511107	0.000064
RPSO	0.0512091	0.0514349	0.0511147	0.000073

To show the convergence process of the four algorithms considered, another set up is made. For all considered algorithms, particles are randomly initialized in the same distributions. Fig. 2 illustrates the comparison of the performances according to iterations of the algorithms considered.

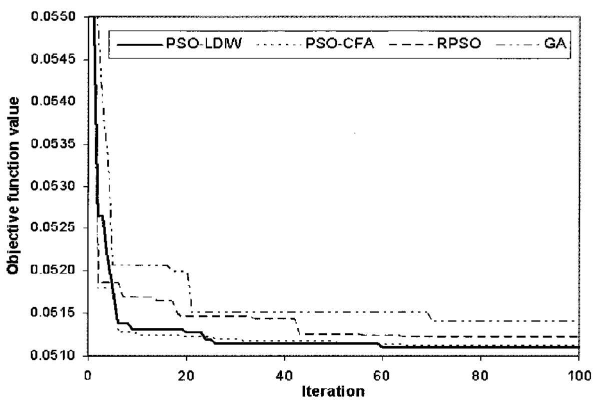


Fig. 2. Comparison of the four algorithms for Case 1

The PSO-LDIW and PSO-CFA have converged in almost the same optimal solution. It appears that the discrete PSO algorithms with the proposed multilevel quantization converge faster compared with the GA and can locate a good quality solution in fewer iterations. Moreover, the PSO-LDIW is far superior compared to other algorithms considered.

5.3 Case 2

To investigate the effect of dimension on the searching quality of a PSO-LDIW combined with multilevel quantization, the number of candidate buses is increased using the same distribution test system. In this case, all load buses are chosen as candidates for installation and the penetration level is set as a constant. By choosing all load buses as candidates for installation, the power loss will vary considerably. This will result in the creation of many multiple minima in the region and is very difficult to optimize due to the restriction in voltages and the degree of PV penetration.

The standard deviation, average, worst and best values of the final solutions of different algorithms for case 2 are shown in Table 3. The PSO-LDIW employed with multilevel quantization achieves the lowest values for all categories (standard deviation, average, worst and best values) even in high dimensional problem. We can see from the result that GA and RPSO are finding difficulties in finding a good solution. In fact, there are many trial runs that the

GA and RPSO algorithms failed to find a solution.

Table 3. Performances of the four algorithms for Case 2

	Objective function value			
	Average	Worst	Best	Std. dev.
GA	20000.03	1000000	0.0254504	141421.356
PSO-LDIW	**0.02501**	**0.027516**	**0.0239057**	**0.000790**
PSO-CFA	20000.02	1000000	0.0239694	141421.352
RPSO	660000	1000000	0.0256233	478518.1

Fig. 3 shows the illustration of performances according to iterations of the GA and PSO algorithms combined with the proposed technique. All three PSO algorithms combined with the proposed technique converge faster in a lower value of solution compared to the GA. The GA's stairway-like characteristic graph shows the GA falling in local minima, and it takes extensive iterations in order to escape from that local minima. It also shows that the PSO-LDIW and PSO-CFA combined with the proposed multi-level quantization technique easily on escapes local minima and are very robust in finding global or quasi-global solutions. In general, the PSO-CFA has shown a comparable performance with the PSO-LDIW while the GA and RPSO had difficulties falling from local minima.

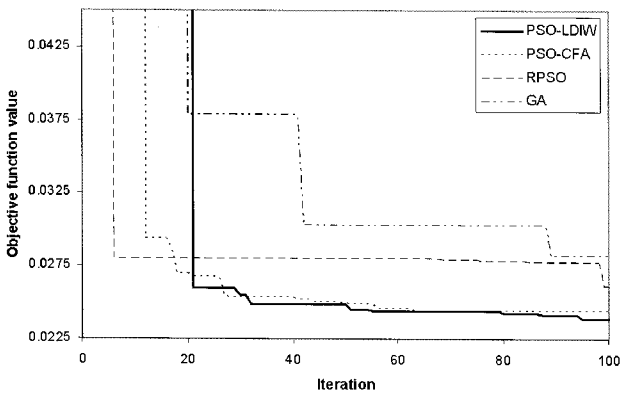


Fig. 3. Comparison of the four algorithms for Case 2

5.4 Discussions

Furthermore, the final solution is recorded for every separate run. If the final solution of each individual run is within the range of 0.1% of the global optimal value, we call it as a successful run. This range is chosen as very low because, for case 1, the deviation of the objective function is small. Since the global optimal value of the problem is unknown, in this case we define the lowest optimal solution amongst the four algorithms as the global optimum. We then evaluate the index named 'success rate (SR)' defined as follows:

$$SR = \frac{N_{sr}}{N_{tr}} \times 100\% \tag{16}$$

where N_{tr} (N_{tr} =50) denotes the total number of runs and N_{sr} denotes the number of successful runs among 50 independent runs.

The result for robustness analysis is shown in Table 4. All three different PSO algorithms combined with the proposed multilevel quantization technique achieved a 100% success rate while the GA has only 90%. This simply shows that PSO algorithms applied with the proposed technique have a very high probability of finding global or quasi-global solutions.

Table 4. Result of Robustness Analysis

	Success rate (%)	
	Case 1	Case 2
GA	90.0	8.0
PSO-LDIW	**100.0**	**96.0**
PSO-CFA	100.0	92.0
RPSO	100.0	4.0

As can be seen in Table 4, if the final solution of each individual run for Case 2 is within the range of 1% of the global optimal value (the lowest optimal solution amongst the four algorithms), we consider it to be a successful run. A PSO-LDIW combined with the proposed multilevel quantization can find global optima with very high probability, even in higher dimensional problems. In fact, in all 50 trial runs the PSO-LDIW combined with the proposed technique found a solution, and only in a few cases did it fail to find a global or quasi-global solution. On the other hand, a PSO-CFA with multi-level quantization has a higher success rate, compared to GA and RPSO, but the average value and standard deviation of objective function values obtained is rather high. The poor performance of the RPSO may be caused by the loss of particle information on the global best position, which is crucial information for population based optimization techniques. The high values on the average and standard deviation are caused in some cases when the algorithm fails to find a solution.

Computational efficiency analysis is also carried out based on CPU computational time. Each individual algorithm was separately run for 50 trials with a fixed number of iterations, and in this case the maximum iteration is set to 100. The average CPU time to complete the fixed number of iterations is shown in Table 5. The PSO-LDIW combined with the proposed multilevel quantization achieved the lowest CPU computational time in two cases. The GA, on the other hand, has the highest CPU computational rate among the algorithms considered. The added computational burden for the GA is due to the fact its operators like mutation, crossover and selection. In general, the PSO-LDIW combined with the proposed multilevel quantization outperformed the GA and the two other PSO versions (RPSO and PSO-CFA) combined with the proposed multilevel quantization.

Table 5. Average CPU computational time

	Average time (ms)	
	Case 1	Case 2
GA	32799.58	32864.90
PSO-LDIW	**32437.78**	**32397.66**
PSO-CFA	32514.86	32751.82
RPSO	32475.52	32647.76

6. Conclusions

This paper presents a new technique for the PV allocation problem using a discrete PSO. The proposed discrete PSO includes a novel quantization scheme to handle multi-valued discrete variables in the mixed integer nonlinear programming problem. The multilevel quantization scheme maps a set of continuous values representing each particle's position into discrete values using integer multiples of powers-of-two terms in the sigmoid function. The new technique was applied to a PSO-LDIW algorithm as well as the two other continuous PSO versions (RPSO and PSO-CFA), and they are applied to the PV system allocation problem. Using the IEEE 37 bus distribution test system, two cases with different problem dimensions were employed and simulated. From these results, one can see that a PSO-LDIW applied with the proposed quantization technique could better handle multi-valued discrete problems as compared with the two other continuous PSO versions and GA. Also, it shows that the PSO-LDIW and PSO-CFA employed with the quantization scheme could have a very high probability of finding a global optimum and provide a better quality solution even in high dimensional problems, compared to a GA.

Acknowledgements

This work was supported by the Korea Ministry of Knowledge Economy (MKE) through the EIRC program at Korea University.

References

[1] *Trends in Photovoltaic Applications: Survey Report of Selected IEA Countries between 1992 and 2007*, IEA Annual Report, Aug. 2008. Available: http://www.iea-pvps.org/

[2] Paatero, J. V., and Lund, P. D., "Effects of large-scale photovoltaic power integration on electricity distribution networks", *Renewable Energy*, vol. 32, no. 2, Feb. 2007, pp. 216-234.

[3] Hernández, J. C., Medina, A., and Jurado, F., "Optimal allocation and sizing for profitability and voltage enhancement of PV systems on feeders", *Renewable Energy*, vol. 32, no. 10, Aug. 2007, pp. 1768-1789.

[4] Ide, A., and Yasuda, K., "A Basic Study of Adaptive Particle Swarm Optimization", *Electrical Engineering in Japan*, vol. 151, no. 3, 7 March 2005, pp. 41-49.

[5] El-Dib, A. A., Youssef, H. K. M., El-Metwally, M. M., and Osman, Z., "Optimum VAR sizing and allocation using particle swarm optimization", *Electric Power Systems Research*, vol. 77, no. 8, June 2007, pp. 965-972

[6] Eberhart, R., and Kennedy, J., "A New Optimizer Using Particle Swarm Theory", *Proceedings of Sixth International Symposium on Micro Machine and Human Science*, Nagoya, Japan, 4-6 October 1995.

[7] Veeramachaneni, K., Osadciw, L., and Kamth, G., "Probabilistically Driven Particle Swarms for Optimization of Multi-valued Discrete Problems: Design and Analysis", *Proceedings of IEEE Swarm Intelligence Symposium*, Honolulu, Hawaii, 1-5 April 2007, pp. 141-149.

[8] Kennedy, J., and Eberhart, R., "A Discrete Binary Version of Particle Swarm Optimization Algorithm", *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, Orlando, Florida, USA, vol. 5, 12-15 Oct. 1997, pp. 4104-4108.

[9] Yang, S., Wang, M., and Jiao, L., "A Quantum Particle Swarm Optimization", *Proceedings on IEEE Congress on Evolutionary Computation*, Portland, Oregon, vol. 1, 19-23 June 2004, pp. 320-324.

[10] Khanesar, M. A., Teshnehlab, M., and Shoorehdeli, M. A., "A Novel Binary Particle Swarm Optimization", *Proceedings of Mediterranean Conference on Control and Automation*", Athens, Greece, 27-29 July 2007.

[11] Pampara, G., Franken, N., and Engelbrecht, A. P., "Combining Particle Swarm Optimisation with angle modulation to solve binary problems", *Proceedings of IEEE Congress on Evolutionary Computing*, Edinburgh, Scotland, vol. 1, 5 Sept. 2005, pp. 89-96.

[12] Sadri, J., and Suen, C. Y., "A Genetic Binary Particle Swarm Optimization Model", *Proceedings of IEEE Congress on Evolutionary Computation*, Vancouver, BC, Canada, 16-21 July 2006, pages 656-663.

[13] Lee, S., Soak, S., Oh, S., Pedrycz, O., and Jeon, M., "Modified Binary Particle Swarm Optimization", *Progress in Natural Science*, vol. 18, no. 9, 10 Sept. 2008, pp. 1161-1166.

[14] Rastegar, R., Meybodi, M. R., and Badie, K., "A New Discrete Binary Particle Swarm Optimization Based on Learning Automata", *Proceedings of IEEE International Conference on Machine Learning and Applications*, Louisville, KY, USA, 16-18 Dec. 2004, pp. 456-462.

[15] Moradi, A., and Fotuhi-Firuzabad, M., "Optimal Switch Placement in Distribution Systems using Trinary Particle Swarm Optimization Algorithm", *IEEE Transactions on Power Delivery*, vol. 23, no. 1, Jan. 2008, pp. 271-279.

[16] Pugh, J., and Martinoli, A., "Discrete Multi-Valued Particle Swarm optimization", *Proceedings of IEEE Swarm Intelligence Symposium*, Indianapolis, Indiana, USA, 12-14 May 2006, pp. 103-110.

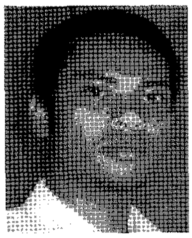
[17] Kennedy, J., and Eberhart, R., "Particle swarm optimization", *IEEE International Conference on Neural Network*, Perth, Australia, 1995.

[18] Shi, Y., and Eberhart, R., "A Modified Particle Swarm Optimizer", *Proceedings of IEEE International Conference on Evolutionary Computation*, Anchorage, AK, 4-9 May 1998.

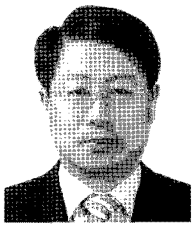
[19] Clerc, M., "The Swarm and the Queen: Towards a Deterministic and Adaptive Particle Swarm Optimization", *IEEE Proceedings on Congress on Evolu-*

tionary Computation, Washington DC, USA, 6-9 July 1999.
[20] Yoshida, H., Kawata, K., Fukuyama, Y., Takayama, S., and Nakanishi, Y., "A Particle Swarm Optimization for Reactive Power and Voltage Control Considering Voltage Security Assessment", *IEEE Transactions on Power Systems*, vol. 15, no. 4, Nov. 2000, pp. 1232-1239.
[21] Ko, J. H., Joo, J. S., and Lee, Y. H., "On the Use of Sigmoid Functions for Multistage Detection in Asynchronous CDMA Systems", *IEEE Transactions on Vehicular Technology*, vol. 48, no. 2, March 1999, pp. 522-526.
[22] Al-Hinai, A., Sedhisigarchi, K., and Feliachi, A., "Stability Enhancement of a Distribution Network Comprising a Fuel Cell and a Microturbine", *Proceedings on IEEE Power Engineering Society General Meeting*, Denver, CO., USA, 6-10 June 2004.
[23] Urfalioğlu, O., "Robust Estimation of Camera Rotation, Translation and Focal Length at High Outlier Rates", *Canadian Conference on Computer and Robot Vision*, Canada, 17-19 May 2004.
[24] Lee, K. H., Baek, S. W., and Kim, K. W., "Inverse Radiation Analysis Using Repulsive Particle Swarm Optimization Algorithm", *International Journal of Heat and Mass Transfer*, vol. 51, no. 11-12, June 2008, pp. 2772-2783.

Hwachang Song received his B.S., M.S. and Ph.D. degrees in Electrical Engineering from Korea University in 1997, 1999 and 2003, respectively. He was a Post-doctoral Visiting Scholar at Iowa State University from 2003 to 2004, and a Post-doctor at Korea University from Sept. 2004 to March 2005. He worked as an Assistant Professor in the School of Electronic and Information Engineering, Kunsan National University, from 2005 to 2008. Currently, he is an Assistant Professor in the Department of Electrical Engineering, Seoul National University of Technology.



Ryan Diolata received his B.S. degree in Electrical Engineering from the Western Institute of Technology, Iloilo, Philippines in 2005. Also, he received his M.S. in Electrical, Electronic and Control Engineering, from Kunsan National University, in 2009.



Young Hoon Joo received his B.S., M.S. and Ph.D. degrees in Electrical Engineering from Yonsei University, Seoul, Korea, in 1982, 1984, and 1995, respectively. He worked with Samsung Electronics Company from 1986 to 1995 as a project manager. He was with the University of Houston, TX, from 1998 to 1999, as a Visiting Professor. He is currently a Professor in the School of Electronic and Information Engineering, Kunsan National University. He is serving as President for the Korean Institute of Intelligent Systems (KIIS) (2007-2008) and Director for the Transactions of the Korean Institute of Electrical Engineers (KIEE) (2009-2010) and Editor for International Journal of Control, Automation and Systems (IJCAS) (2006-9).