International Journal of Computer Science & Applications
© 2007 Technomathematics Research Foundation
Vol. 4 Issue 3, pp 69-83

Binary Particle Swarm Optimization Based Defensive Islanding Of Large Scale Power Systems

Wenxin Liu[1] `wliu@caps.fsu.edu`
Li Liu[1] `lliu@caps.fsu.edu`
David A. Cartes[1] `dave@caps.fsu.edu`

Ganesh K. Venayagamoorthy[2]
`gkumar@ieee.org`

[1] Center for Advanced Power Systems (CAPS), Florida State University
2000 Levy Avenue Building A, Tallahassee, FL 32310, U.S.A.
[2] Department of Electrical and Computer Engineering, University of Missouri at Rolla
132 Emerson Electric Co. Hall, Rolla, MO 65409, U.S.A.

Abstract

Power system defensive islanding is an efficient way to avoid catastrophic wide area blackouts, such as the 2003 North American Blackout. Finding defensive islands of large-scale power systems is a combinatorial explosion problem. Thus, it is very difficult to find an optimal solution, if it exists, within reasonable time using analytical methods. This paper presents to utilize the computational efficiency property of binary particle swarm optimization to find some efficient splitting solutions for large-scale power systems. The solutions are optimized based on a fitness function considering the real power balance between generations and loads in islands, the relative importance of customers, and the desired number of islands. Besides providing information about the opening of transmission lines, the algorithm can also provide necessary load shedding information. Furthermore, the algorithm can provide a number of candidate solutions in order to select one satisfying the transmission system capacity constraint. Simulations with power systems of different scales demonstrate the accuracy and effectiveness of the proposed algorithm.

Keywords: power system islanding, splitting strategies, particle swarm optimization.

1 Introduction

Although power systems are designed to be robust and tolerant to disturbances, they may become unstable during severe faults, especially when they are operated close to their stability limits. The sources of severe disturbances include earthquakes, hurricanes, human operation errors, control system failures, hidden failures in protection system, malicious attacks, etc. These disturbances may cause the system to lose stability and even lead to catastrophic failures [1], such as the North American Blackout on August 14, 2003.

Studies show that many blackouts (including the 2003 Blackout [2]) could have been avoided and significant losses could have been reduced if proper defensive islanding operations were taken prior to or following a catastrophe. Defensive islanding, also called as system splitting or controlled system separation, is different from passive islanding. Passive islanding is not under control and usually results from damage and protection. Defensive islanding intentionally split an interconnected power system into islands by opening selected transmission lines [3]. During dangerous events, if defensive islanding and load shedding are properly deployed, although power system is running in a less versatile and degraded state, catastrophic losses can be avoided because the blackout is isolated and prevented from further spreading.

In literature, two major groups have investigated the islanding problem. Vittal's group contributed a lot to this area [1-2, 4-7]. In their papers, the generators are first grouped according to normal forms [4] or slow coherency [1-2, 5-7] and then some searching algorithms are employed to find the minimum cutting set from the interface network between the generator groups. Since the slow coherency algorithm considers the dynamic behaviors of large-scale power systems, the solutions cannot only maintain good active power generation and load balance, but also provide good dynamic transient performance during islanding operations. Simulation studies demonstrated that the proposed algorithms could effectively improve system stability and avoid the possibility of wide area blackouts. Lu's group also presented a method for system splitting based on its steady state stability [3, 8]. To narrow the searching space for large-scale power systems, the original power network is first simplified by graph theory, and then Ordered Binary Decision Diagrams (OBDD) are used to find the splitting strategies candidates.

As shown in [3], the balanced partition problem is an NP- complete problem. It is very difficult to find the optimal solution for a large-scale power system. Currently, most of the islanding algorithms are optimized based on either simplified or selected subset of the original power systems. The reason is that the deterministic searching algorithms have difficulties exploring a large NP-hard searching space. Because of these simplifications, it is possible to miss some better solutions that may exist for the original systems. Therefore, it is better to consider the original power system configuration data directly. Then, the next question is how to conquer the computational complexity. The proposed solution in this paper is to take advantage of the computational efficiency of random search algorithms to improve the searching speed and exploring capability. It has to be admitted that the random searching algorithms cannot guarantee the optimality of solutions. However, even if it is possible, it is not necessary to search this optimal solution for large-scale power systems. As it is know, it may take unacceptably long time to find the optimal solution if the searching space is intractable. Furthermore, there may be several splitting solutions having the same performance for large-scale power systems. Considering that the decision time is the most critical issue for defensive islanding, finding an efficient solution is much more significant than finding an actual optimal solution in extra long time. Thus, the objective for this problem is to find some efficient solutions as fast as possible.

This paper proposes to use the Binary Particle Swarm Optimization (BPSO) to find some efficient solutions directly from the original power system bus and line matrices. The optimization is based on a fitness function that is defined according to the amount and the importance of working loads in islands. Besides generating the open line information to form the desired number of islands, the algorithm can also provide the necessary load shedding information if the available generation is not enough to power all loads in an island. During the optimization process, a number of good candidate solutions are recorded in case the top one solution does not satisfy the Transmission System Capacity (TSC) constraint. In the worst case, if all of the candidate solutions fail the TSC, the algorithm can avoid the known failed solutions and find some additional solutions that are suboptimal compared with the previously generated solutions, in the sense of the defined fitness function. Since the algorithm can generate desired number of islands, it has the potential to be combined with the slow coherency based generator grouping. Simulation results with different scales of power systems demonstrate the effectiveness of the proposed BPSO algorithm.

The rest of the paper is organized as follows. Section 2 presents an overview of the problems to be considered during islanding and an introduction to the BPSO algorithm. Section 3 presents the detail introduction to the BPSO based power system splitting algorithm. Section 4 provides some simulation results performed with different scales of power systems, and finally, conclusions and discussions in Section 5.

2 Background

2.1 Power System Defensive Islanding

The issues to be considered during power system defensive islanding are summarized as follows.

2.1.1 Generation/Load Balance And Load Shedding

In each island, the balance between generation and load must be maintained. Otherwise, the islanding operation may result in frequency and voltage droop or even blackout within the islands. If the power generation in some island is insufficient to power all loads, necessary load shedding information should be provided together with the transmission lines opening information. In practical power systems, local reactive power compensators can compensate reactive power for unbalance. Thus, only active power balance is considered in most research.

2.1.2 Transmission System Capacity Constraint

Since the capacity of transmission system is limited, it is necessary to check if they are loaded above their thermal or static stability limits. Thus, the islanding algorithm should be able to provide a number of candidate solutions for the TSC constraint check. If the candidate solutions fail, the algorithm should be able to provide another set of solutions until a feasible solution is found.

2.1.3 Priorities Of Loads

Some customers (or loads), such as hospitals, airports, and government buildings, should have higher priority to receive power supply over other loads. Thus, loads should be classified into different classes according to their difference in importance, such as critical loads and non-critical loads, and given corresponding priority indexes. The performance of an islanding solution should not only be judged by the amount of working loads but also be evaluated according to the priorities of the loads.

2.1.4 Computational Efficiency

Since the status of transmission lines can be either open or closed, a power system with n transmission lines will have a total of 2n possible solutions. As mentioned in Section I, the objective is to find some efficient solutions within limited time not trying to find the optimal one. Even finding the efficient solutions from a huge searching space is a difficult job. Thanks to the recent development of evolutionary computation techniques, a number of random search based optimization algorithms can be chosen.

2.1.5 Isolation Of Possible Impacted Region

During some predictable events, such as the approaching of a destructive hurricane, it is desirable to isolate the possible impacted region from the rest of system. This kind of isolation can prevent the possible spreading of blackout formed in the impacted region. To minimize possible losses, further splitting of the isolated region may help.

2.1.6 Dynamic Response Consideration

To form islands, some transmission lines need to be opened, the interruption of power flow through these transmission lines will result in transient dynamics. These transients are harmful and need to be damped out in time. If some islands do not have enough damping ability, these oscillations will exist for a long time and may result in dangerous consequence. For this reason, it is better for the islanding algorithm to consider some dynamic response indices. As mentioned before, slow coherency is a good candidate. Other dynamic response indexes, which require less computation time and less information about the power system, can also be applied to the islanding problem. But they are not addressed in this paper.

2.2 Binary Particle Swarm Optimization

PSO algorithm is a relative new collaborative computation technique that is first proposed by Kennedy and Eberhart [9]. The algorithm is derived from the social psychological theory and has been found to be robust in solving problems featuring nonlinearity and nondifferentiability, multiple optima, and high dimensionality through adaptation.

Like other evolutionary computation techniques, PSO is a population based search algorithm and is initialized with a population of random solutions, called particles. Unlike other evolutionary computation techniques, each particle in PSO is also associated with a velocity. Particles fly through the search space with velocities, which are dynamically adjusted according to their and the swarm's historical behaviors. Therefore, the particles have a tendency to fly towards better and better solutions over the course of search process.

PSO algorithm is simple in concept, easy to implement and computationally efficient. The updating rules for PSO algorithm are listed in (1) and (2).

$$v_i = w \cdot v_i + c_1 \cdot rand_1 \cdot (x_p - x_i) + c_2 \cdot rand_2 \cdot (x_g - x_i) \tag{1}$$

$$x_i = x_i + v_i \tag{2}$$

where w, c_1, and c_2 are the inertia weight, cognitive acceleration and social acceleration constants respectively; $rand_1$ and $rand_2$ are two random numbers; x_i represents the location of the ith particle; x_p represents the best solution the particle has achieved so far - *pbest*; x_g represents the best location obtained so far by all particles in the population - *gbest*; v_i represents the velocity of the particle.

The inertia parameter, w, controls the influence of previous velocity on current velocity. Without the inertia parameter, the particle may erratically jump around in the search space making it more difficult for the algorithm to converge. The inertia parameter can drastically improve the manner in which the PSO algorithm converges to a solution by smoothing particle trajectories.

A limit, $\left[V_i^{\min}, V_i^{\max}\right]$, may be placed on v_i to ensure that the velocities are acceptable. Velocities that are not within the range are clamped. Care should be taken in selecting the value of $V_i^{\min}$ and $V_i^{\max}$. If it is too small, step sizes are very small which may cause the algorithm to be trapped in a local minimum, and may increase the number of iterations to reach a good solution.

The above canonical PSO algorithm operates in continuous space. However, many optimization problems are set in a space featuring discrete, qualitative distinctions between variables and between levels of variables. To enable the canonical PSO algorithm operates on discrete binary variables, a binary version of PSO is proposed in [10]. In BPSO, the updating rule of the position vector is modified according to (3).

$$\text{if } (rand_3 < S(v_i)),\ \text{then } x_i = 1,\ \text{else } x_i = 0 \tag{3}$$

where $S(v_i)$ is a sigmoid limiting transformation function defined as $S(v_i) = 1/(1+e^{-v_i})$, and rand3 is a random number.

3 BPSO Based Power System Splitting Algorithm

The BPSO based power system splitting algorithm is illustrated using the flowchart shown in Fig. 1.

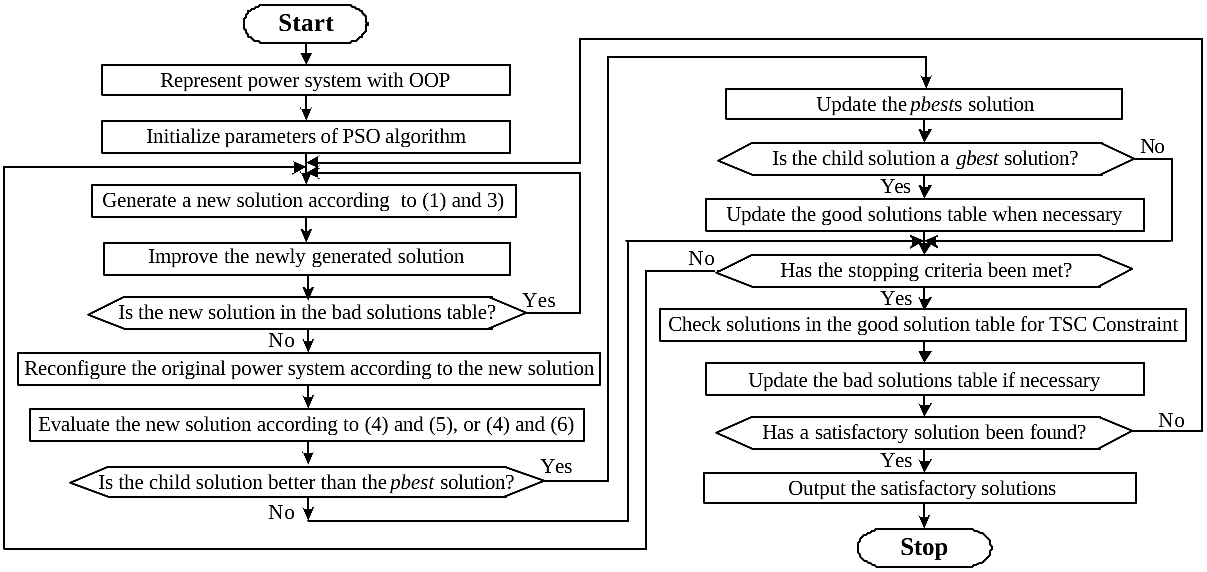

Figure 1: Flowchart of the BPSO based splitting algorithm

3.1 Preprocessing

In this stage, the original power system is represented using n_{bus} (the number of buses) "bus objects". Fields describing the bus object include the index of the bus, the indexes of the buses that it is connected to, the type of the bus (generator, load, and intermediate), the importance of the bus (critical and non-critical), the rated active power generation/consumption, etc. This kind of representation can be easily implemented with the Object Oriented Programming (OOP) languages. In this paper, the algorithm is coded in Microsoft© Visual J# 2005 Express Edition.

It is necessary to note that, the above representation does not considerate the situation when multiple transmission lines exist between two buses. For those cases, the multiple transmission lines are considered together as one combined line. If the combined line is opened according to a solution, then all the corresponding transmission lines need to be opened.

For the previously mentioned predictable events, the possible impacted region should be considered as one bus, either a generator bus or a load bus, depending on the net active power of the region. Assuming that there are n_{reg} lines in the impacted region, for a power system with n_{line} distinct transmission lines, the number of transmission lines to be considered is n_{line}-n_{reg}+1, which is the dimension of the PSO solution vector. The elements of the binary vector can be either '1' or '0' to represent the status of the transmission line, '1' means the transmission line is "connected" and '0' means "disconnected".

Other parameters of the BPSO algorithm are selected as follows. The population size is set to be 20; w, c_1 and c_2 are set to 0.8, 2, and 2 respectively; top 20 solutions are recorded in the good solution table during optimization process; and the optimization process is set to terminate after 200 iterations.

3.2 Optimization

The following fitness function is defined to evaluate the performance of candidate solutions.

$$F = \left(\sum_{i=1}^{n_{isl}} \sum_{j=1}^{m_i} L_{ij} \cdot p_{ij} \right) \left(1 + e^{-\frac{(n_d - n_a)^2}{k}} \right) \tag{4}$$

where n_{isl} is the number of islands formed by a candidate solution; m_i is the number of working loads in the ith island; L_{ij} is the active power rating of the j_{th} load in the ith island; and p_{ij} is the priority index used to represent the importance property of load L_{ij}; n_d and n_a are the desired and actual island numbers respectively; and k is a positive constant used to decide the impact of the number of islands to the fitness function.

During optimization process, a candidate solution is compared with a bad solutions table and a good solutions table. The bad solutions table includes all known solutions that fail the TSC constraint test and the good solutions table includes a number of good candidate solutions found during the optimization process. If a candidate solution is found listed in the bad solution table, this candidate solution is discarded and a new one is generated until finding one not in the bad solutions table. If the new solution is better than the worst solution in the good solutions table, the worst solution in the table is replaced with the new solution.

Due to the random nature of the optimization algorithm, the generated solution may result in some isolated buses. An isolated bus means it is not connected to any island. An isolated bus can be any of the three types of buses (a generator bus, a load bus, or an transmission bus). The isolated buses cannot be named as an island because an island needs both generation and load buses connected. These impractical solutions will reduce the computational efficiency, waste storage space, and harm the comparison process described above. To improve the algorithm in this regard, the three cases and the corresponding solutions are introduced as follows.

If the isolated bus is a load bus, the isolated bus is assigned to a connected island according to redundant generations of the connected islands and the rating of the isolated load. If there are more than one isolated load buses, these buses are assigned to appropriate islands one after another. The largest load is assigned to the island with the most redundant generation. After the related information is updated, the second largest load is assigned to the island with the most redundant generation according to this updated information. The process continues until all of the isolated load buses have been assigned.

If the isolated bus is a generator bus, then this bus is assigned to some island to support the loads in that island. The assignment is based on the redundant generations of the connected islands and the rating of the isolated generator bus. The largest generator is assigned to the island with the least redundant generation. After related information is updated, the remaining generators are assigned one after another. The process is similar to the assignment of isolated load buses.

Since transmission buses are connected to neither generators nor loads, the assignment of the isolated transmission buses does not affect the fitness. But these isolated buses will generate excessive "same" solutions, which may impact the comparison with good/bad solutions tables, and reduce the efficiency of the algorithm. To minimize the number of these solutions, the isolated transmission buses are assigned according to an objective of balancing the number of buses in islands.

After the newly generated solution has been found to be a reasonable one, this solution needs to be evaluated according to (4). To do this, the original power system needs to be reconfigured by opening some transmission lines according to the solution vector. This can be achieved by updating the connected bus information of all the affected buses. Then the reconfigured power network needs to be investigated for the number of islands and the buses in each island. Since an island is meaningful only when there are both generator(s) and load(s) buses connected, the investigation of the power system configuration should begin with some generator bus. According to the connected bus information, all of the connected buses can be found. After that, the remaining generator buses that are not included in the existing islands are investigated one after another until all generator buses have been investigated. During programming, the above process is simplified by defining a recursive function.

After the configuration of islands has been found, the loads to be powered are determined. This problem can be modeled as the bin-packing problem in computational complexity theory. The bin-packing problem is an NP hard problem. For large-scale problems, it is very difficult to find the optimal solution within limited time. To reduce the computation burden for large-scale power systems, greedy algorithms can be applied [14]. In this way, the load shedding information is obtained as a byproduct during the selection of working loads.

For a large-scale power system, there may be more than one candidate solutions that have the same fitness, thus a secondary criteria is needed to sort the candidate solutions. For this purpose, an index is defined according to (5).

$$index1 = \sum_{i=1}^{n_{isl}} \left| nob_i - \overline{n} \right| \tag{5}$$

This criteria higher preference is given to solutions that have the closest numbers of buses in islands. In (5), n_{isl} is the number of islands, nob_i is the number of buses in the i^{th} island, and $\overline{n}$ is the average value of buses in all the islands.

3.3 Postprocessing

After optimization, the top 20 candidate solutions in the good solutions table are decoded to display the islanding solutions, such as the transmission lines to be opened, the number of islands to be formed, the number of buses, the total generations and loads in each island, and the loads to be shed in order to maintain the active power generation and load balance.

The candidate solutions have been optimized according to the defined fitness function but have not been checked for the TSC constraint. A solution is not a feasible one unless it passes TSC constraint test. Since there are more than one candidate solutions, if a solution fails the TSC constraint test, one just needs to check the rest. Usually a feasible one can be found. In the worst case, if all of the candidate solutions fail the TSC constraint test, we can save all of these solutions to the bad solutions table and start the program again. Since the algorithm can avoid solutions in the bad solution table, we can always find some solutions that satisfy the TSC constraint, even though the newly generated solutions may not be as good as the previously generated ones in the sense of the defined fitness function.

4. Simulation Results

The proposed splitting algorithm is tested with three power systems of different scales, which are the simplified WECC 9-bus system, the IEEE 30-bus system [11], and the IEEE 118-bus system [11].

4.1 WSCC 9-bus Power System

The configuration and parameters of the simplified WECC 9-bus 9-line power system are shown in Fig. 2., where dark circles stand for generator buses, and transparent circles stand for load buses or transmission buses. The number inside a circle is the index of the bus, and the number next to a circle is the active power generation or load rating of the corresponding bus. The lines connecting two circles represent transmission lines. This system has a total generation of 345MW and a total load of 305MW.

The objective is to split the system into two islands. As shown in Fig. 2, only one solution exists, which is also the unique optimal solution. The vector representation of the optimal solution is shown in Fig. 3, where the two bits corresponding to line 4-6 and line 8-9 are zeros.

Simulation studies show that the algorithm can always find the optimal solution within 5 iterations. Since the population size is 20, it means that at most 100 (5*20) out of 512 (29) possible solutions are investigated. Thus, we can conclude that the proposed algorithm is accurate and computationally efficient.

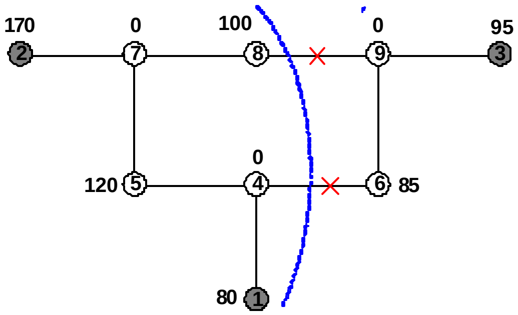

Figure 2: WSCC 9-bus 9-line power system

Line No.:	1-4	2-7	3-9	4-5	4-6	5-7	6-9	7-8	8-9
Solution Vector:	1	1	1	1	0	1	1	1	0

Figure 3: Vector representation of the optimal solution

4.2 IEEE 30-bus Power System

The configuration and parameters of the IEEE 30-bus, 41-line power system are shown in the Fig. 4. Notations in Fig. 4 are the same as those in Fig. 2. In this system, there are 6 generators with a total generation of 180MW, and with 16 loads of 137.5MW in all.

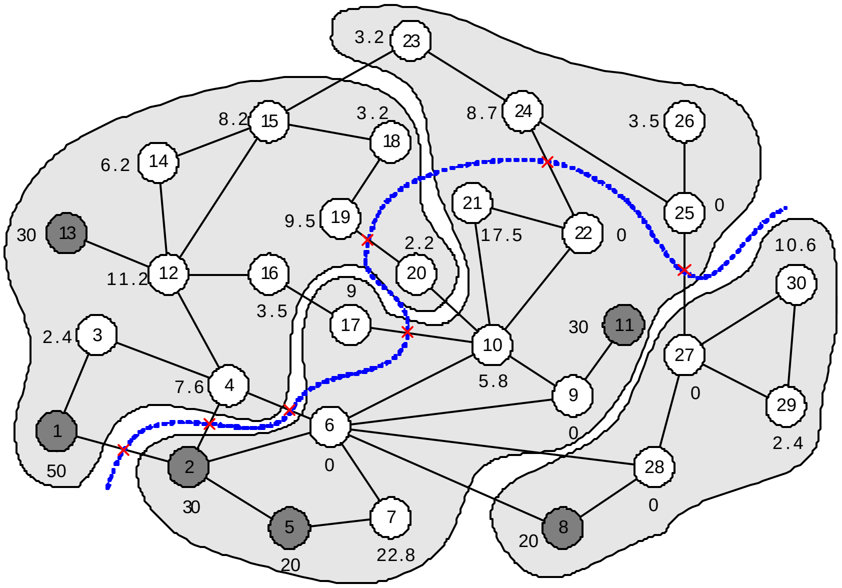

Figure 4: IEEE 30-bus 41-line power system (Dotted line for 4.2.1, shaded islands for 4.2.2)

Since it is difficult to illustrate solutions using figures for large-scale systems, three tests are conducted with this system. The first test is to split the system into two islands. The objective of this test is to evaluate the algorithm's computational efficiency. The second test is to split the system into three islands. The objective is to test if the algorithm can split a system into a desired number of islands. The third test is to split a tight system, which means that the system has an insufficient total generation to power all the loads. The objective is to test if the algorithm can provide necessary load shedding information.

4.2.1 Split The System Into Two Islands

Top 10 solutions of this simulation according to the fitness function in (4) are listed in Table 1. Since all the loads can be powered according to the splitting solution, these solutions are sorted with an increase of indexes defined in (5). In Table 1, first column gives the indexes of the candidate solutions; the second column gives the number and the indexes of the buses in each island; the third column lists the total active power generation and load in each island; and the fourth column lists the transmission lines to be opened in order to form the two islands.

To help understand the solution table, the first candidate solution is illustrated in Fig. 4, where the lines need to be opened is marked with crosses, and the two islands are separated by a dotted line.

No.	Islands Info		Opened Lines
	Buses	P_G/P_L (MW)	
1	15 (1, 3~4, 12~19, 23~26)	80/76.2	1-2, 2-4, 4-6, 10-17, 19-20, 22-24, 25-27
	15 (2, 5~11, 20~22, 27~30)	100/61.3	
2	15 (1, 3~4, 12~16, 18~20, 23~26)	80/69.4	1-2, 2-4, 4-6, 10-20, 16-17, 22-24, 25-27
	15 (2, 5~11, 17, 21~22, 27~30)	100/68.1	
3	16 (1~4, 12~16, 18~20, 23~26)	110/69.4	2-5, 2-6, 4-6, 10-20, 16-17, 22-24, 25-27
	14 (5~11, 17, 21~22, 27~30)	70/68.1	
4	14 (1~4, 12~19, 23~24)	110/72.7	2-5, 2-6, 4-6, 10-17, 19-20, 22-24, 24-25
	16 (5~11, 20~22, 25~30)	70/64.8	
5	14 (1, 3~4, 12~16, 18~19, 23~ 26)	80/67.2	1-2, 2-4, 4-6, 16-17, 19-20, 22-24, 25-27
	16 (2, 5~11, 17, 20~22, 27~30)	100/70.3	
6	16 (1, 3~4, 12~20, 23~26)	80/78.4	1-2, 2-4, 4-6, 10-17, 10-20, 22-24, 25-27
	14 (2, 5, 6~11, 21~22, 27~30)	100/59.1	
7	13 (1, 3~4, 12~20, 23)	80/66.2	1-2, 2-4, 4-6, 10-17, 10-20, 23-24
	17 (2, 5~11, 21~22, 24~30)	100/71.3	
8	17 (1~4, 12~17, 23~27, 29~30)	100/76.5	2-5, 2-6, 4-6, 10-17, 15-18, 22-24, 27-28
	13 (5~11, 18~22, 28)	70/61	
9	12 (1, 3~4, 12~16, 18~20, 23)	80/57.2	1-2, 2-4, 4-6, 10-20, 16-17, 23-24
	18 (2, 5~11, 17, 21~22, 24~30)	100/80.3	
10	11 (1, 3~4, 12~15, 23~26)	80/55	1-2, 2-4, 4-6, 16-17, 19-20, 23-24
	19 (2, 5~11, 16~22, 27~30)	100/82.5	

Table 1: Top 10 solutions to split the 30-bus system into two islands

The simulation time to do a 50-iteration optimization with a Pentium IV 3.6G Hz CPU PC is about 1 second. Simulation studies show that the algorithm always converges within the first 5 iterations. That means only 100 possible solutions are investigated. Considering that there is a total number of 241 ˜ 2.199*1012 possible solutions for the 41-line power system, it can be concluded that the proposed algorithm is very computationally efficient and fast.

4.2.2 Split The System Into Three Islands

In this test, the desired number of islands is set to 3. The parameter setting of the BPSO algorithm is the same as in the first test. After 200 iterations, the top 10 good candidate solutions found by the algorithm are shown in Table 2. The first candidate solution in Table 2 is also illustrated in Fig. 4, where the three islands are circled within three shaded areas.

4.2.3 Split A Tight System Into Two Islands

According to the original data, the total generations (180 MW) are larger than the total loads (137.5 MW), thus load shedding is not necessary for the top candidate solutions. To make the problem more challenging and to demonstrate the algorithm's capability in providing necessary load shedding information, the generations of bus 1 and 2 are reduced to 10MW to make a total generation of 120MW, which is not enough to power all loads. Similarly, the optimization process stops after 200 iterations. The top 10 good solutions are listed in Table 3. In this table, the amount of total working loads and the loads to shed are listed in columns five and six respectively. For example, according to the first solution, in the second island, the total generations is less than the total loads, thus the loads connected to buses 10 and 23 need to be shed, and the final total working load in the two islands is 119.5 MW.

No.	Islands Info		Opened Lines
	Buses	P_G/P_L (MW)	
1	11 (1, 3~4, 12~16, 18~20)	80/54	1-2, 2-4, 4-6, 6-8, 6-28, 10-20, 15-23, 16-17, 25-27
	14 (2, 5~7, 9~11, 17, 21~26)	80/70.5	
	5 (8, 27~30)	20/13	
2	12 (1, 3~4, 12~16, 23~26)	80/54.5	1-2, 2-4, 4-6, 6-8, 6-28, 15-18, 16-17, 22-24, 25-27
	13 (2, 5~7, 9~11, 17~22)	80/70	
	5 (8, 27~30)	20/13	
3	14 (1, 3~4, 12~16, 18~19, 23~26)	80/67.2	1-2, 2-4, 4-6, 6-8, 6-28, 16-17, 19-20, 22-24, 25-27
	11 (2, 5~7, 9~11, 17, 20, 21~22)	80/57.3	
	5 (8, 27~30)	20/13	
4	12 (1, 3~4, 12~15, 22~26)	80/51	1-2, 2-4, 4-6, 6-8, 6-28, 10-22, 12-16, 15-18, 21-22,25-27
	13 (2, 5~7, 9~11, 16 ~21)	80/73.5	
	5 (8, 27~30)	20/13	
5	15 (1, 3~4, 12~19, 23~26)	80/76.2	1-2, 2-4, 2-6, 4-6, 6-7, 10-17, 19-20, 22-24, 25-27
	3 (2, 5, 7)	50/22.8	
	12 (6, 8~11, 20~22, 27~30)	50/38.5	
6	13 (1, 3~4, 12~16, 18~20, 23~24)	80/65.9	1-2, 2-4, 2-6, 4-6, 6-7, 10-20, 16-17, 22-24, 24-25
	3 (2, 5, 7)	50/22.9	
	14 (6, 8~11, 17, 21~22, 25~30)	50/48.8	
7	14 (1, 3~4, 12~16, 18~19, 23~26)	80/67.2	1-2, 2-4, 2-6, 4-6, 6-7, 16-17, 19-20, 22-24, 25-27
	3 (2, 5, 7)	50/22.8	
	13 (6, 8~11, 17, 20~22, 27~30)	50/47.5	
8	17 (1, 3~4, 12~16, 18, 23~30)	80/70.7	1-2, 2-4, 2-6, 4-6, 6-7, 6-28, 16-17, 18-19, 22-24
	3 (2, 5, 7)	50/22.8	
	10 (6, 8~11, 17, 19~22)	50/44	
9	20 (1~7, 9~11, 15~24)	130/103.6	4-12, 6-8, 6-28, 12-15, 12-16, 14-15, 24-25
	7 (8, 25~30)	20/16.5	
	3 (12~14)	30/17.4	
10	3 (1, 3, 4)	50/10	1-2, 2-4, 4-6, 4-12, 10-17, 12-15, 14-15
	22 (2, 5~11, 15, 18~30)	100/97.6	
	5 (12~14, 16~17)	30/29.9	

Table 2: Top 10 solutions to split the 30-bus system into three islands

No.	Islands Info		Opened Lines	Working load	Loads to shed
	Buses	P_G/P_L (MW)			
1	20 (1~8, 12~16, 18~20, 27~30)	90/89.8	6-9, 6-10, 10-20, 15-23, 16-17, 25-27	119.5	10, 24, 26
	10 (9~11, 17, 21~26)	30/47.7			
2	3 (1, 3, 4)	10/10	1-2, 2-4, 4-6, 4-12,	119.5	16, 18, 20, 23, 26, 29
	27 (2, 5~30)	110/127.5			
3	17 (1~8, 21~22, 24~30)	60/75.5	4-12, 6-9, 6-10, 10-21, 10-22, 23-24	119.4	3, 4, 14, 18, 23, 26
	13 (9~20, 23)	60/62			
4	8 (1, 3~4, 12~14, 16~17)	40/39.9	1-2, 2-4, 4-6, 10-17, 12-15,14-15	119.4	10, 18, 23, 26, 29
	22 (2, 5~11, 15, 18~30)	80/97.6			
5	25 (1~11, 15, 18~30)	90/107.6	4-12, 10-17, 12-15, 14-15	119.4	10, 18, 23, 26, 29
	5 (12~14, 16~17)	30/29.9			
6	21 (1~2, 5~ 11, 17, 19~22, 24~30)	90/92	1-3, 2-4, 4-6, 16-17, 18-19, 23-24	119.2	14, 16, 18, 23
	9 (3, 4, 12~16, 18, 23)	30/45.5			
7	15 (1~5, 7, 12~18, 23~24)	70/86	2-6, 4-6, 6-7, 10-17, 18-19, 22-24, 24-25	119.2	14, 16, 18, 23
	15 (6, 8~11, 19~22, 25~30)	50/51.5			
8	13 (1~4, 12~15, 18~20, 23~24)	50/62.4	2-5, 2-6, 4-6, 10-20, 12-16, 22-24, 24-25	119.0	14, 16, 18, 23, 29
	17 (5~11, 16~17, 21~22, 25~30)	70/75.1			
9	23 (1~8, 12~16, 21~30)	90/107.8	6-9, 6-10, 10-21, 15-18, 16-17	118.7	14, 16, 23, 26, 29
	7 (9~11, 17 ~20)	30/29.7			
10	22 (1~3, 5~11, 16~17, 21~30)	90/89.4	2-4, 3-4, 4-6, 10-20, 12-16, 15-23	118.3	4, 14, 18, 27
	8 (4, 12~15, 18~20)	30/48.1			

Table 3: top 10 solutions to split the 30-bus system into two islands.

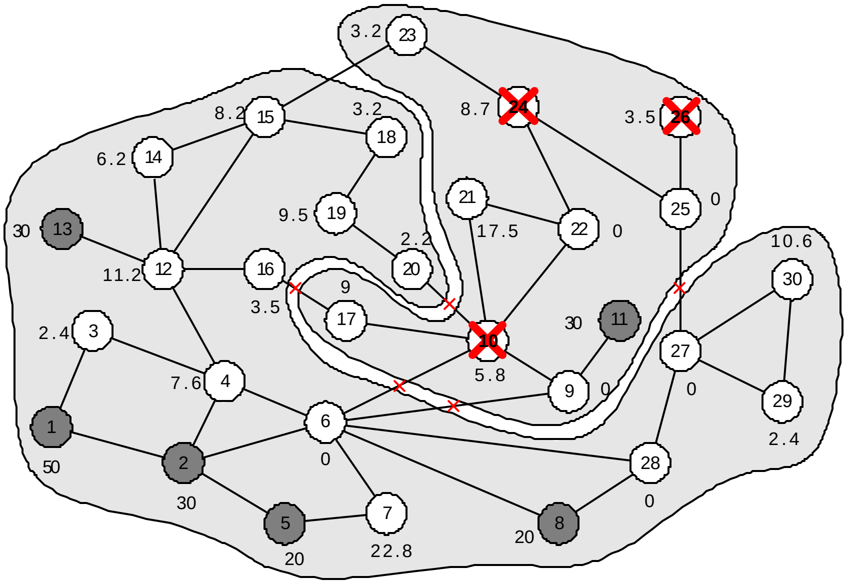

Figure 5: Illustration of the first candidate solution in Table 3.

The first candidate solution is illustrated with Fig. 5, where the transmission lines to be opened are marked with smaller red crosses and the loads to shed are marked with bigger red crosses. Since the total active power generation is not enough to support all the loads in the smaller island, the loads connected to buses 10, 24, and 26 need to be shed.

4.3 IEEE 118-Bus Power System

The detail data of the IEEE 118-bus 179-line power system can be found in [12]. This system has 15 generator buses, 93 load buses, and 10 transmission buses. The bus and transmission line data can be found in [12]. Considering the huge number of possible solutions ($7.6625*10^{53} \sim 2^{179}$), the system is good to test if the proposed algorithm can be applied to large-scale power systems.

4.3.1 Split The 118-Bus System Into Two Islands

The algorithm is run for 500 iterations and the top 10 candidate solutions are listed in Table 4. The simulation time for the 500 iterations is approximate one minute with the above-mentioned computer. Considering the huge search space, the proposed algorithm is very computationally efficient.

No	Islands Info		Opened Lines
	Buses	P_G/P_L (MW)	
1	59 (1~20, 26, 30~31, 33~67, 117)	1823/1791	17-113, 20-21, 25-26, 29-31, 31-32, 47-69, 49-69, 65-68
	59 (21~25, 27~29, 32, 68~116, 118)	1962.4/1841	
2	39 (1~32, 34, 36, 72, 113~115, 117)	1022/1006	15-33, 24-70, 30-38, 34-37, 34-43, 35-36, 71-72
	79 (33, 35, 37~71, 73~112, 116, 118)	2763.4/2626	
3	80 (1~75, 113~117)	2559.4/2425	68-81, 69-77, 75-77, 76-118
	38 (76~112, 118)	1226/1207	
4	34 (1~21, 25~33, 113~115, 117)	1022/897	1934, 21-22, 23-25, 23-32, 30-38, 33-37
	84 (22~24, 34~112, 116, 118)	2763.4/2735	
5	90 (1~83, 99, 113~118)	2906.4/2813	80-96 80-97, 80-98, 82-96, 83-84, 83-85, 99-100
	28 (84~98, 100~112)	879/819	
6	95 (1~87, 96~97, 113~118)	2910.4/2880	80-98, 80-99, 85-88, 85-89, 94-96, 95-96
	23 (88~95, 98~112)	875/752	
7	17 (1~16, 117)	488/483	8-30, 15-17, 15-19, 15-33, 16-17
	101 (17~116, 118)	3297.4/3149	
8	101 (1~22, 30, 33~69, 75~112, 116~118)	3251.4/3217	17-31, 17-113, 22-23, 26-30, 69-70, 70-75, 74-75
	17 (23~29, 31~32, 70~74, 113~115)	534/415	
9	12 (3~11, 13~15)	450/367	1-3, 3-12, 7-12, 8-30, 11-12, 12-14, 15-17, 15-19, 15-33
	106 (1~2, 12, 16~118)	3335.4/3265	
10	113 (1~24, 26, 30~114, 116~118)	3565.4/3498	23-25, 25-26, 27-32, 29-31, 114-115
	5 (25, 27~29, 115)	220/134	

Table 4: Top 10 solutions to split the 118-bus system into two islands

An example of the optimization process is shown in Figs. 6 and 7, where the fitness function and the total working load are plotted versus the number of iterations respectively. According to the definition of the fitness function, the fitness is not only decided by the total amount of working load but also the number of the islands. This explains the oscillations at the beginning of the optimization process in Fig. 7.

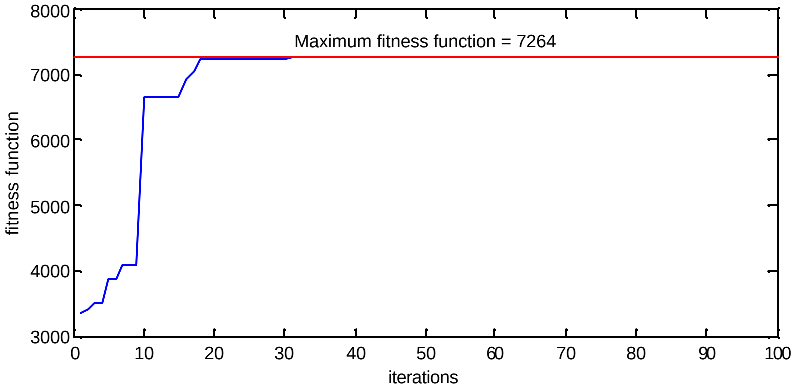

Figure 6: Fitness function updating process of the IEEE 118-bus power system

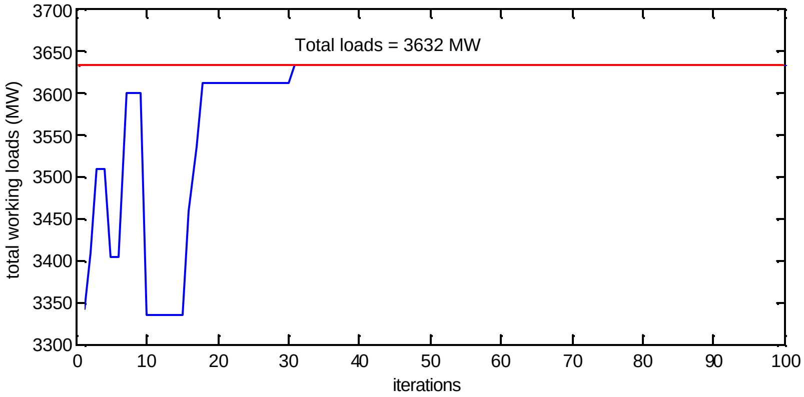

Figure 7: Working load updating process of the IEEE 118-bus power system

It is necessary to note that, though some good candidate solution can be found within 50 iterations, the optimization process will proceed. From Table 4, it can be seen that there are many equivalent solutions (in the sense of the defined fitness function) for large-scale power systems. Given more time to the optimization process, the algorithm will have a better chance to find multiple good candidate solutions.

4.3.2 Split Of A Possible Impacted Region

In this test, it is assumed that part of the system is to be impacted by some upcoming event, such as a dangerous hurricane. To isolate the Possible Impacted Region (PIR), the PIR is considered as a whole by defining a bus called PIR bus. If the total generation is larger than total load at the PIR, then the PIR bus is modeled as generator bus, otherwise, the PIR bus is modeled as a load bus. During simulation study, it is assumed that the PIR include 14 buses, i.e. buses 1~14. Since total generation is 488MW, which is larger than the total load of 348MW, the PIR is modeled as an equivalent generator bus with a net generation of 140MW. The PIR bus is connected to the rest part of the power system through four transmission lines (lines 8-30, 12-16, 12-117, and 13-15). The idea is explained using Fig. 8.

To sort solutions with the same fitness, the second criteria defined in (5) is now changed to the amount of loads in the PIR Island as defined in (6). A solution with less loading in the PIR Island is given more preference.

$$index2 = \sum_{i=1}^{n_{PIR}} L_i \cdot p_i \tag{6}$$

where n_{PIR} is the number of loads in the PIR.

Again, the algorithm is run for 500 iterations and the top 10 candidate solutions for this test are listed in Table 5.

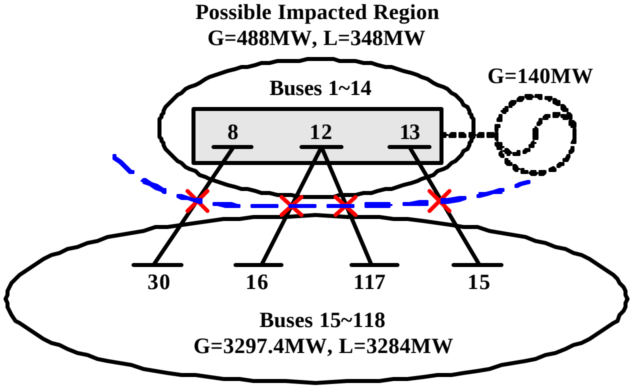

Figure 8: The consideration for possible impacted region

No.	Islands Info		Opened Lines
	Buses	P_G/P_L (MW)	
1	14 (1~14)	488/348	13-15, 12-16, 12-117, 8-30
	104 (15~118)	3297.4/3284	
2	15 (1~14, 117)	488/368	13-15, 12-16, 8-30
	103 (15~116, 118)	3297.4/3264	
3	16 (1~14, 16, 117)	488/396	13-15, 8-30, 16-17
	102 (15, 17~116, 118)	3297.4/3239	
4	30 (1~17, 26~33, 38, 113~115, 117)	802/760	15-19, 17-18, 23-32, 25-26, 25-27, 33-37, 37-38, 38-65
	88 (18~25, 34~37, 39~112, 116, 118)	2983.4/2872	
5	39 (1~33, 69, 113~115, 117)	1022/939	19-34, 24-70, 30-38,33-37, 71-72
	79 (34~68, 70~112, 116, 118)	2763.4/2693	
6	41 (1~33, 38, 71~73, 113~115, 117)	1022/945	19-34, 24-70, 33-37, 37-38, 38-65, 70-71
	77 (34~37, 39~70, 74~112, 116, 118)	2763.4/2687	
7	67 (1~23, 25~45, 49~67, 113~115, 117)	2043/1966	23-24, 45-46, 47-49, 48-49, 49-69, 65-68
	51 (24, 46~48, 68~112, 116, 118)	1742.4/1666	
8	81 (1~75, 81, 113~117)	2559.4/2425	69-77, 75-77, 75-118, 80-81
	37 (76~80, 82~112, 118)	1226/1207	
9	89 (1~81, 98~99, 113~118)	2906.4/2773	77-82, 80-96, 80-97, 98-100, 99-100
	29 (82~97, 100~112)	879/859	
10	102 (1~81, 98~118)	3174.4/3079	77-82, 80-100, 80-101, 92-100, 92-102, 94-100
	16 (82~97)	611/553	

Table 5: Top 10 solutions to isolate the possible impacted region

Since the PIR has enough generation to power all the loads in the region (488MW vs. 348MW) and the rest part of the system's generation and loads can be maintain balanced (3297.4MW vs. 3284MW), open the four lines that connect the PIR and the rest of the system will result in the best solution.

5. Conclusions And Discussions

Based on an analysis of the problems to be considered during islanding decision, this paper has proposed a binary particle swarm optimization algorithm for power system splitting. The optimization process is based on a fitness function pursuing a balance between total generation and load in islands. The relative importance of loads and the desired number of islands are also modeled in the fitness function. To evaluate its efficiency and accuracy, the proposed algorithm is simulated with power systems of different scales. Simulation results demonstrate that the proposed algorithm can find the only optimal solution for small-scale power systems and a number of efficient candidate solutions for large-scale power system in a fast manner.

It should be note that the transmission line losses are not considered in the current simulation. However, taking this loss into consideration will only affect the checking for load shedding necessity. To cover this loss, one only needs to consider an extra percent of the total load. The modification will not affect other part of the algorithm.

In this paper, the studies demonstrate that the BPSO algorithm can split a power system into a desired number of islands. Additionally, other requirements can also be realized if constraints can be properly defined. For example, if dynamic response needs to be considered, the generator groups can be found first based on slow coherency, and then the fitness function need to be modified to generate the highest fitness to a solution that can provide the most similarity to the desired generator groups. A heuristic response test is required to ensure reasonable, efficient, and timely solutions.

References

[1] Wang, X. and Vittal, V., "System islanding using minimal cutsets with minimum net flow," *2004 IEEE PES Power Systems Conference and Exposition*, vol. 1, pp. 379-384, October 2004.

[2] Yang, B., Vittal, V., and Heydt, G.T., "Slow-Coherency-Based Controlled Islanding-A Demonstration of the Approach on the August 14, 2003 Blackout Scenario," *IEEE Transactions on Power Systems*, vol. 21, no. 4, pp. 1840-1847, November 2006.

[3] Zhao, Q., Sun, K., Zheng, D., Ma, J., and Lu, Q., "A study of system splitting strategies for island operation of power system: a two-phase method based on OBDDs," *IEEE Transactions on Power Systems*, vol. 18, no. 4, pp. 1556, November 2003.

[4] Vittal, V., Kliemann, W., Ni, Y., Chapman, D.G., Silk, A.D., and Sobajic, D.J., "Determination of generator groupings for an islanding scheme in the Manitoba hydro system using the method of normal forms," *IEEE Transactions on Power Systems*, Vol. 13, No. 4, pp. 1345-1351, November 1998.

[5] Jung, J., Liu, C.C., Tanimoto, S.L., and Vittal, V., "Adaptation in load shedding under vulnerable operating conditions," *IEEE Transactions on Power Systems*, vol. 17, no. 4, pp. 1199-1205, November 2002.

[6] You, H., Vittal, V., and Yang, Z., "Self-healing in power systems: an approach using islanding and rate of frequency decline-based load shedding," *IEEE Transactions on Power Systems*, vol. 18, no. 1, pp. 174-181, February 2003.

[7] You, H., Vittal, V., and Wang, X., "Slow coherency-based islanding," *IEEE Transactions on Power Systems*, vol. 19, no. 1, pp. 183-491, February 2004.

[8] Sun, K., Zheng, D., and Lu, Q., "Splitting strategies for islanding operation of large-scale power systems using OBDD-based methods," *IEEE Transactions on Power Systems*, vol. 18, No.2, pp. 912-923, May 2003.

[9] Kennedy J., and Eberhart, R., "Particle swarm optimization," *IEEE International Conference on Neural Networks*, vol. IV, pp. 1942-1948, Perth, Australia, 1995.

[10] Kennedy J., and Eberhart, R., " A Discrete Binary Version of the Particle Swarm Optimization," *the 1997 International Conference on Neural Networks*, vol. IV, pp. 4104-4108, Perth, Australia, 1997.

[11] Power systems test case archive, University of Washington, available at: http://www.ee.washington.edu/research/pstca/.

[12] Liu, W., Cartes, D.A., and Venayagamoorthy, G.K., "Particle Swarm Optimization Based Defensive Islanding Of Large Scale Power Systems," *the 2006 International Joint Conference on Neural networks*, Vancouver, Canada, July 16-21, 2006.