Artif Intell Rev (2015) 43:243–258
DOI 10.1007/s10462-012-9373-8

Particle swarm optimisation for discrete optimisation problems: a review

Ahmad Rezaee Jordehi · Jasronita Jasni

Published online: 13 November 2012

Abstract In many optimisation problems, all or some of decision variables are discrete. Solving such problems are more challenging than those problems with pure continuous variables. Among various optimisation techniques, particle swarm optimisation (PSO) has demonstrated more promising performance in tackling discrete optimisation problems. In PSO, basic variants are merely applicable to continuous problems. So, appropriate strategies should be adopted for enabling them to be applicable to discrete problems. This paper analyses all strategies adopted in PSO for tackling discrete problems and discusses thoroughly about pros and cons of each strategy.

Keywords Particle swarm optimisation · Optimisation · Discrete environments

1 Introduction

There are so many optimisation problems in various areas of science and engineering. For solving them, there exist twofold approaches; classical approaches and heuristic approaches. Classical approaches such as linear programming and non-linear programming are not efficient enough in solving optimisation problems. Since they suffer from curse of dimensionality and also require preconditions such as continuity and differentiability of objective function that usually are not met.

Heuristic approaches which are usually bio-inspired include a lot of approaches such as genetic algorithms, evolution strategies, differential evolution and so on. Heuristics do not expose most of the drawbacks of classical and technical approaches. Among heuristics, particle swarm optimisation (PSO) has shown more promising behavior.

PSO is a stochastic, population-based optimisation technique introduced by Kennedy and Eberhart (1995). It belongs to the family of swarm intelligence computational techniques

A. Rezaee Jordehi (✉) · J. Jasni
Department of Electrical Engineering, University Putra Malaysia, 43400 UPM Serdang,
Selangor, Malaysia
e-mail: ahmadrezaeejordehi@gmail.com

and is inspired of social interaction in human beings and animals (especially bird flocking and fish schooling).

Some PSO features that make it so efficient in solving optimisation problems are the followings:

- In comparison with other heuristics, it has less parameters to be tuned by user.
- Its underlying concepts are so simple. Also its coding is so easy.
- It provides fast convergence.
- It requires less computational burden in comparison with most other heuristics.
- It provides high accuracy.
- Roughly, initial solutions do not affect its computational behavior.
- Its behavior is not highly affected by increase in dimensionality.
- There exist many efficient strategies in PSO for mitigating "premature convergence." Thus, its success rate is so high.

However, in many optimisation problems, all or some of decision variables are discrete. Solving such problems are more challenging than those problems with pure continuous variables. Since in PSO, basic variants are merely applicable to continuous problems, appropriate strategies should be adopted for enabling them to be applicable to discrete problems. This paper analyses deeply all strategies which are adopted in PSO for tackling discrete problems and discusses about their pros and cons. The paper is organised as follows; in Sect. 2, an overview of PSO is presented. In Sect. 3, all strategies adopted in PSO for dealing with discrete problems are analysed in details. As an example of wide applicability of discrete PSO, its applications in power system problems are mentioned in Sect. 4. Finally, drawing conclusions and proposing some directions for future research are implemented in Sect. 5.

2 PSO overview

PSO starts with the random initialisation of a population (swarm) of individuals (particles) in the n-dimensional search space (n is the dimension of problem in hand). The particles fly over search space with adjusted velocities. In PSO, each particle keeps two values in its memory; its own best experience, that is, the one with the best fitness value (best fitness value corresponds to least objective value since fitness function is conversely proportional to objective function) whose position and objective value are called P_i and P_{best} respectively and the best experience of the whole swarm, whose position and objective value are called P_g and g_{best} respectively. Let denote the position and velocity of particle i with the following vectors:

$$X_i = (X_{i1}, X_{i2}, \ldots, X_{id}, \ldots, X_{in})$$
$$V_i = (V_{i1}, V_{i2}, \ldots, V_{id}, \ldots, V_{in})$$

The velocities and positions of particles are updated in each time step according to the following equations:

$$V_{id}(t+1) = V_{id}(t) + C_1 r_{1d}(P_{id} - X_{id}) + C_2 r_{2d}\left(P_{gd} - X_{id}\right) \quad (1)$$
$$X_{id}(t+1) = X_{id}(t) + V_{id}(t+1) \quad (2)$$

where C_1 and C_2 are two positive numbers and r_{1d} and r_{2d} are two random numbers with uniform distribution in the interval [0,1]. Here, according to (1), there are three following terms in velocity update equation:

(1) The first term this models the tendency of a particle to remain in the same direction it has traversing and is called “inertia”, “habit” or “momentum”.
(2) The second term is a linear attraction toward the particle’s own best experience scaled by a random weight C_1r_{1d}. This term is called “memory”, “nostalgia” or “self-knowledge”.
(3) The third term is a linear attraction toward the best experience of the all particles in the swarm, scaled by a random weight C_2r_{2d}. This term is called “cooperation”, “shared information” or “social knowledge”.

The procedure for implementation of PSO is as follows:

(1) Particles’ velocities and positions are initialised randomly, the objective value of all particles are calculated, the position and objective of each particle are set as its P_i and P_{best} respectively and also the position and objective of the particle with the best fitness (least objective) is set as P_g and g_{best} respectively.
(2) Particles’ velocities and positions are updated according to Eqs. (1) and (2).
(3) Each particle’s P_{best} and P_i are updated, that is, if the current fitness of the particle is better than its P_{best}, P_{best} and P_i are replaced with current objective value and position vector respectively.
(4) P_g and g_{best} are updated, that is, if the current best fitness of the whole swarm is fitter than g_{best}, g_{best} and P_g are replaced with current best objective and its corresponding position vector respectively.
(5) Steps 2–4 are repeated until stopping criterion (usually a prespecified number of iterations or a quality threshold for objective value) is reached.

It should be mentioned that since the velocity update equations are stochastic, the velocities may become too high, so that the particles become uncontrolled and exceed search space. Therefore, velocities are bounded to a maximum value V_{max}, that is Eberhart et al. (2001)

$$If\ |V_{id}| > V_{max}\ then\ V_{id} = sign\left(V_{id}\right) V_{max} \tag{3}$$

where sign represents sign function.

However, primary PSO characterised by (1) and (2) does not work desirably; especially since it possess no strategy for adjusting the trade-off between explorative and exploitative capabilities of PSO. Therefore, the inertia weight PSO is introduced to remove this drawback. In inertia-weight PSO, which is the most commonly-used PSO variant, the velocities of particles in previous time step is multiplied by a parameter called inertia weight. The corresponding velocity update equations are as follows Shi and Eberhart (1998, 1999):

$$\begin{aligned} V_{id}(t+1) &= \omega V_{id}(t) + C_1r_{1d}\left(P_{id} - X_{id}\right) + C_2r_{2d}\left(P_{gd} - X_{id}\right) \\ X_{id}(t+1) &= X_{id}(t) + V_{id}(t+1) \end{aligned} \tag{4}$$

Inertia weight adjusts the trade-off between exploration and exploitation capabilities of PSO. The less the inertia weight is, the more the exploration capability of PSO will be and vice versa. Commonly, it is decreased linearly during the course of the run, so that the search effort is mainly focused on exploration at initial stages and is focused more on exploitation at latter stages of the run.

3 Strategies in PSO for tackling discrete optimisation problems

In this section, all strategies adopted in PSO for tackling discrete problems are analysed deeply.

3.1 Rounding off

This is the most commonly used approach for tackling discrete/integer variables (Zhang and Xie 2003; Laskari et al. 2002; Venter and Sobieszczanski-Sobieski 2004; Salman et al. 2002; Onate Yumbla et al. 2008; Yare and Venayagamoorthy 2007; Jin et al. 2007; AlRashidi and El-Hawary 2007; Parsopoulos and Vrahatis 2002; Abdelaziz et al. 2009; Sivanagaraju et al. 2008; Ziari and Jalilian 2010; Eajal and El-Hawary 2010; Ziari et al. 2012; Yoshida et al. 2000; Fukuyama 2001). In this approach, discrete/integer variables are treated the same with continuous variables during the course of optimization but at the end of optimization, they are rounded off to the nearest discrete/integer value. Round off procedure, in some researches, is implemented at each iteration whereas in most cases, it is implemented once at the end of optimization. The results show that the accuracy is not much affected by rounding off. The main advantages of this approach are its simplicity and low computational cost. Although it exposes two crucial disadvantages; firstly, the solution may enter infeasible regions during rounding off. And secondly, the fitness value in rounded point may be so different from that in original point and it is also possible that the discrete point with more distance from the continuous optimum possesses better fitness in comparison to rounded off point.

3.2 Approaches for solving binary problems

Any optimization problem either continuous or discrete can be represented as a binary problem. Furthermore, some problems especially engineering problems are binary in nature and just can be represented by binary coding. So, devising approaches for effective handling of binary problems in PSO is of high importance. In Binary PSO, each particle contains a combination of 0's and 1's. Also, P_i's and P_g's are binary variables.

3.2.1 Conventional binary approach

In this approach, at each iteration t, the velocity update equation in (1) is used. But after using (1), V_{id} is mapped into interval [0,1] via sigmoid function as (Kennedy and Eberhart 1997; Ting et al. 2006; Khalil et al. 2006; Li et al. 2006; Liao et al. 2007; Liu and Gu 2007; Wu and Tsai 2008; Chang and Lu 2002; Yin and Lu 2009; Robinson 2005):

$$s\left(V_{id}\right) = \frac{1}{1 + exp\left(-V_{id}\right)} \tag{5}$$

Then a random number r in [0,1] is generated and

$$X_{id}\left(t+1\right) = \begin{cases} 1 \; r < s\left(V_{id}\right) \\ 0 \; Otherwise \end{cases} \tag{6}$$

That is, $\mathrm{V_{id}}$ represents the probability that bit $\mathrm{X_{id}}$ takes the 1 value. Indeed, in this approach the concept of velocity has been changed. The above processes are implemented for all dimensions and all particles. They are iterated over iterations till termination criterion is met.

One crucial issue in binary PSO is velocity clamping. According to (6), the more the velocity is, the less is the likelihood of changing the bit, thereby the less is the exploration capability of the algorithm. So, the velocity should be clamped in a small enough $\mathrm{V_{max}}$. Indeed in binary PSO, in contrast to continuous versions, smaller $\mathrm{V_{max}}$ result in more efficient exploration. Typically, $\mathrm{V_{max}}$ is set to 4, so that there is always at least a probability of $\mathrm{s}\left(\mathrm{V_{max}}\right) = 0.018$ for any bit to flip.

3.2.2 Angle modulation-based approach

This approach transforms a high-dimensional binary problem into a four-dimensional continuous problem. It utilises a trigonometric function as a bit string generator. The function is (Pampara et al. 2005; Liu et al. 2007):

$$g\left(X\right) = sin\left(2\pi\left(X - a\right).b.cos\left(A\right)\right) + d \tag{7}$$

where $A = 2\pi c\left(X - a\right)$ is a single element from a set of evenly separated intervals determined by the number of bits specified to be generated, b is maximum frequency of sine function and d indicates the vertical shift of function. Standard PSO is used to optimize the simple four-dimensional tuple (a, b, c, d). The optimized parameters are placed in (7) so that by sampling of resultant function at evenly spaced intervals, a bit for each interval is produced. The set of all generated bits represents the binary vector solution to the original problem. Generation of bit string involves taking the sampled points, feeding them into equation (7) and evaluating the output. If the output is positive, the corresponding bit is set to 1, otherwise, it is set to 0.

In summary, angle modulation-based approach significantly diminishes the complexity of problem by reducing its dimensionality. As expected, the results show more accuracy in lower computational time.

3.2.3 Boolean approach

In this approach, modified flight equations based on “XOR” and “OR” operators of Boolean algebra are utilised (Afshinmanesh et al. 2005).

$$V_i\left(t+1\right) = C_1 \otimes \left(P_i \oplus X_i\right) + C_2 \otimes \left(P_g \oplus X_i\right) \tag{8}$$

$$X_i\left(t+1\right) = X_i\left(t\right) \oplus V_i\left(t+1\right) \tag{9}$$

A noticeable feature of this approach is its velocity bounding mechanism which is inspired from “negative selection” mechanism existent in immune systems. In this mechanism at each iteration and for each particle the number of 1’s in the binary velocity vector V_i which called velocity length $\mathrm{L_i}$ is counted and if it is below a specified threshold L_{max}, V_i is considered as a non-self antigen and will not be changed. Otherwise, if $\mathrm{L_i}$ exceeds the threshold, V_i is considered as a self antigen and negative selection is applied to V_i such that randomly chosen 1’s in V_i are flip to 0’s until $L_i = L_{max}$.

In Marandi et al. (2006), the notion of inertia weight which is missing in (8) is incorporated so that the flight equations are as follows:

$$V_i\left(t+1\right) = \omega \odot V_i\left(t\right) + C_1 \otimes \left(P_i \oplus X_i\right) + C_2 \otimes \left(P_g \oplus X_i\right) \tag{10}$$

where $\odot$ represents “AND ” operator.

In an extension of Boolean PSO in order to enhance exploration capability a mutation operator with a prespecified probability is applied to particles’ velocities. This mutation operator just can flip 0 bits to 1and does not work in reverse (Deligkaris et al. 2009).

3.2.4 Quantum-based binary PSO

In this binary-based approach, each particle besides its position vector, possesses a quantum vector $Q_i = \{Q_{i1}, Q_{i2}, \ldots, Q_{in}\}$. $\mathrm{Q_{id}}$ represents the probability that $\mathrm{X_{id}}$ takes the value 1. That is, for each bit $\mathrm{X_{id}}$, a random number in [0,1] is generated and if it is below $\mathrm{Q_{id}}$, $\mathrm{X_{id}}$ takes the value 1. At each iteration, Q is updated as follows Shuyuan et al. (2004):

$$Q_{self-based}(t) = \alpha P_i(t) + \beta\,(1 - P_i(t)) \tag{11}$$

$$Q_{social-based}(t) = \alpha P_g(t) + \beta\left(1 - P_g(t)\right) \tag{12}$$

$$\mathrm{Q\,(t+1) = C_1\,Q\,(t) + C_2\,Q_{self-based}\,(t) + C_3\,Q_{social-based}\,(t)} \tag{13}$$

After updating Q by (11–13), X is updated. The main drawback of this approach is its heavy computational burden required for tuning additional parameters α and β.

3.2.5 Binary PSO based on estimation of distribution

Evolutionary paradigms that use information gathered during the optimization process to construct probabilistic models of the distribution of promising regions in search space and utilise these models to generate new solutions are called “estimation of distribution algorithms (EDA’s)”. Generation of new solutions is conducted by sampling the search space according to the estimated distribution. The distribution is re-estimated after each iteration. From EDA models, in Wang (2007), univariate marginal distribution (UMD) is applied. For applying UMD to binary PSO, firstly all local best’s are selected. Then based on these selected local best’s, UMD model estimates the distribution of promising regions in search space and uses a probability vector $P = (P_1, P_2, \ldots, P_n)$ to characterise the distribution of promising solutions where $\mathrm{P_d}$ is the probability that dth dimension of a promising solution take the value 1. The probability vector guides particles to search the binary space in the following way:

$$\text{If rand}\,(.) < \beta,\, then\ \mathrm{X_{id}\,(t+1)} = \begin{cases} 1 \text{ If rand}\,(.) < \mathrm{P_d} \\ 0 \text{ Otherwise} \end{cases} \tag{14}$$

Otherwise:

$$\mathrm{X_{id}\,(t+1) = P_{gd}\,(t)}$$

From (14), it is obvious that the larger the β is, the more elements of $\mathrm{X_i}$ (t) are sampled from vector P.

The probability vector P is initialised by the following rule:

$$\mathrm{P_d} = \frac{\sum_{\mathrm{i=1}}^{\mathrm{N_p}} \mathrm{P_{id}}}{\mathrm{N_p}} \tag{15}$$

At each iteration, P is updated via:

$$\mathrm{P_d} = (1-\lambda)\,\mathrm{P_d} + \lambda \frac{\sum_{\mathrm{i=1}}^{\mathrm{N_p}} \mathrm{P_{id}}}{\mathrm{N_p}} \tag{16}$$

where $\lambda \in (0, 1)$ is a learning rate.

3.3 Trinary PSO

Trinary PSO addresses problems which possess three different states for each bit. Three states are represented by three unity vectors; $1\measuredangle$-120°, $1\measuredangle 0°$ and $1\measuredangle$-120°. For initialisation, particles' dimensions are initialised randomly with one of above three states. Flight Equation in Trinary PSO is as follows Moradi and Fotuhi-Firuzabad (2008):

$$\mathrm{V_i}\,(\mathrm{t}+1)=\mathrm{V_i}\,(\mathrm{t})+\mathrm{C_1}\,\mathrm{r_1}\,\mathrm{P_i}+\mathrm{C_2}\,\mathrm{r_2}\,\mathrm{P_g} \tag{17}$$

After updating velocity, the phase of V_i is computed and mapped into interval [0,1] as the probability of states. Thus, the angles of states; $1\measuredangle$-120°, $1\measuredangle 0°$ and $1\measuredangle$-120° are mapped into 1/6, 1/2 and 5/6 respectively. Then, the differences between the phase of $\mathrm{V_i}$ and the three numbers (1/6, 1/2, 5/6) is calculated in each iteration and resulting numbers ($\mathrm{d_1}$, $\mathrm{d_2}$, $\mathrm{d_3}$) are transformed via following function.

$$\mathrm{T}\,(\mathrm{d_k})=\frac{\mathrm{a.exp}\,(\tan\,(\pi\,(0.5-\mathrm{d_k})))}{1+\mathrm{a.exp}\,(\tan\,(\pi\,(0.5-\mathrm{d_k})))} \tag{18}$$

where k = 1,2,3 and a is a constant.

After transformation by (18), a random number r in [0,1] is generated and new states of particles' dimensions are determined according to the following rules:

1. If r is more than all $\mathrm{T}\,(\mathrm{d_1})$, $\mathrm{T}\,(\mathrm{d_2})$ and $\mathrm{T}\,(\mathrm{d_3})$, or it is less than all of them, one state is chosen randomly for next iteration.
2. If r is more than all $\mathrm{T}\,(\mathrm{d_1})$, $\mathrm{T}\,(\mathrm{d_2})$ and $\mathrm{T}\,(\mathrm{d_3})$, $1\measuredangle$-120° is selected as the state for next iteration.
3. If r is less than all $\mathrm{T}\,(\mathrm{d_1})$, $\mathrm{T}\,(\mathrm{d_2})$ and $\mathrm{T}\,(\mathrm{d_3})$, one of the states associated with $\mathrm{d_1}$ and $\mathrm{d_2}$ is selected as the state for next iteration.

After updating particles, their objective function is calculated, then $\mathrm{P_i}$'s and $\mathrm{P_g}$'s are updated. The main issue in the proposed trinary PSO is that there exists an overlap between first and third states.

3.4 Penalty approach

In this approach, discrete variables are treated as continuous ones by penalising points which are existent at intervals. Penalty function is as follows Kitayama et al. (2006), Li et al. (2005), Kitayama and Yasuda (2006).

$$\mathrm{P}\,(\mathrm{X})=\sum_{\mathrm{i=1}}^{\mathrm{i=n_d}}\frac{1}{2}\left[\sin\frac{2\,\pi\,\left\{\mathrm{X_i^c}-0.25\left(\mathrm{d_{i,j+1}}+3\mathrm{d_{i,j}}\right)\right\}}{\mathrm{d_{i,j+1}}-\mathrm{d_{i,j}}}+1\right] \tag{19}$$

where $\mathrm{d_{i,j}}$ and $\mathrm{d_{i,j+1}}$ are the jth and (j + 1)th section of ith discrete variable, $\mathrm{X_i^c}$ represents the continuous value between them and $\mathrm{n_d}$ represents the number of discrete variables. According to (19) the penalty value is small around the discrete values and is large at regions away from discrete values. Augmented objective function is represented as:

$$\varphi\,(\mathrm{X})=\mathrm{f}\,(\mathrm{X})+\mathrm{SP}\,(\mathrm{X}) \tag{20}$$

where S represents penalty factor. For initialising S, $\mathrm{S}\,(\mathrm{X_i})$ for all particles are computed via:

$$\mathrm{S}\,(\mathrm{X_i})=1+\mathrm{P}\,(\mathrm{X_i}) \tag{21}$$

Then, initial value of S called S_{init} is determined as:

$$S_{init} = \min_i S(X_i) \tag{22}$$

During the course of optimization, at each iteration t, S is updated by:

$$S(t+1) = S(t).\exp(1 + P(X_i)) \tag{23}$$

The main drawback of penalty approach is it heavy computational burden which extremely limits its application in complex optimization problems.

3.5 Approaches based on modification of flight equations

In Liu et al. (2009), Al-Kazemi and Mohan (2005), Pan et al. (2008), Tasgetiren et al. (2007), Wu and Tsai (2011), Tao et al. (2010) PSO's conventional flight equations are modified to make it appropriate for discrete problems. In Liu et al. (2009), following new flight equations are applied to discrete optimization problems:

$$X_i^1(t) = X_i(t) + \alpha V_i(t) \tag{24}$$

$$X_i^2(t) = \beta.PMX\left[X_i^1(t), P_i(t)\right] \tag{25}$$

$$X_i(t+1) = \gamma PMX\left[X_i^2(t), P_{l_i}(t)\right] \tag{26}$$

where PMX and P_{l_i} represent partially matched crossover and particle i's neighborhood best, respectively.

In another approach called multi-phase discrete PSO, the flight equations are modified as Al-Kazemi and Mohan (2005):

$$V_{id}(t+1) = \mathrm{Min}\left(V_{max}, \mathrm{Max}\left(-V_{max}, V_{id}(t) + C_i(t)\left(P_{gd} - X_{id}\right)\right)\right) \tag{27}$$

$$X_{id}(t+1) = \begin{cases} 1 \text{ If } V_{id}(t+1) + X_{id}(t+1) > 0.5 \\ 0 \text{ Otherwise} \end{cases} \tag{28}$$

where $C_i(t)$ equals either 1 or -1 depending on the current phase of the corresponding particle. After each pcf (phase change frequency) number of iterations, the sign of $C_i(t)$ is reversed. The sign of $C_i(t)$ determines whether the particle i is attracted toward or repelled from the global best.

Moreover in Pan et al. (2008), Tasgetiren et al. (2007)), new flight equations particularly appropriate for flowshop scheduling problem is introduced. In Wu and Tsai (2011) a new flight equation for integer problems is devised and in Tao et al. (2010) new flight equations based on simple cyclic rules are proposed.

In an overall view on approaches proposed in literature which are based on modification of flight equations, it can be said that the proposed flight equations are normally appropriate for a particular problem and in most of cases, they do not work well in other problems.

3.6 Set-based approach

This approach uses a set-based representation scheme wherein particles' positions and velocities are represented as crisp sets and sets with possibilities respectively (Chen et al. 2010; Yue-Jiao et al. 2012).

If E is a crisp set, a set with possibilities V defined on E is represented by:

$$V = \{e/P(e) \,|\, e \in E\} \tag{29}$$

That is, each element $e \in E$ possesses a possibility $P(e) \in [0, 1]$ in V.

In this approach, the flight equations in (4) are used, so due to the new entities of velocities, the operators in (4) should be redefined. The redefined operators are as follows:

1. Coefficient × velocity (multiplication of a coefficient and a set with possibilities) is as follows:

$$cV = \left\{e/P'(e)\,|e \in E\right\} \tag{30}$$

where

$$P'(e) = \begin{cases} 1 & If\ cP(e) > 1 \\ cP(e) & Otherwise \end{cases} \tag{31}$$

2. Position-position (Subtraction of two crisp sets) is defined as: If A and B are two position sets, A−B is:

$$A - B = \{e|e \in A\ and\ e \notin B\} \tag{32}$$

3. Multiplication of a coefficient and a crisp set is defined as:

$$cE' = \left\{e/P'(e)\,|e \in E\right\} \tag{33}$$

where

$$P'(e) = \begin{cases} 1\ If\ e \in E'\, and\, c > 1 \\ c\ If\ e \in E'\, and\, 0 \le c \le 1 \\ 0\ If\ e \notin E' \end{cases} \tag{34}$$

4. Velocity plus velocity (Addition of two sets with possibilities) is defined as follows.

If $V_1 = \{e/P_1(e)\,|e \in E\}$ and $V_2 = \{e/P_2(e)\,|e \in E\}$ are two velocity vectors, then:

$$V_1 + V_2 = \{e/max\,(P_1(e)\,, P_2(e))\,|e \in E\} \tag{35}$$

According to the operators defined in 1 through 4, particles velocities are updated at each iteration.

For updating position, a random number α in (0,1) is generated for each particle. Then, for each element e in the dth dimension, if the corresponding possibility P (e) in $\mathrm{V_{id}}$ is not smaller than α, element e is reversed in the crisp set, that is:

$$cut_\alpha\,(V_{id}) = \{e|e/P(e) \in V_{id}\ and\ P(e) \ge \alpha\} \tag{36}$$

After construction of $\mathrm{cut}_\alpha\,(V_{id})$, new position of particle i is determined by learning from the elements in $\mathrm{cut}_\alpha\,(V_{id})$.

3.7 Hybrid approach

In Nema et al. (2008) PSO is hybridized with branch and bound (BB) algorithm which is an efficient classic technique for tackling discrete variables. By this hybridization, both global search capability of PSO and rapid convergence capability of BB are benefited to form a more efficient approach.

At the beginning of this hybrid approach, the BB is invoked for determining an initial feasible solution for PSO and at each iteration, whenever an improvement in P_g is achieved, it is passed over to the BB module as the starting point. Actually, in this approach, BB is not called at every iteration, but it is called when P_g is improved. When BB completes, P_g is updated and the PSO resumes.

The main advantage of this approach is the need to less function evaluations due to the deterministic nature and quickness of BB. On the other hand, incorporation of BB which is derivative-based, limits the applicability of the hybrid approach.

3.8 Other approaches

Besides all the approaches explained for tackling discrete variables in PSO, there exist some other approaches in literature which are not explained here due to space limitation (Clerc 2004; Pang et al. 2004a, b; Niasar et al. 2009; Hoffmann et al. 2011; Sha and Hsu 2006; Qin et al. 2011). Table 1 has tabulated all different discrete variable handling approaches adopted in PSO, their applicability, pros and cons.

4 Applications of DPSO in power system problems

Most of the power system problems can be transformed into optimization problems and most of them contain discrete/integer/binary variables. For instance, location of a compensator represents an integer variable, or reactive power of a static var compensator and the setting of a tap changer introduce discrete variables. Also, the states of switches in distribution systems (close/open) represent binary variables.

In literature discrete PSO has been used for optimal power flow (Onate Yumbla et al. 2008; AlRashidi and El-Hawary 2007), optimal scheduling of generator maintenance (Yare and Venayagamoorthy 2007), transmission network expansion planning (Jin et al. 2007), distribution system reconfiguration (Sivanagaraju et al. 2008; Liu and Gu 2007; Chang and Lu 2002; Yin and Lu 2009; Wu and Tsai 2011), allocation of active power line controllers (Ziari and Jalilian 2010), optimal capacitor placement (Eajal and El-Hawary 2010; Khalil et al. 2006), integrated distribution system planning (Ziari et al. 2012), reactive power control (Fukuyama 2001), unit commitment (Ting et al. 2006). Reliability analysis (Robinson 2005), power system defensive islanding (Liu et al. 2007) and switch placement (Moradi and Fotuhi-Firuzabad 2008).

For solving each of the mentioned power system problems, a particular type of discrete/binary variable handling strategy has been adopted. Round off strategy is used in (Onate Yumbla et al. 2008; Yare and Venayagamoorthy 2007; Jin et al. 2007; AlRashidi and El-Hawary 2007; Sivanagaraju et al. 2008; Ziari and Jalilian 2010; Eajal and El-Hawary 2010; Ziari et al. 2012; Yoshida et al. 2000; Fukuyama 2001), conventional binary PSO has been applied in Ting et al. (2006), Khalil et al. (2006), Liu and Gu (2007), Robinson (2005), angle-modulated PSO is used in Liu et al. (2007) trinary PSO has been used in Moradi and Fotuhi-Firuzabad (2008) and finally an approach based on modification of flight equations is introduced in Wu and Tsai (2011). Table 2 tabulates the PSO applications in discrete power system problems and the type of discrete PSO adopted in them.

5 Conclusions and future research directions

In this paper, all strategies adopted in PSO for tackling discrete problems have been analysed. Conclusively, the followings are proposed as some directions for future research.

- In round off approach, at the end of continuous optimization, when final continuous optimum is found, instead of rounding off, it is recommended that the two nearest discrete

Table 1 Different discrete variable handling approaches adopted in PSO, their applicability, pros and cons

Approach	References	Pros	Cons
Rounding off	Abdelaziz et al. (2009), AlRashidi and El-Hawary (2007), Eajal and El-Hawary (2010), Fukuyama (2001), Jin et al. (2007), Laskari et al. (2002), Onate Yumbla et al. (2008), Parsopoulos and Vrahatis (2002), Salman et al. (2002), Sivanagaraju et al. (2008), Venter and Sobieszczanski-Sobieski (2004), Yare and Venayagamoorthy (2007), Yoshida et al. (2000), Zhang and Xie (2003), Ziari and Jalilian (2010), Ziari et al. (2012)	It is so simple and imposes no additional computational burden on optimization algorithm. In most of cases, the results are not affected significantly by rounding off procedure	The solution may enter infeasible regions during rounding off. The fitness value in rounded off point may be so different from that in original point and it is also possible that the discrete point farther from the continuous optimum possesses better fitness in comparison to rounded off point
Conventional binary PSO	Chang and Lu (2002), Kennedy and Eberhart (1997), Khalil et al. (2006), Li et al. (2006), Liao et al. (2007), Liu and Gu (2007), Robinson (2005), Ting et al. (2006), Wu and Tsai (2008), Yin and Lu (2009)	It is simple	The nonlinear sigmoid function incorporates nonlinearity to the algorithm. It is much sensitive to high velocities
Angle-modulated PSO	Liu et al. (2007), Pampara et al. (2005)	It greatly diminishes the complexity of problems by transforming high-dimensional binary problems into a continuous four-dimensional problem	No strategy for tuning the new parameters that are introduced by angle-modulation approach has been provided
Boolean PSO	Afshinmanesh et al. (2005), Deligkaris et al. (2009), Marandi et al. (2006)	Unlike conventional PSO, it does not incorporate any nonlinearity	Introduces a more efficient velocity bounding strategy based on negative-selection
Quantum PSO	Shuyuan et al. (2004)	It is relatively simple	Tuning its parameters (α and β) imposes additional computational burden

Table 1 continued

Approach	References	Pros	Cons
Estimation of distribution-based approach	Wang (2007)		Tuning its parameters (λ *and* β) is difficult. It is complex in comparison to most other binary approaches
Trinary PSO	Moradi and Fotuhi-Firuzabad (2008)	It is the unique approach existent in literature especially proposed for problems with three-state variables	No strategy for tuning its parameter (a) has been provided
Penalty function approach	Kitayama et al. (2006), Kitayama and Yasuda (2006), Li et al. (2005)	It imposes heavy computational burden	Prevents problems such as reaching to an infeasible final solution
Approaches based on flight equation modification	Al-Kazemi and Mohan (2005), Liu et al. (2009), Pan et al. (2008); Tao et al. (2010), Tasgetiren et al. (2007), Wu and Tsai (2011)		Normally, the proposed flight equations are merely appropriate for a particular type of problems
Set-based approach	Chen et al. (2010), Yue-Jiao et al. (2012)		It is not simple
Hybrid approach	Nema et al. (2008)	Needs less function evaluations due to the deterministic nature and quickness of BB	Incorporation of BB which is derivative-based, limits its applicability
Other Approaches	Clerc (2004), Hoffmann et al. (2011), Niasar et al. (2009), Pang et al. (2004a, b) Qin et al. (2011), Sha and Hsu (2006)		Mostly, are somehow similar to previous approaches or not applicable to a wide range of problems

Table 2 PSO applications in discrete power system problems and their type of discrete PSO

Reference number	Type of discrete PSO	Application in power system
Onate Yumbla et al. (2008)	Round off	Optimal power flow subject to security constraints
Yare and Venayagamoorthy (2007)	Round off	Optimal scheduling of generator maintenance
Jin et al. (2007)	Round off	Transmission network expansion planning
AlRashidi and El-Hawary (2007)	Round off	Optimal power flow considering the valve loading effects
Sivanagaraju et al. (2008)	Round off	Network reconfiguration for loss reduction and load balancing
Ziari and Jalilian (2010)	Round off	Allocation and sizing of multiple active power-line conditioners
Eajal and El-Hawary (2010)	Round off	Optimal capacitor placement and sizing in unbalanced distribution systems with harmonics consideration
Ziari et al. (2012)	Round off	Integrated distribution systems planning to improve reliability
Yoshida et al. (2000)	Round off	Reactive power and voltage control
Fukuyama (2001)	Round off	Power system state estimation
Ting et al. (2006)	Conventional binary PSO	Unit Commitment
Khalil et al. (2006)	Conventional binary PSO	Optimal placement and sizing of capacitor banks in radial distribution feeders
Liu and Gu (2007)	Conventional binary PSO	Skeleton-network reconfiguration
Wu and Tsai (2008)	Conventional binary PSO	Feeder reconfiguration in distribution systems
Chang and Lu (2002)	Conventional binary PSO	Feeder reconfiguration for load factor improvement
Yin and Lu (2009)	Conventional binary PSO	Feeder reconfiguration in distribution systems
Robinson (2005)	Conventional binary PSO	Reliability analysis of bulk power systems
Liu et al. (2007)	Angle-modulated binary PSO	Defensive islanding of large scale power systems
Moradi and Fotuhi-Firuzabad (2008)	Trinary PSO	Optimal switch placement in distribution systems
Wu and Tsai (2011)	Based on new flight equations	Feeder reconfiguration in distribution systems

values to the continuous optimum are evaluated and the one with better fitness is considered as the optimum discrete point.

- In conventional binary PSO, replacing sigmoid function with other functions may lead to superior results.
- Devising a time-variant (preferably time-deceasing) velocity bound for conventional binary PSO may result in better results.
- In Boolean PSO, transforming other basic PSO variants (like constricted PSO) to Boolean structure may lead to more promising results.

- In binary PSO, using different probability distribution functions for generating the probability vector (the vector whose entries indicate the probability that their corresponding bits in position vector take the value 1) is recommended. Maybe certain probability functions lead to outperformance in binary PSO.
- In penalty approach, devising more appropriate and simple penalty functions is of high value.
- Approaches like estimation of distribution-based PSO and set-based PSO should be tested on a wider range of discrete problems in order to evaluate their performance.
- Devising new flight equations more appropriate for discrete problems can be a potential research direction.
- Focusing more on binary PSO and devising more efficient binary coded approaches for solving continuous and binary problems may direct attractions toward binary coded PSO in early future.
- So limited number of theoretical analysis on discrete variable handling in PSO has been conducted.

References

Abdelaziz A, Mohammed F, Mekhamer S, Badr M (2009) Distribution Systems Reconfiguration using a modified particle swarm optimization algorithm. Electr Power Syst Res 79(11):1521–1530

Afshinmanesh F, Marandi A, Rahimi-Kian A (2005) A novel binary particle swarm optimization method using artificial immune system. In: IEEE, pp 217–220

Al-Kazemi B, Mohan C (2005) Discrete multi-phase particle swarm optimization. Inf Process Evol Algorithms, 305–327

AlRashidi M, El-Hawary M (2007) Hybrid particle swarm optimization approach for solving the discrete OPF problem considering the valve loading effects. IEEE Trans Power Syst 22(4):2030–2038

Chang R, Lu C (2002) Feeder reconfiguration for load factor improvement. In: IEEE, vol 982, pp 980–984

Chen WN, Zhang J, Chung HSH, Zhong WL, Wu WG, Shi YH (2010) A novel set-based particle swarm optimization method for discrete optimization problems. IEEE Trans Evol Comput 14(2):278–300

Clerc M (2004) Discrete particle swarm optimization illustrated by the traveling salesman problem. New Optim Tech Eng 141:219–239

Deligkaris KV, Zaharis ZD, Kampitaki DG, Goudos SK, Rekanos IT, Spasos MN (2009) Thinned planar array design using Boolean PSO with velocity mutation. IEEE Trans Magn 45(3):1490–1493

Eajal AA, El-Hawary M (2010) Optimal capacitor placement and sizing in unbalanced distribution systems with harmonics consideration using particle swarm optimization. IEEE Trans Power Deliv 25(3):1734–1741

Eberhart RC, Shi Y, Kennedy J (2001) Swarm intelligence. Elsevier, Amsterdam

Fukuyama Y (2001) State estimation and optimal setting of voltage regulator in distribution systems. In: IEEE, vol 932, pp 930–935

Hoffmann M, MA1/4hlenthaler M, Helwig S, Wanka R (2011) Discrete particle swarm optimization for TSP: theoretical results and experimental evaluations. Adapt Intell Syst, 416–427

Jin YX, Cheng HZ, Yan J, Zhang L (2007) New discrete method for particle swarm optimization and its application in transmission network expansion planning. Electr Power Syst Res 77(3):227–233

Kennedy J, Eberhart R (1995) Particle swarm optimization. In: IEEE international conference on, proceedings neural networks, Nov/Dec 1995, vol 1944 . pp 1942–1948. doi:10.1109/icnn.1995.488968

Kennedy J, Eberhart RC (1997) A discrete binary version of the particle swarm algorithm. In: IEEE, vol. 4105, pp 4104–4108

Khalil TM, Youssef HKM, Aziz MMA (2006) A binary particle swarm optimization for optimal placement and sizing of capacitor banks in radial distribution feeders with distorted substation voltages. TM Khalil, HKM Youseef, MM Abdel Aziz, 129–135

Kitayama S, Yasuda K (2006) A method for mixed integer programming problems by particle swarm optimization. Electr Eng Jpn 157(2):40–49

Kitayama S, Arakawa M, Yamazaki K (2006) Penalty function approach for the mixed discrete nonlinear problems by particle swarm optimization. Struct Multidiscip Optim 32(3):191–202

Laskari EC, Parsopoulos KE, Vrahatis MN (2002) Particle swarm optimization for integer programming. In: IEEE, pp 1582–1587

Li D, Wang B, KitaYama S, Yamazaki K, Arakawa M (2005) Application of particle swarm optimization to the mixed discrete non-linear problems. In: Artificial intelligence applications and innovations, vol 187. IFIP—The International Federation for Information Processing. Springer, USA, pp 315–324. doi:10.1007/0-387-29295-0_34

Li X, Tian P, Hua J, Zhong N (2006) A hybrid discrete particle swarm optimization for the traveling salesman problem. Simul Evol Learn, 181–188

Liao CJ, Tseng CT, Luarn P (2007) A discrete version of particle swarm optimization for flowshop scheduling problems. Comput Oper Res 34(10):3099–3111

Liu Y, Gu X (2007) Skeleton-network reconfiguration based on topological characteristics of scale-free networks and discrete particle swarm optimization. IEEE Trans Power Syst 22(3):1267–1274

Liu W, Liu L, Cartes DA (2007) Angle modulated particle swarm optimization based defensive islanding of large scale power systems. In: IEEE, pp 1–8

Liu H, Liu X, Wang Q (2009) Routing optimization for dispatching vehicles based on an improved discrete particle swarm optimization algorithm with mutation operation. In: IEEE, pp 624–627

Marandi A, Afshinmanesh F, Shahabadi M, Bahrami F (2006) Boolean particle swarm optimization and its application to the design of a dual-band dual-polarized planar antenna. In: IEEE, pp 3212–3218

Moradi A, Fotuhi-Firuzabad M (2008) Optimal switch placement in distribution systems using trinary particle swarm optimization algorithm. IEEE Trans Power Deliv 23(1):271–279

Nema S, Goulermas J, Sparrow G, Cook P (2008) A hybrid particle swarm branch-and-bound (HPB) optimizer for mixed discrete nonlinear programming. IEEE Trans Syst Man Cybern Part A Syst Hum 38(6):1411–1424

Niasar NS, Shanbezade J, Perdam M, Mohajeri M (2009) Discrete fuzzy particle swarm optimization for solving traveling salesman problem. In: IEEE, pp 162–165

Onate Yumbla PE, Ramirez JM (2008) Optimal power flow subject to security constraints solved with a particle swarm optimizer. IEEE Trans Power Syst 23(1):33–40

Pampara G, Franken N, Engelbrecht A (2005) Combining particle swarm optimisation with angle modulation to solve binary problems. In: IEEE, vol. 81, pp 89–96

Pan QK, Fatih Tasgetiren M (2008) A discrete particle swarm optimization algorithm for the no-wait flowshop scheduling problem. Comput Oper Res 35(9):2807–2839

Pang W, Wang K, Zhou C, Dong L (2004a) Fuzzy discrete particle swarm optimization for solving traveling salesman problem. In: IEEE, pp 796–800

Pang W, Wang KP, Zhou CG, Dong LJ, Liu M, Zhang HY, Wang JY (2004b) Modified particle swarm optimization based on space transformation for solving traveling salesman problem. In: IEEE, vol 2344, pp 2342–2346

Parsopoulos KE, Vrahatis MN (2002) Recent approaches to global optimization problems through particle swarm optimization. Nat Comput 1(2):235–306

Qin J, Li X, Yin Y (2011) An algorithmic framework of discrete particle swarm optimization. Appl Soft Comput

Robinson DG (2005) Reliability analysis of bulk power systems using swarm intelligence. In: IEEE, pp 96–102

Salman A, Ahmad I, Al-Madani S (2002) Particle swarm optimization for task assignment problem. Microprocess Microsyst 26(8):363–371

Sha D, Hsu CY (2006) A hybrid particle swarm optimization for job shop scheduling problem. Comput Ind Eng 51(4):791–808

Shi Y, Eberhart R (1998) A modified particle swarm optimizer. In: The 1998 IEEE international conference on, evolutionary computation proceedings, 1998. IEEE world congress on computational intelligence, 4–9 May 1998. pp 69–73. doi:10.1109/icec.1998.699146

Shi Y, Eberhart RC (1999) Empirical study of particle swarm optimization. In: Proceedings of the 1999 congress on, evolutionary computation, 1999. CEC 99, 1999, vol 1953, p 1950. doi:10.1109/cec.1999.785511

Shuyuan Y, Min W, Licheng j (2004) A quantum particle swarm optimization. In: Congress on, evolutionary computation, CEC2004. 19–23 June 2004, vol. 321, pp 320–324. doi:10.1109/cec.2004.1330874

Sivanagaraju S, Rao JV, Raju PS (2008) Discrete particle swarm optimization to network reconfiguration for loss reduction and load balancing. Electr Power Compon Syst 36(5):513–524

Tao Q, Chang H, Yi Y, Gu C, Li W (2010) A novel cyclic discrete optimization framework for particle swarm optimization. Adv Intell Comput Theories Appl, 166–174

Tasgetiren MF, Suganthan P, Pan QQ (2007) A discrete particle swarm optimization algorithm for the generalized traveling salesman problem. In: ACM, pp 158–167

Ting T, Rao M, Loo C (2006) A novel approach for unit commitment problem via an effective hybrid particle swarm optimization. IEEE Trans Power Syst 21(1):411–418

Venter G, Sobieszczanski-Sobieski J (2004) Multidisciplinary optimization of a transport aircraft wing using particle swarm optimization. Struct Multidiscip Optim 26(1):121–131

Wang J (2007) A novel discrete particle swarm optimization based on estimation of distribution. Adv Intell Comput Theories Appl Aspects Artif Intell, 791–802

Wu WC, Tsai MS (2011) Application of enhanced integer coded particle swarm optimization for distribution system feeder reconfiguration. IEEE Trans Power Syst, (99):1–9

Wu WC, Tsai MS (2008) Feeder reconfiguration using binary coding particle swarm optimization. Int J Control Autom Syst 6(4):488–494

Yare Y, Venayagamoorthy GK (2007) Optimal scheduling of generator maintenance using modified discrete particle swarm optimization. In: IEEE, pp 1–8

Yin SA, Lu CN (2009) Distribution feeder scheduling considering variable load profile and outage costs. IEEE Trans Power Syst 24(2):652–660

Yoshida H, Kawata K, Fukuyama Y, Takayama S, Nakanishi Y (2000) A particle swarm optimization for reactive power and voltage control considering voltage security assessment. IEEE Trans Power Syst 15(4):1232–1239

Yue-Jiao G, Jun Z, Ou L, Rui-Zhang H, Chung HSH, Yu-Hui S (2012) Optimizing the vehicle routing problem with time windows: a discrete particle swarm optimization approach. IEEE Trans Syst Man Cybern Part C Appl Rev 42(2):254–267. doi:10.1109/tsmcc.2011.2148712

Zhang WJ, Xie XF (2003) DEPSO: hybrid particle swarm with differential evolution operator. In: IEEE, vol 3814, pp 3816–3821

Ziari I, Jalilian A A new approach for allocation and sizing of multiple active power-line conditioners. IEEE Trans Power Deliv 25(2):1026–1035

Ziari I, Ledwich G, Ghosh A, Platt G (2012) Integrated distribution systems planning to improve reliability under load growth. IEEE Trans Power Del (99):1–1