energies

MDPI

Article

Bacteria Foraging Reinforcement Learning for Risk-Based Economic Dispatch via Knowledge Transfer

Chuanjia Han [1], Bo Yang [2], Tao Bao [1], Tao Yu [1,*] and Xiaoshun Zhang [1]

[1] School of Electric Power, South China University of Technology, Guangzhou 510640, China; chuanjia71@126.com (C.H.); baotaowork@foxmail.com (T.B.); xszhang1990@126.com (X.Z.)
[2] Faculty of Electric Power Engineering, Kunming University of Science and Technology, Kunming 650500, China; yangbo_ac@outlook.com
* Correspondence: taoyu1@scut.edu.cn; Tel.: +86-20-2223-6205

Academic Editor: Josep M. Guerrero
Received: 18 January 2017; Accepted: 24 April 2017; Published: 6 May 2017

Abstract: This paper proposes a novel bacteria foraging reinforcement learning with knowledge transfer method for risk-based economic dispatch, in which the economic dispatch is integrated with risk assessment theory to represent the uncertainties of active power demand and contingencies during power system operations. Moreover, a multi-agent collaboration is employed to accelerate the convergence of knowledge matrix, which is decomposed into several lower dimension sub-matrices via a knowledge extension, thus the curse of dimension can be effectively avoided. Besides, the convergence rate of bacteria foraging reinforcement learning is increased dramatically through a knowledge transfer after obtaining the optimal knowledge matrices of source tasks in pre-learning. The performance of bacteria foraging reinforcement learning has been thoroughly evaluated on IEEE RTS-79 system. Simulation results demonstrate that it can outperform conventional artificial intelligence algorithms in terms of global convergence and convergence rate.

Keywords: bacteria foraging reinforcement learning; risk-based economic dispatch; knowledge matrix; knowledge transfer

1. Introduction

In recent years, the interconnection of regional power grids and high voltage, long-distance and bulk capacity transmission [1] have become new trends of power systems integrated with large-scale renewable energy sources such as wind and solar energy [2–5], which however may result in severe challenges to the secure and stable operation of power grids. In order to obtain an appropriate trade-off between system security and economical operation, risk assessment theory has been introduced into automatic generation control (AGC) [6] so as to improve the economic dispatch (EC) in the presence of various operation risks [7].

The security constrained optimal power flow (SCOPF) is an extension of conversional optimal power flow (OPF). The operation constraints of assumed contingencies are employed to enhance the EC security [8,9]. With the development of SCOPF theory, the 1990s, several studies have discussed challenges and future trends of SCOPF [10,11]. With the development of SCOPF theory, the '$N-1$' deterministic security regulations have been widely adopted as a well-known benchmark of SCOPF nowadays. However, such method is inadequate to quantitatively analyse the operation risks, which may sometimes obtain an over-conservative result. To remedy this flaw, the probabilistic risks based OPF and relevant algorithms were developed in [12,13]. Meanwhile, some researchers investigated the application of binding contingencies identification and post-contingency model approximation, such

that the size of SCOPF can be considerably reduced [14]. In addition, [15] proposed a novel risk-based security-constrained EC, in which a risk index was adopted to accurately describe the overall power system security level. However, as the mathematical models presented in the work mentioned above are based on direct current power flow calculation, which normally ignores the influence of node voltage deviation. Actually, the assessment of the operation risk is inadequate for real power system. In addition, the actual active power demand is constantly fluctuating. Accordingly, generation control should be adjusted with changes of load level [16] in real time. Nevertheless, these existing studies focused only on single load level, which could not satisfy stricter requirement of practical operation.

To deal with the aforementioned issues, this paper introduces an advanced risk index considering the risk of both line overload and node voltage deviation under normal and fault conditions, which is based on nonlinear power flow calculations. The two objectives of risk-based economic dispatch (RBED), that is, fuel costs of generators and operation risk index, are both calculated in the presence of inner connections under different time scenarios during a day. As the fluctuation of load level is considered, 96 scenarios are uniformly selected in a day (24 h) to evaluate the risk based dispatch, with an interval of 15 min between two consecutive scenarios.

Generally, RBED is a complex mixed nonlinear programming problem. Conventional optimization algorithms, such as nonlinear programming [17], gradient decent method [18], interior point method [19], and the Newton method [20], may be easily trapped in a local optimum. Besides, an accurate system model and appropriate feasible initial solutions are needed to achieve a good application effect, based on which software (Gurobi [21] and CPLEX [22]) is not flexible enough and inapplicable for some complex problems. Hence, their application is relatively difficult and usually consumes a long period of time due to the large number of constrains under multiple operation conditions in RBED.

So far, an enormous variety of artificial intelligence (AI) algorithms, including genetic algorithm (GA) [23], quantum genetic algorithm (QGA) [24], artificial bee colony (ABC) [25], particle swarm optimization (PSO) [26] and bacteria foraging optimization (BFO) [27,28] have been successfully applied for an optimal power system operation due to their elegant merits of global convergence, model free feature, and applicability to discrete nonlinear problems. In particular, an optimization task can be tackled by variables, objective functions and the number of unsatisfied constraints. However, they usually tend to cost a long optimization period for each new task as no prior knowledge is exploited. Since there are 96 sub-tasks that need to be executed in RBED, it will consume plenty of time. It is assumed that either the scale of system is large or a large number of faults occur, so that the time limit of RBED is very difficult to meet.

Recently, transfer learning [29,30] has become a very powerful tool to accelerate the optimization based on machine learning. It is inspired from the fact that many practical engineering issues are related to historical ones which often share plenty of similar features in essence. Therefore, the optimization of a new task can be dramatically accelerated by appropriately exploiting the similarities from the experience (prior knowledge) of historical tasks (also called source tasks). Transfer learning has been widely applied in various problems, such as reactive power optimization [31], decentralized optimal carbon-energy combined-flow [32], cross-domain activity recognition [33], and pedestrian detection [34], etc. Q-learning algorithm, as one of the most widely used reinforcement learning, can be adopted for transfer learning. However, It merely employs a single agent to update the Q-value matrix, which leads to a relatively low convergence rate and sometimes even cause the curse of dimension in complex problems. Furthermore, a large number of iterations may be required due to the time-consuming trial-and-error mechanism of Q-learning.

In order to resolve the above disadvantages, this paper proposes a novel bacteria foraging reinforcement learning (BFRL) associated with knowledge transfer to handle RBED, which is developed from the BFO and Q-learning algorithm [35]. A Q-value matrix is chosen as the knowledge matrix. The learning mode of BFO is introduced in BFRL to achieve a multi-agent collaboration, which can considerably accelerate the knowledge matrix update and reduce the iteration number. Then, the

knowledge extension is employed to dramatically reduce the dimension of knowledge matrix, such that the curse of dimension can be effectively avoided. Through pre-learning, the knowledge matrices save the optimal prior knowledge from source tasks at first, on which the initial knowledge matrices of new tasks are developed thereafter. As a consequence, RBRD can be rapidly resolved by properly exploiting the similarity between source tasks and new tasks. Hence, BFRL is adequate to satisfy the fast calculation of RBED in practice, whose global convergence and the stability of new tasks can also be guaranteed through the knowledge transfer from source tasks. At last, BFRL is applied for RBED of 96 scenarios on RTS-79 system, which achieves better performance compared with that of some typical algorithms.

The following are the main motivations and innovations of this paper:

- The conventional economic dispatch usually just focuses on the fuel costs of generators. In contrast, the proposed RBED is implemented to obtain a proper trade-off between economical operation and system security, which can simultaneously reduce the fuel costs and the operation risk of power systems.
- Compared to the conventional method which merely considers the line overload in the SCOPF [9], the risks of both line overload and node voltage deviation are evaluated quantitatively based on the nonlinear power flow calculation by the proposed approach. In addition, it is resolved under various load scenarios thus being applicable to the load changes in practice.
- The conventional optimization algorithms might be easily trapped at a local optimum due to their dependence on an accurate system model and the feasible initial solutions. In contrast, no accurate system model is required by BFRL, such that it can be easily implemented for a much broader range of practical issues, e.g., nonlinear objective functions and different complex constraints.
- The knowledge learning of BFO and the trial-and-error of Q-learning can effectively cooperate in BFRL. Particularly, the knowledge matrix can significantly reduce the blindness of the random search via the cooperating bacteria. In turn, the update efficiency of knowledge matrix can be improved greatly via the multi-agent (i.e., the bacteria) collaboration. Besides, the dimension of knowledge matrix can also be reduced by knowledge extension. These merits accelerate the learning process hence being more feasible in practice.
- The existing AI algorithms are usually incapable of knowledge storage or knowledge transfer, which may easily lead to a high computation burden as significant iterations and population size are needed to obtain a high-quality optimal solution. This would be unable to satisfy the requirement of RBED period (less than 15 min). In contrast, BFRL employs the Q-value matrix as the knowledge matrix to save the optimal knowledge in pre-learning, and then the prior knowledge obtained from the similar source tasks can be fully exploited for the new tasks. Therefore the convergence of BFEL can be dramatically accelerated and cost less than 15 min for practical implementation.
- The simulation results verify the excellent performance of BFRL, especially on the convergence rate, which can reach 9 to 20 times faster than that of other AI algorithms, while a high-quality optimal solution and a high convergence stability can also be guaranteed.

2. Mathematical Model of RBED

2.1. Operation Risk Assessment

The operation risk assessment of power systems means a comprehensive evaluation with the possibility and severity of contingencies [36]. The risk index I_{R} can be calculated as follows:

$$I_{\mathrm{R}}(X_{\mathrm{f}}) = \sum_{i} P_{\mathrm{r}}(E_i) S_{\mathrm{ev}}(E_i) \tag{1}$$

The current condition X_f represents the current operating condition of a power system, which is associated with the operation risk Equations (2)–(8), thus it can be encoded with the output power of each generator P_G, each node voltage U_i, the power flow of each transmission line T_i, the load demand of each node P_{Di}, and the topology of power grid; E_i represents the ith contingency; $P_r(E_i)$ and $S_{ev}(E_i)$ are the probability and severity of E_i, respectively.

According to the statistical data, the failure rate of alternating current (AC) transmission line i at a certain time interval Δt follows the Poisson distribution, thus its cumulative failure rate $P_r(E_{Fi})$ can be described as:

$$P_r(E_{Fi}) = 1 - \exp(-\lambda_i \Delta t) \tag{2}$$

where E_{Fi} and λ_i denote the fault and failure rate of the ith transmission line, respectively.

Assuming there are m transmission lines in a power system with a single fault (the fault of the ith line) occuring at time t, the probability of this fault $P_r(E_{S\_Fi})$ is calculated by [37]:

$$P_r(E_{S\_Fi}) = P_r(E_{Fi}) \prod_{j \in \boldsymbol{S}_{UN}, j \neq i} (1 - P_r(E_{F\_j})) \tag{3}$$

where E_{S_Fi} represents a single line fault in system; $\boldsymbol{S}_{UN}$ is the set of all the normal operational transmission lines.

The outage of a transmission line may results in a sudden line overload or a severe node voltage deviation in a power system, whose effect can be usually described by a linear function. However, a linear risk index may not be capable of effectively distinguishing between a minor fault and a severe fault. As a consequence, a nonlinear utility function is employed so as to fully describe different faults.

The overload of the ith faulty line is defined as:

$$\omega_{Li} = \begin{cases} L_i - L_0, & L_i > L_0 \\ 0, & L_{\leq}^{i} L_0 \end{cases} \tag{4}$$

where ω_{Li} is the overload of the ith faulty line; L_i is the ratio of actual transmission power to transmission power constraint of the ith line; L_0 is the threshold and set to 0.9. Specifically, if L_i is less than L_0, the risk of the i-th line overload is set to zero.

The severity of the i-th line overload $S_{ev}(\omega_{Li})$ can be described by:

$$S_{ev}(\omega_{Li}) = \frac{\exp[a(\omega_{Li}) + b] - 1}{c} \tag{5}$$

where a, b and c are all positive constants. Meanwhile, the first-order derivative and second-order derivative of S_{ev} are also positive, which means the severity of the overload line increases monotonously.

Similarly, the voltage deviation of the ith node is defined by:

$$\omega_{Vi} = \begin{cases} U_{i\min} - U_i, \ 0 < U_i < U_{i\min} \\ 0, \ U_{i\min} \leq U_i \leq U_{i\max} \\ U_i - U_{i\max}, \ U_i > U_{i\max} \end{cases} \tag{6}$$

where ω_{Vi} represents the voltage deviation of the ith node ; U_i is the voltage amplitude of the ith node, while $U_{i\max}$ and $U_{i\min}$ are its upper and lower bounds, respectively.

The severity of the i-th node voltage deviation $S_{ev}(\omega_{Vi})$ is written by:

$$S_{ev}(\omega_{Vi}) = \frac{\exp[a(\omega_{Vi}) + b] - 1}{c} \tag{7}$$

Hence, the global operation risk index I_R of power system is developed by combining the risk of line overload and the risk of node voltage deviation, which yields:

$$I_R = \mu_1 I_{\mathrm{R\,L}} + \mu_2 I_{\mathrm{R\,V}} \tag{8}$$

where I_{RL} and I_{RV} are the total risk index of line overload and node voltage deviation, respectively; Moreover, μ_1 and μ_2 are the corresponding weight coefficients, with $\mu_1 + \mu_2 = 1$.

2.2. Multi-Objective Risk Economic Dispatch

The aim of RBED is to considerably reduce the fuel costs of generators and the operation risk of power systems together with all the security constraints being satisfied. To simplify the problem, the RBED constraints are replaced by outer penalty functions. With this method, the likelihood of infeasibility can be minimized. In future study, barrier method will be employed to guarantee a feasible solution. Since the outer penalty function C_{V} can be optimized throughout the n-dimensional real space, an initial solution outside the feasible field is acceptable, which can effectively reduce the difficulty of finding a feasible initial solution. The economic objective and the security objective are integrated as a single objective function through the linear weight method, as follows:

$$\min f(\boldsymbol{x}) = \mu_3 F_{\mathrm{C}}(\boldsymbol{x})/z_1 + \mu_4 I_{\mathrm{R}}(\boldsymbol{x})/z_2 + MC_{\mathrm{V}}(\boldsymbol{x}) \tag{9}$$

Subject to [38]:

$$\begin{cases} P_{\mathrm{G}i} - P_{\mathrm{D}i} = U_i \sum\limits_{j\in i} U_j (G_{ij}\cos\theta_{ij} + B_{ij}\sin\theta_{ij}) \\ Q_{\mathrm{G}i} - Q_{\mathrm{D}i} = U_i \sum\limits_{j\in i} U_j (G_{ij}\sin\theta_{ij} - B_{ij}\cos\theta_{ij}) \\ P_{\mathrm{G}i\min} \le P_{\mathrm{G}i} \le P_{\mathrm{G}i\max},\ i \in \boldsymbol{S}_{\mathrm{G}} \\ Q_{\mathrm{G}i\min} \le Q_{\mathrm{G}i} \le Q_{\mathrm{G}i\max},\ i \in \boldsymbol{S}_{\mathrm{G}} \\ U_{i\min} \le U_i \le U_{i\max},\ i \in \boldsymbol{S}_{\mathrm{D}} \\ |T_i| \le T_{i\max},\ i \in \boldsymbol{S}_{\mathrm{L}} \end{cases} \tag{10}$$

where F_{C} is the fuel costs of generators; C_{V} represents the value of total constraint violations obtained under normal operation; M is the penalty factor; μ_3 and μ_4 are the weight coefficients, with $\mu_3 \in [0, 1]$, $\mu_4 \in [0, 1]$, and $\mu_3 + \mu_4 = 1$; state vector variable $\boldsymbol{x} = [\boldsymbol{U}, \boldsymbol{\theta}, \boldsymbol{P}_{\mathrm{G}}, \boldsymbol{Q}_{\mathrm{G}}, \boldsymbol{T}]^{\mathrm{T}}$ represents the node voltage amplitude, node voltage phase angle, active and reactive power of generator, the apparent power of line, respectively; z_1 and z_2 are the normalization references; $P_{\mathrm{D}i}$ and $Q_{\mathrm{D}i}$ are the active and reactive power of the ith load, respectively; θ_{ij} is the voltage phase angle difference between the the ith and the jth node, G_{ij} and B_{ij} are the conductance and susceptance of line i–j, respectively; $P_{\mathrm{G}i\max}$ and $P_{\mathrm{G}i\min}$ are the upper and lower bounds of the generator active power while $Q_{\mathrm{G}i\max}$ and $Q_{\mathrm{G}i\max}$ are the upper and lower bounds of the generator reactive power, respectively; $T_{i\max}$ denotes the power limit of the ith line; $\boldsymbol{S}_{\mathrm{G}}$, $\boldsymbol{S}_{\mathrm{D}}$ and $\boldsymbol{S}_{\mathrm{L}}$ are the set of generators, load buses, and lines, respectively.

The fuel costs can be chosen as:

$$F_{\mathrm{C}} = \sum_{i\in \boldsymbol{S}_{\mathrm{G}}} (\zeta_{0i} + \beta_i \zeta_{1i} + \gamma_i \zeta_{2i}^2) \tag{11}$$

where ζ_{0i}, ζ_{1i} and ζ_{2i} are the coefficients of fuel costs, respectively.

The value of total constraint violations can be defined as:

$$\begin{aligned} C_{\mathrm{V}} &= \sum_{j=1}^{N_{\mathrm{v}}} \max(0, g_j(\boldsymbol{x})) \\ &= \max(0, P_{\mathrm{Gs}} - P_{\mathrm{Gsmax}}, P_{\mathrm{Gsmin}} - P_{\mathrm{Gs}}) + \sum_{i\in \boldsymbol{S}_G} \max(0, Q_{\mathrm{G}i} - Q_{\mathrm{G}i\max}, Q_{\mathrm{G}i\min} - Q_{\mathrm{G}i}) \\ &\quad + \sum_{i\in \boldsymbol{S}_D} \max(0, U_i - U_{i\max}, U_{i\min} - U_i) + \sum_{i\in \boldsymbol{S}_L} \max(0, |T_i| - T_{i\max}) \end{aligned} \tag{12}$$

where P_{Gs} is the generator active power on the slack bus; N_v is the number of variables.

It can be found that the integrated objective function f is the linear sum of two objective function (i.e., fuel costs F_C and global operation risk index I_R) with the linear weights, and total constraint violations C_V with the penalty factor M. Hence, the quality of an obtained optimal solutions is determined by the integrated objective function f, instead of the fuel costs F_C or the global operation risk index I_R solely. In general, a smaller fuel costs F_C will not always lead to a smaller f due to an inevitable conflict between fuel costs F_C and the global operation risk index I_R. In other words, a smaller F_C may even results in a larger f.

3. BFRL with Knowledge Transfer

3.1. Standard BFO Algorithm

Standard BFO algorithm is inspired by the foraging behaviour of bacteria which normally has three typical modes: chemotactic mode, dispersal mode and reproduction mode [28].

Normally, the local searching of BFO is enhanced through the chemotactic mode, which can be described as:

$$\psi^i(j+1,k,l) = \psi^i(j,k,l) + C_k^i \frac{\Delta^i}{\sqrt{(\Delta^i)^T \Delta^i}} \tag{13}$$

where $\psi^i(j, k, l)$ represents the position of the ith bacteria during the lth dispersal, the kth reproduction and the jth chemotactic; C_k^i is the step of swimming of the ith bacterium at the kth iteration; and Δ is a unit vector in the direction of swimming, respectively.

Here the nonlinear decreasing inertia step C_k^i is introduced to replace the fixed step in standard BFO so as to balance the global and local search, which is written as:

$$C_k^i = C_{\text{start}}^i - (C_{\text{start}}^i - C_{\text{end}}^i)[\frac{2k}{Iter_{\max}} - (\frac{k}{Iter_{\max}})^2] \tag{14}$$

where C_{start}^i and C_{end}^i are the initial and the final steps, respectively; $Iter_{\max}$ is the maximum iteration number.

In the reproduction mode, the bacteria are ranked according to the energy intensity firstly. Millions of years of struggle in harsh environment has driven bacteria to gradually evolve an optimal survival pattern for the overall benefits of the whole species: the superiors (those have the highest energy intensity) are eligible to freely and rapidly reproduce while the inferiors (those have the lowest energy intensity) are forced to inevitably die out. Assuming the number of employed bacteria in standard BFO to be N_P, the number of bacteria to be eliminated is $N_P/2$. Then the ones with the energy intensity ranking the second half of all bacteria are replaced by the other half bacteria.

In this paper, the reproduction mode is improved by introducing a crossover to spread the diversity of bacteria. The new bacteria to replace the eliminated ones are generated as:

$$\psi^{i+N_P/2}(j,k,l) = r_1 \psi^i(j,k,l) + (1-r_1)\psi^{i+N_P/2}(j,k,l) \tag{15}$$

where $i \in [1, N_P/2]$; and $r_1 \in [0, 1]$ is a random number, respectively.

The global convergence is improved via the dispersal mode, which occurs at a given probability P_{r_ed}. When the dispersal probability is satisfied, the positions of the bacteria will change randomly.

3.2. Knowledge Matrix

Q-learning is one of the most famous and widely used reinforcement learning techniques, which contains three key elements including state s, action a, and reward R. The state-action value function $\boldsymbol{Q}$ is the knowledge matrix of all the state-action pairs, i.e., $\boldsymbol{Q}(s, a)$. The agent of Q-learning can update the knowledge matrix by feedback reward R from taking an action a to the environment in the current

state s. Each element of the matrix represents the knowledge of the corresponding state-action pair, which is used to estimate the discounted sum of future rewards started from the current state and action policy. Note that the text 'Q' represents the name of Q-learning while the symbol '$\boldsymbol{Q}$' denotes the knowledge matrix.

Since the knowledge matrix saves the optimization policy, it can be treated as the brain of the agent. The knowledge matrix in Q-learning algorithm is updated by a single agent through the trial-and-error. The agent tries an action a and obtains a reward R from the environment in state s. Thus the corresponding knowledge value of the state-action pair $\boldsymbol{Q}(s, a)$ can be updated. Then in a certain state s, the agent will prefer to choose the action related to a large knowledge element $\boldsymbol{Q}(s, a)$. Hence, the knowledge matrix will gradually converge.

To accelerate the update of the knowledge matrix, the bacteria are employed as multiple agents of BFRL. The bacteria (i.e., agents) in different modes can change their positions (i.e., select actions a) according to Equations (13)–(15), (18) and (19) and acquire the energy (i.e., get rewards R) from the solution space (i.e., the environment).

As shown in Figure 1, a bacterium can obtain an action policy under a given state from the knowledge matrix and update its prior knowledge by the feedback of reward, which helps to boost the accumulative energy intensity of bacteria during foraging. The bacteria can obtain higher energy intensity from the red area of the figure.

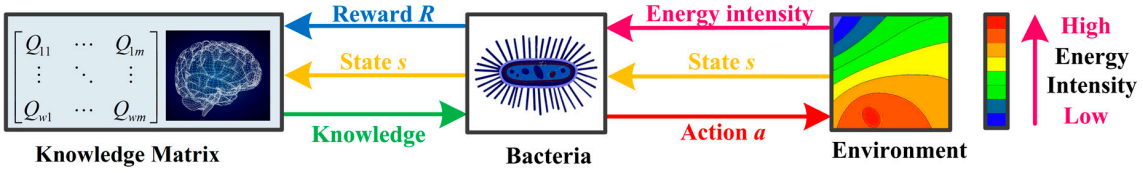

Figure 1. Interaction between a bacteria's knowledge matrix and the environment.

Basically, the knowledge matrix of Q-learning is a lookup table with the size of $|\boldsymbol{S}_{\text{pa}}| \times |\boldsymbol{A}|$, where $\boldsymbol{S}_{\text{pa}}$ is the state space and $\boldsymbol{A}$ is the action space. If Q-learning is used for solving RBED, the actions, namely the value of controlled variables, are independent from each other [38]. Assuming there are k variables and N_i available actions in each space, then the size of action space $|\boldsymbol{A}|$ is calculated by $N_1 \times N_2 \times \ldots \times N_{k-1} \times N_k$. It is obvious that the curse of dimension may be emerged if the action space is too large.

As illustrated in Figure 2, BFRL employs a knowledge extension in order to considerably reduce the dimension of the original knowledge matrix $\boldsymbol{Q}$. $\boldsymbol{Q}$ is divided into several knowledge sub-matrices $\boldsymbol{Q}^i$, which are one-to-one correspondence with the variables. Furthermore, the elements of neighboring sub-matrices are defined as related knowledge, which means the action space of $\boldsymbol{Q}^i$, i.e., the range of the ith variable, is the same as the state space of $\boldsymbol{Q}^{i+1}$. In other words, the value of the $(I + 1)$th controlled variable cannot be selected until the ith variable has been determined. Note that the original high-dimension knowledge matrix is decomposed into multiple low-dimension sub-matrices through extension chains between related knowledge.

The Knowledge matrix is merely updated by a single agent in Q-learning. As a result, only one element can be updated in each cycle, which leads to a relatively slow convergence. In contrast, the multi-agent collaboration is adopted in BFRL, where the bacteria share the same knowledge sub-matrices. Consequently, multiple elements can be updated in a single iteration, which would significantly accelerate the learning rate. The knowledge sub-matrix $\boldsymbol{Q}^i$ is updated as follows [39]:

$$\rho_k^{ij} = R(s_k^{ij}, s_{k+1}^{ij}, a_k^{ij}) + \gamma \max_{a^i \in A^i} \boldsymbol{Q}_k^i(s_{k+1}^{ij}, a) - \boldsymbol{Q}_k^i(s_k^{ij}, a_k^{ij}) \tag{16}$$

$$\boldsymbol{Q}_{k+1}^i(s_k^{ij}, a_k^{ij}) = \boldsymbol{Q}_k^i(s_k^{ij}, a_k) + \alpha \rho_k^{ij} \tag{17}$$

where i denotes the ith knowledge sub-matrix and j denotes the jth bacteria; ρ_k^{ij} is the update component of $\boldsymbol{Q}^i$; $R(s_k^{ij}, s_{k+1}^{ij}, a_k^{ij})$ is the reward of a transition from state s_k^{ij} to state s_{k+1}^{ij} under a selected action a_k^{ij} in the kth iteration; α and γ are the learning factor and discount factor, respectively.

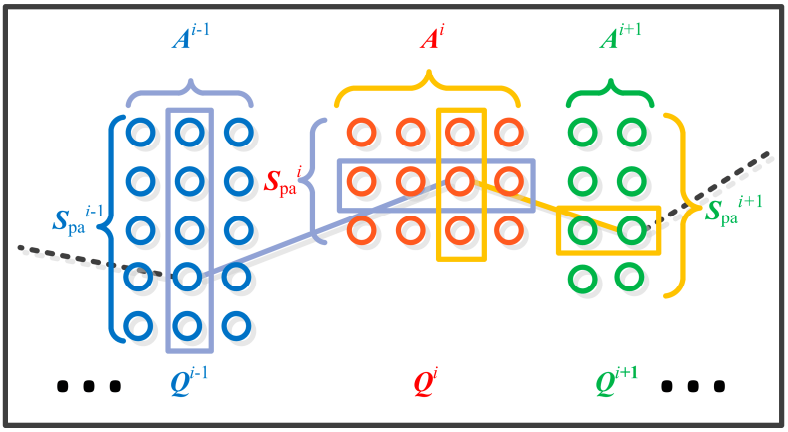

Figure 2. Dimension reduction through knowledge extension.

3.3. Knowledge Learning

The search pattern of the BFO is completely random, which usually leads to a blindness and inefficiency of problem solving. Different from the random exploration of BFO, BFRL can search the solution space according to the knowledge matrix and update the knowledge matrix using the received reward, such that a more informative and meaningful exploration can be realized.

As illustrated in Figure 3, there are bacteria in either chemotactic mode or dispersal mode at the beginning of each iteration. In a given iteration, the mode of each bacterium is assigned in a certain percentage. Then the learning of bacteria in two modes is conducted in different ways, while each bacterium receives a reward and updates the knowledge matrix accordingly. Furthermore, all the bacteria move to the reproduction mode, which means the end of each iteration. As described in Section 3.1, the bacteria are either reproduced or died out according to their obtained reward ranking.

In the next iteration, the modes of bacteria are reassigned. The bacteria with higher reward are assigned to the chemotactic mode while others are assigned to the dispersal mode.

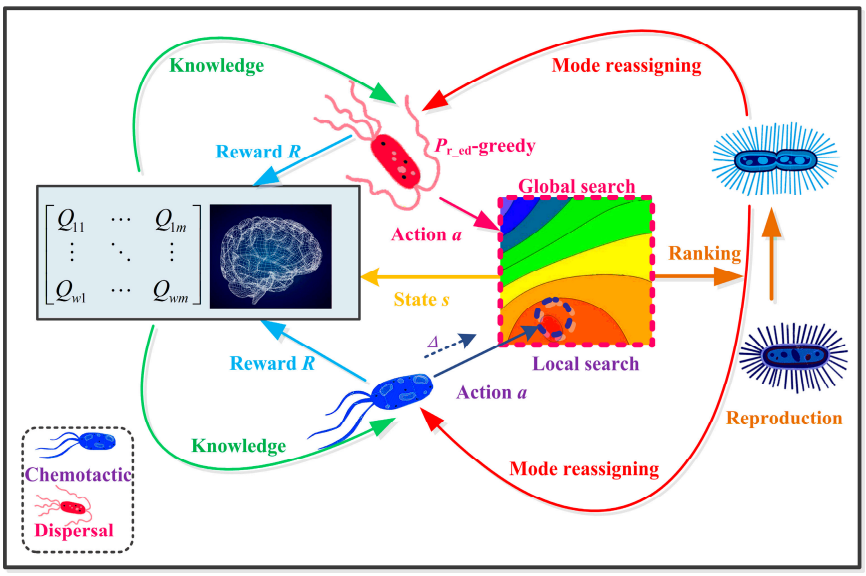

Figure 3. Knowledge learning of BFRL associated with chemotactic and dispersal.

In BFRL, the knowledge learning of bacteria in dispersal mode is guided by the knowledge matrix, which is different from that of standard BFO. For a given state, a larger knowledge element means a higher reward value obtained under the corresponding action. In other words, the information belonging to superiors has been saved with the update of the knowledge matrix. Furthermore, a roulette wheel selection is used based on the state-action probability matrix $\boldsymbol{O}^i$ when the dispersal probability $P_{\text{r\_ed}}$ is satisfied. Otherwise, the action with the largest knowledge element $\underset{a^i \in A_i}{\arg\max} \boldsymbol{Q}^i_{k+1}(s^{ij}_{k+1}, a^i)$ is selected. For a controlled variable, an action of each bacterium is selected as follows:

$$a^{ij}_{k+1} = \begin{cases} \underset{a^i \in A_i}{\arg\max} \boldsymbol{Q}^i_{k+1}(s^{ij}_{k+1}, a^i), & r_2 \geq P_{\text{r\_ed}} \\ a_S, & \text{otherwise} \end{cases} \tag{18}$$

where $r_2 \in [0, 1]$ is a random number; a_s denotes a random global action determined by the distribution of state-action probability matrix $\boldsymbol{O}^i$, which is updated by:

$$\begin{cases} \boldsymbol{e}^i(s^i, a^i) = \dfrac{1}{\boldsymbol{Q}^i(s^i, a^i) - \beta \max\limits_{a' \in A^i} \boldsymbol{Q}^i(s^i, a')} \\ \boldsymbol{O}^i(s^i, a^i) = \dfrac{\boldsymbol{e}^i(s^i, a^i)}{\sum\limits_{a' \in A^i} \boldsymbol{e}^i(s^i, a')} \end{cases} \tag{19}$$

where β is the divergence factor to magnify the divergence of $\boldsymbol{Q}^i$ and $\boldsymbol{e}^i$ is the introduced transition matrix in the calculation.

3.4. Knowledge Transfer

Assuming there are multiple similar tasks to complete for BFRL, the efficiency of new tasks can be improved greatly via knowledge transfer.

As shown in Figure 4, knowledge transfer can accelerate the learning of new tasks based on the existing ones. If the state space and the action space remain constant, the optimal knowledge matrices of the source tasks can be treated as the initial knowledge matrices of the target tasks, which are called the prior knowledge [31]. The source tasks need to be executed during the pre-learning to obtain the optimal knowledge matrices, from which the prior knowledge is exploited for the relevant new tasks. Then the initial knowledge matrices of source tasks $\boldsymbol{Q}_{\text{S}}$ will be transferred to the prior knowledge matrices of new tasks $\boldsymbol{Q}_{\text{N}}$ in transfer learning.

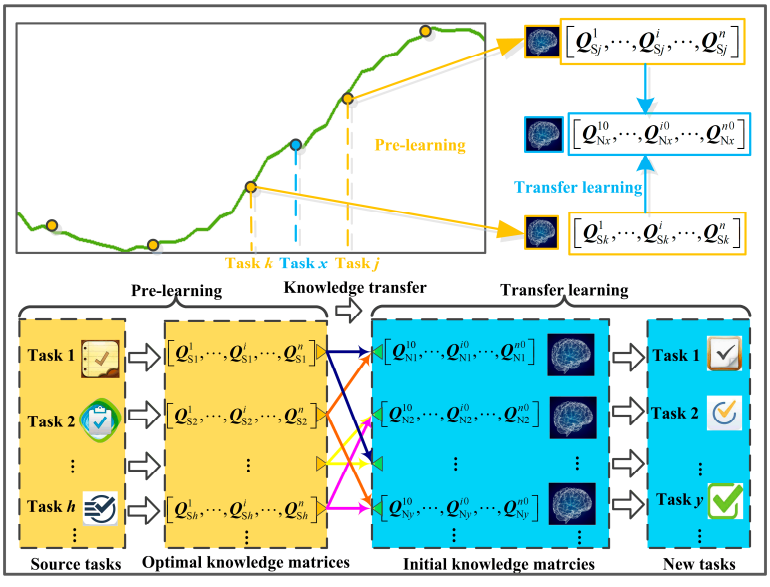

Figure 4. The procedure of knowledge transfer.

Here, the similarities among different tasks are contained in the prior knowledge matrices, together with some unrelated ones. As a result, a malignant negative transfer may sometimes emerge. To handle this, the extraction of closely relevant knowledge and the identification of similarities among different tasks are emphasized during the transfer learning of BFRL.

3.5. Convergence Characteristics

Firstly, it is important to note that the conventional Q-learning can converge to the optimal Q-value matrix $\boldsymbol{Q}^*$ as all the actions are sufficiently explored in each state space, while the global optimum can be determined by the optimal Q-value matrix $\boldsymbol{Q}^*$, in which the detailed proof can be found in [40]. Moreover, the learning mode of BFRL is the same as that of Q-learning, while two main improvements of BFRL compared with Q-learning can be summarized as: (1) knowledge transfer and (2) exploration and exploitation based on bacteria foraging mechanism. Specifically, the first one only changes the initial Q-value matrix, thus it can approximate the optimal Q-value matrix for a current optimization task. Besides, the second one only accelerates the update efficiency of Q-value matrix. Therefore, BFRL will only accelerate the convergence compared with Q-learning, while the convergence can be completely guaranteed as all the actions are sufficiently explored in each state space.

4. BFRL with Transfer Learning for RBED

4.1. BFRL Structure

The RBED is different from the conventional AC optimal power flow. To obtain the objective function, the risk index of a power system should be calculated at first, so the AC power flow calculations under normal condition and all the fault conditions need to be executed by AI algorithms. When N_f faults are included in the contingency, the number of power flow calculation in RBED will be $(N_\mathrm{f} + 1)$ times higher than that of the conventional AC optimal power flow. Therefore, RBED requires much longer time. Assuming there are N_ed dispersals, N_re reproductions and N_c chemotactic in BFO for RBED, as well as a maximum swimming number N_s, the total times of power flow calculation becomes $N_\mathrm{ed} \times N_\mathrm{re} \times N_\mathrm{c} \times N_\mathrm{s} \times (N_\mathrm{f} + 1)$; this leads to an extremely slow calculation. In contrast, the optimization efficiency can be dramatically improved due to the removal of nested cycles in BFRL.

4.2. Design of State and Action

The generator active power on PV nodes is selected as the controlled variable. The action space $\boldsymbol{A}(\boldsymbol{A}^{\mathrm{PG1}}, \boldsymbol{A}^{\mathrm{PG2}}, \ldots, \boldsymbol{A}^{\mathrm{PG}Nq})$ is consistent with the controlled variable space, namely the positions of the bacteria, where N_q is the number of controlled variables. Besides, the action space of the former one is the state space of the latter one. The knowledge sub-matrices corresponding to the state-action pair of the variables are denoted as $\boldsymbol{Q}^{\mathrm{PG1}}, \boldsymbol{Q}^{\mathrm{PG2}}, \ldots, \boldsymbol{Q}^{\mathrm{PG}Nq}$, respectively.

4.3. Design of Reward Function

The reward function R is designed as:

$$R = \frac{1}{\omega_1(F_\mathrm{C}/z_1) + \omega_2(I_\mathrm{R}/z_2) + MC_\mathrm{V}} \tag{20}$$

where it shows that the fuel costs of different scenarios range from \$21,135 to \$39,402 while the risk index varies between 2.42×10^{-5} p.u.~5.2×10^{-5} p.u. by trial-and-error, so z_1 and z_2 are set to 10^4 (\$) and 10^{-5}, respectively. Additionally, $(F_\mathrm{C}/z_1) \in [2.1, 3.9]$ and $(I_R/z_2) \in [2.4, 5.2]$ are the intervals. In general, the penalty factor M should be appropriately chosen: If it is too small, the minimal point of the penalty function is apart from the optimal solution and results in a low efficiency; if it is too large, the penalty function minimization would be very slow [41]. Since C_V is large enough compared to that of the normalized fuel costs and risk index, M is chosen to be 1.

4.4. Knowledge Transfer

The core task of learning efficiency improvement is to extract the similarities between the source tasks and the new tasks. The optimization of RBED is mainly determined by the power flow of power systems. In practice, it is closely dependent on the active power demand as the topology and the operation conditions are relatively steady in a short time. Thus, the active power deviation is defined as the similarity between the source tasks and the new tasks. The active power demand is divided into multiple load intervals as follows:

$$[P_{\mathrm{Ds}1}, P_{\mathrm{Ds}2}), [P_{\mathrm{Ds}2}, P_{\mathrm{Ds}3}), \ldots, [P_{\mathrm{Ds}i-1}, P_{\mathrm{Ds}i}), \ldots, [P_{\mathrm{Ds}n-1}, P_{\mathrm{Ds}n}) \tag{21}$$

where $[P_{\mathrm{Ds}i\text{-}1}, P_{\mathrm{Ds}i})$ is a half-open load interval; $P_{\mathrm{Ds}i}$ represents the power demand of the ith load intervals in the source task, with $P_{\mathrm{Ds}1} < P_{\mathrm{Ds}2} < P_{\mathrm{Ds}i} < P_{\mathrm{Ds}n\text{-}1} < P_{\mathrm{Ds}n}$. Moreover, the closely related knowledge of source tasks should be exploited in priority for a new task in order to enhance the transfer learning effectiveness.

Assuming the power demand of a new task x is represented by $P_{\mathrm{D}x}$, with $P_{\mathrm{D}i} < P_{\mathrm{D}x} < P_{\mathrm{D}k}$, the similarities between the new task and two source tasks can be calculated as:

$$\begin{cases} \eta_1 = \frac{P_{\mathrm{D}x} - P_{\mathrm{D}j}}{P_{\mathrm{D}k} - P_{\mathrm{D}j}} \\ \eta_2 = \frac{P_{\mathrm{D}k} - P_{\mathrm{D}x}}{P_{\mathrm{D}k} - P_{\mathrm{D}j}} \end{cases} \tag{22}$$

where η_1 and η_2 are the similarities coefficients, with $\eta_1 + \eta_2 = 1$.

The knowledge matrix of the new task x can be obtained by a linear transfer, which yields:

$$\boldsymbol{Q}_x^i = \eta_1 \boldsymbol{Q}_j^i + \eta_2 \boldsymbol{Q}_k^i \tag{23}$$

where $\boldsymbol{Q}_x^i$, $\boldsymbol{Q}_j^i$ and $\boldsymbol{Q}_k^i$ denote the knowledge sub-matrices of the ith variable in source task x, source task j and new task k, respectively.

The overall knowledge transfer can be summarized as follows:

Step 1 Select several scenarios as the source tasks from the daily load curve at a fixed time interval.
Step 2 Execute the pre-learning and save the knowledge in the knowledge matrices of the source tasks.
Step 3 Calculate the similarities between the new tasks and the closest source tasks based on the active power deviation.
Step 4 Obtain the initial knowledge matrix of the source tasks.

4.5. Execution Procedure of BFRL for RBED

The execution procedure of BFRL for RBED is shown in Figure 5.

4.6. Parameters Setting

In BFRL, the crucial parameters include the population size N_{P}, dispersal probability $P_{\mathrm{r\_ed}}$, learning factor α, discount factor γ and $Iter_{\max}$ [42]. Basically, these parameters should be carefully set by the following guidelines:

- A larger population size N_{P} may increase the probability of approaching the global optimum with longer time, here $N_{\mathrm{P}} \geq 1$.
- The dispersal probability $P_{\mathrm{r\_ed}}$ determines the trade-off between exploration and exploitation. A larger $P_{\mathrm{r\_ed}}$ means the roulette wheel selection is preferred, with $0 < P_{\mathrm{r\_ed}} < 1$.
- The learning factor α influences the learning rate. A larger α tends to accelerate the learning rate while the algorithm may however reach a pre-convergence.

- The discount factor γ discounts the future rewards of the knowledge matrix. A smaller discount factor γ means the current reward is more important.
- $Iter_{\max}$ is the maximum number of the iterations, which determines the quality of optimal solutions and the calculation time. In this paper, $Iter_{\max}$ is selected from some given values such as 50, 100, 150, 200 and 250. $Iter_{\max}$ is designed to balance the quality of optimal solutions and the calculation time via trial-and-error. Generally, a larger $Iter_{\max}$ will result in a higher quality of optimal solutions while it will consume more time. According to the result of trial-and-error, it can be found that the objective function obtained by BFRL can achieve a stable minimum value or fluctuate in a very small range when the number of iterations is larger than 150. So the $Iter_{\max}$ is set to 150 as it is large enough to ensure stable optimal solutions and shorten the calculation time.

Through extensive trial-and-error, the optimal parameters are listed in Table 1.

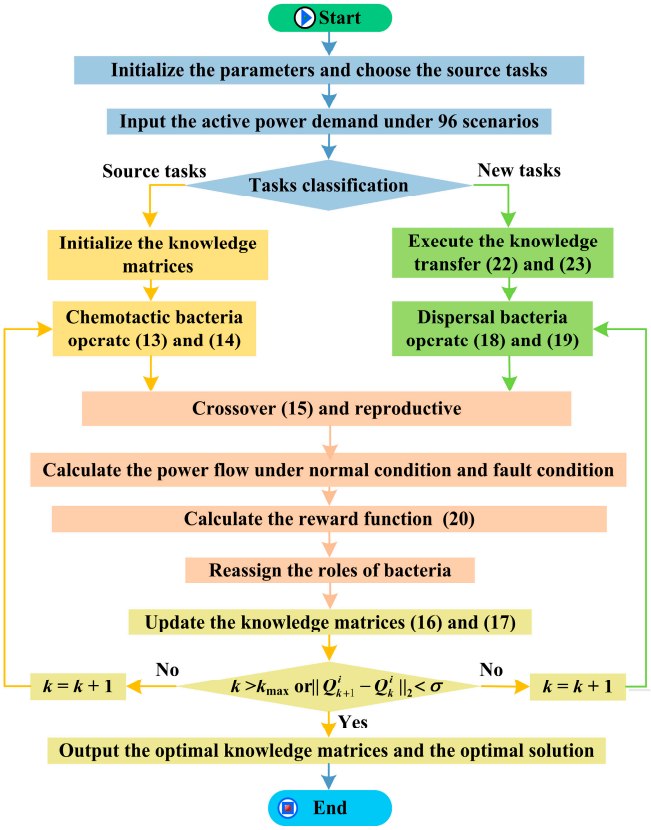

Figure 5. The flow chart of RBRD.

Table 1. Optimal BFRL parameters obtained through trial-and-error.

Parameters	Range	Learning Procedure	
		Source Task	New Task
N_P	$N_P > 0$	200	64
P_{r_ed}	$0 < P_{r\_ed} < 1$	0.5	0.02
α	$0 < \alpha < 1$	0.1	0.99
γ	$0 < \gamma < 1$	0.9	0.99
$Iter_{\max}$	50, 100, 150, 200, 250	150	50

5. Case Studies

The simulation is undertaken on an AMAX server with an Intel Xeon E5-2670 CPU at 2.3 GHz with 64 GB of RAM. The power flow calculation is based on the Matpower 6.0 toolbox in MATLAB R2014a.

The performance of BFRL for RBED has been evaluated on IEEE RTS-79 system [43] compared with that of other algorithms, e.g., GA [23], QGA [24], ABC [25], PSO [26], BFO [27,28] and Q-learning [40]. For each algorithm, there are both feasible and infeasible solutions to the proposed RBED problem. If an algorithm finds infeasible solutions, the power flow calculation may not converge or the controlled variables may violate the constraints. Then C_V is greater than 0 and its fitness function become larger than that of others, so that an individual is forced to find another solution and the previous infeasible solution can be eliminated. If an algorithm finds feasible solutions, then C_V becomes 0 and the fitness function is smaller, which leads others to approach it. When all the algorithms complete the default maximum number of iteration, the optimal one can guarantee the smallest fitness function as well as the convergence of the power flow with all the operation constraints being satisfied. Therefore, for each AI algorithm, their final convergence will be a feasible solution as long as the population size and the maximum number of iterations are set to be sufficiently large.

In this paper, the main parameters of each algorithm have been determined via trial-and-error. Therefore, the simulation results obtained by these algorithms can achieved a proper trade-off between the quality of optimal solutions and the calculation time. To shorten the execution time of fitting the parameters effectively, the uniform design is adopted [44]. For example, there are four crucial parameters which may have a great influence on the performance of GA in different optimization tasks. Assume that the value of each parameter is divided into 10 discrete levels, e.g., the mutation probability can be quantized into 10 discrete levels as [0.05, 0.1, . . . , 0.5], then $10 \times 10 \times 10 \times 10 = 10^4$ experiments should be executed to fit all the parameters, which will result in an extremely high computational burden. In contrast, only 10 experiments are needed via the uniform design. The main parameters of other algorithms have been listed in Table 2.

Table 2. The main parameters setting of each algorithm.

Algorithm	Parameter	Value
Q-learning	Learning factor	0.8
	Exploration weighting factor	0.9
	Discount factor	0.1
GA	Population size	200
	Generations	100
	Mutation probability	0.1
	Crossover probability	0.8
PSO	Population size	200
	Maximum generations	150
	Weight coefficients c_1/c_2	0.75/0.75
	Minimum velocity	-5
	Maximum velocity	5
	Inertia factors $\omega_{start}/\omega_{end}$	0.9/0.4
QGA	Population size	150
	Maximum generations	150
	Rotation angle	0.1π
BFO	Population size	60
	Number of chemotactic steps	20
	Limit of the length of a swim	6
	The number of reproduction steps	4
	The number of dispersal events	2
	Dispersal probability	0.2
ABC	Population size	200
	Colony size	50
	Employed bees	30
	Onlookers	30
	Scouts	20
	Limit	6

5.1. Simulation Scheme

The detailed simulation scheme can be illustrated in Table 3.

Table 3. The detailed simulation scheme of the proposed technique.

Number of Step	Detailed Simulation Scheme
Step 1:	Calculate the fault probability of each line in RTS-79 system according to Equations (1)–(3). Then five '$N-1$' line faults and two '$N-2$' line faults are selected as the contingencies.
Step 2:	Choose a typical load curve and divide it into 96 optimization tasks.
Step 3:	Determine the source tasks via trail-and-error, which are usually chosen to be as small as possible to ensure the effectiveness of knowledge transfer.
Step 4:	Select the output active power of generators as the controlled variable, and then determine the action space A of BFRL.
Step 5:	Determine the parameters used in the pre-learning of BFRL via trail-and-error and evaluate the pre-learning, then the optimal knowledge matrices obtained under the selected 21 source tasks will be saved.
Step 6:	Develop the initial knowledge matrices of the new tasks from the prior optimal knowledge matrices according to Equation (23) and select the optimal parameters.
Step 7:	Choose the optimal parameters of other AI algorithms for RBED via trial-and-error.
Step 8:	Implement each algorithm for RBED in 10 runs with 96 new tasks to compare their performance, including computation time, convergence time, quality of obtained optimal solution, and convergence stability.
Step 9:	Analyse and conclude the simulation results.

5.2. Simulation Model

The IEEE RTS-79 is a typical benchmark with a base capacity of 100 MVA, including 24 buses, 34 transmission lines/transformers and 32 generators, the configuration of which is illustrated in Figure 6 [45]. Here, bus 21 is chosen as the slack bus as it has the largest capacity. Furthermore, the generator active power of other buses is chosen as the controlled variables. The fuel costs coefficients of RTS-79 system can be found in [46].

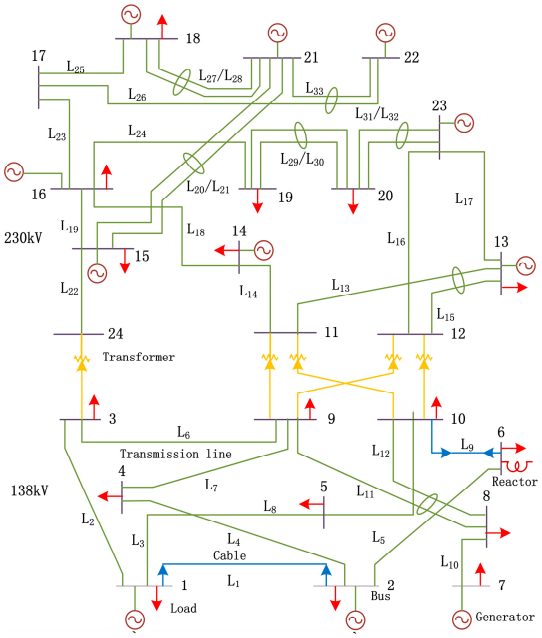

Figure 6. Configuration of IEEE RTS-79 system.

The daily load curve almost represents the trend of each day in a period of time (e.g., a month or a season). Based on this typical daily load curve, the operators will make an optimal operating schedule of the power system. And the typical daily load curve in Figure 7 is modelled from an actual province grid of southern China. As illustrated by Figure 7, a typical daily load curve can be divided into 96 scenarios with 15 min for each. In order to evaluate the adaptability of BFRL under different load levels, several case studies are carried out in all scenarios, which lead to 96 tasks.

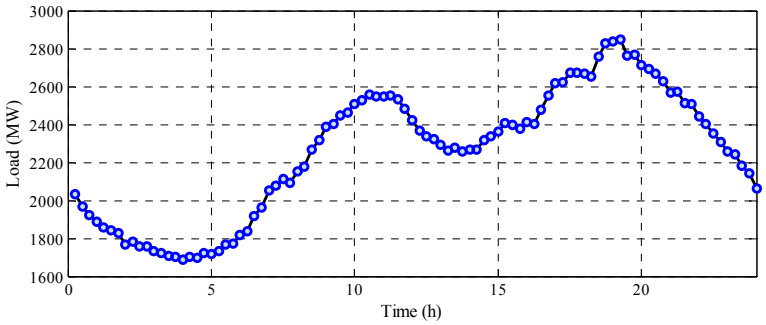

Figure 7. A typical daily load curve.

The contingencies are listed in Table 4, which includes five '$N-1$' transmission line faults and two '$N-2$' transmission line faults most likely to occur with related outage mode. The probability of each fault can be calculated according to Equation (3).

Table 4. The contingencies used in the studied power system.

Fault Type	Fault Line Number	Probability of Fault
'$N-1$' fault	2	1.4549×10^{-5}
	5	1.3693×10^{-5}
	16	1.4834×10^{-5}
	17	1.3978×10^{-5}
	26	1.5405×10^{-5}
'$N-2$' fault	13, 15	1.3026×10^{-10}
	29, 30	1.1756×10^{-10}

5.3. The Pre-Learning

Before the online learning of BFRL, several appropriate scenarios need to be chosen as source tasks in the pre-learning, on which the initial knowledge matrices of new tasks can be based. Figure 7 demonstrates that the active power demand in 96 scenarios is distributed between 1685 MW to 2850 MW, while 21 scenarios are sampled with the same capacity of 55 MW, ranked from low to high as 16, 21, 24, 5, 26, 2, 1, 31, 32, 94, 56, 51, 36, 88, 40, 42, 68, 70, 80, 79 and 77, respectively.

In addition, the convergence of BFRL obtained under scenario 1 is presented in Figure 8, which is compared with that of BFO. In around 270 s, BFRL can almost find the minimal fitness function. In contrast, BFO needs about 768.8 s. The convergence of BFRL is nearly 2.8 times faster than that of BFO with a better optimal solution. Moreover, it needs to claim that the searching efficiency of BFRL is not important in the pre-learning process, thus a large population size and a huge number of iterations are adopted to ensure its global convergence. However, the searching rate of BFO is still slower than that of BFRL due to the nested cycles. Besides, the random search in BFO is relatively blind. To handle this obstacle, the P_{r_ed}-greedy rule and multi-mode exploration are integrated into BFRL. As a result, the optimal objective function of BFRL is 33% smaller than that of BFO.

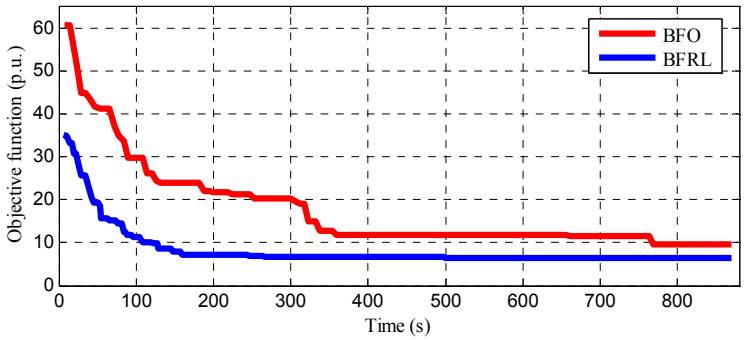

Figure 8. Offline optimization process of scenario 1.

5.4. Transfer Learning

The optimal action policies of source tasks are obtained through pre-learning and saved in the knowledge matrices, which will be transferred to be the initial knowledge matrices of new tasks according to their similarities. For example, the power demand of scenario 4 is 1887 MW while scenario 5 and 26 are the two closest, whose power demands are 1858 MW and 1916 MW, respectively. Then the initial knowledge matrix of scenario 4 can be developed from the linear weighed sum of the optimal matrices of scenario 5 and 26.

The convergence time of each algorithm of the 4th new task are given in Figure 9 and Table 5. In Tables 5–7, the best convergence results of all the algorithms are bolded. Note that the convergence time of BFRL is only 46 s thanks to the knowledge transfer, which is about 5.6 to 10% of that of other algorithms. Furthermore, compared to the convergence time in pre-learning, the rate of BFRL is increased by nearly 10 times, which verifies the efficiency of transfer learning. Since the time period of RBED for each scenario optimization is about 15 min, even if more faults are considered, the BFRL are still fast enough to meet such time limits. Moreover, the reinforcement learning needs to undergo the whole Markov process before convergence. As illustrated in Figure 10, the fuel costs of generators grow with the power demand as the generators should increase the output to balance such load increases. On the other hand, the line overload and node voltage deviation are more severe with a higher load level. Compared with Figure 7, the variations of fuel costs and risk index are consistent with the daily load curve. This demonstrates that the prior knowledge is effectively exploited.

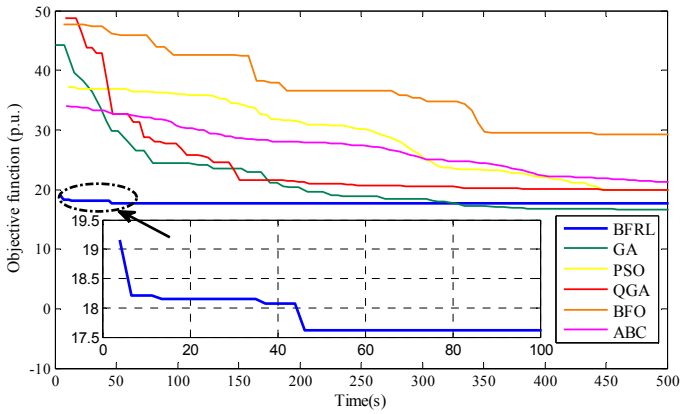

Figure 9. Online optimization process obtained by each algorithm under scenario 4.

Table 5. Convergence time of each algorithm under scenario 4.

Algorithm	GA	PSO	QGA	BFO	ABC	BFRL
Time (s)	464.31	451.52	541.43	820.50	672.58	**46.32**

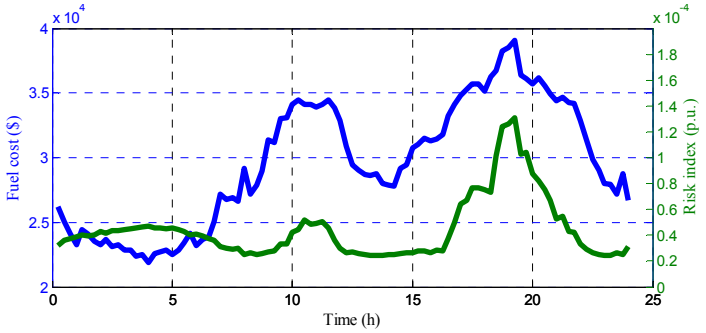

Figure 10. Daily optimization curve of BFRL.

The daily optimal objective function of RTS-79 system obtained by each algorithm is illustrated by Figure 11. The curve of objective function by BFRL is just slightly higher than that of GA while lower than that of other algorithms, which verifies the superior global convergence ability of BFRL.

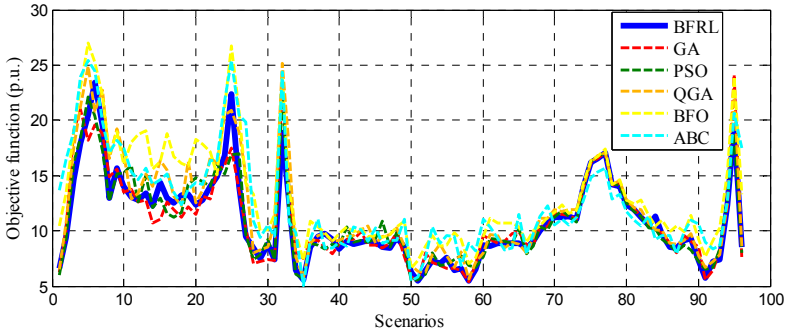

Figure 11. Convergence results of objective function of 96 scenarios obtained by each algorithm.

In general, AI algorithms are random and uncertain in finding an optimal solution, i.e., the obtained optimal solution may vary in different runs. To further compare the optimization performance, each AI algorithm is implemented in 10 runs. In each run, the optimization processes are evaluated under 96 scenarios. So for each algorithm, the total number of runs is equal to 10 times × 96 scenarios = 960, which is considered to be proper to evaluate the convergence stability of each algorithm [32]. Furthermore, the significance of our simulation results has been proven for performance comparisons, including calculation time, convergence time, quality of obtained optimal solution, and the distribution statistical results of obtained objective function (i.e., the convergence stability). In Table 6, the calculation time of each algorithm is the total execution time to solve 96 new tasks while the convergence time is the average optimization time of a single load scenario, which can clearly describe the optimization efficiency of the algorithm. The fuel costs of generators, risk index, and the optimal objective function are the sum of 96 new tasks, which are the statistical data on the obtained optimal solutions. Note that the quality of an obtained optimal solution is only determined by the integrated objective function, instead of the fuel costs or the global operation risk index. Although QGA achieves a less fuel costs F_C than that of BFRL, QGA has a larger integrated objective function f due to a much larger global operation risk index I_R than that of BFRL. Hence, BFRL outperforms QGA as it obtains a higher quality optimal solution with a smaller f, which is shown obviously in Figure 12.

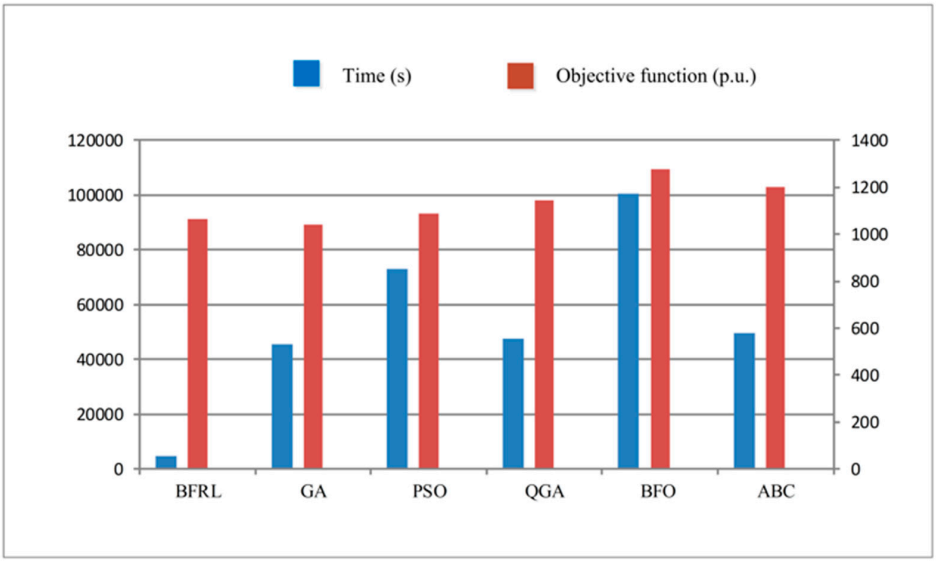

Figure 12. The histogram of optimization results obtained by different algorithms.

Table 6. Average optimization results of 96 scenarios obtained by each algorithm in 10 runs.

Algorithm	Calculation Time (s)	Convergence Time (s)	Fuel Costs ($)	Risk Index (p.u.)	Objective Function (p.u.)
Q-learning	The algorithm fails to converge due to curse of dimensionality.				
GA	45,590.41	474.90	**2,832,148.91**	4.346×10^{-3}	**1041.36**
PSO	73,122.14	761.69	2,865,308.35	$\mathbf{4.273 \times 10^{-3}}$	1086.82
QGA	47,462.28	494.40	2,835,208.40	4.338×10^{-3}	1147.77
BFO	100,409.9	1045.93	2,877,187.63	4.283×10^{-3}	1277.52
ABC	49,824.74	519.01	2,864,487.85	4.305×10^{-3}	1200.89
BFRL	**4904.30**	**51.08**	2,839,588.33	4.279×10^{-3}	1066.52

Table 7 gives the statistical data of the convergence stability of each algorithm obtained in 10 runs. The data in the first two columns of the table are the worst and the best objective functions obtained by each algorithm in 10 runs. Variance is the expectation of the squared deviation of objective functions from their mean value, which measures how far the results in 10 runs are spread out from the mean value. Furthermore, standard deviation is the arithmetic square root of the variance. The ratio of the standard deviation to the mean value of objective functions is the relative standard deviation (RSD), which is used to indicate the precision of the simulation results.

Table 7. Convergence performances of objective function of 96 scenarios obtained by each algorithm in 10 runs.

Algorithm	Worst	Best	Variance	Standard Deviation	RSD
Q-learning	The algorithm fails to converge due to curse of dimensionality.				
GA	**1027.24**	**1052.51**	70.00	8.36	8.031×10^{-3}
PSO	1057.90	1103.27	168.13	12.396	1.196×10^{-2}
QGA	1137.83	1163.64	54.43	7.38	6.098×10^{-3}
BFO	1267.58	1280.03	77.48	8.80	6.884×10^{-3}
ABC	1187.72	1212.23	48.47	6.96	$\mathbf{5.740 \times 10^{-3}}$
BFRL	1054.07	1074.97	**44.92**	**6.70**	6.285×10^{-3}

It's obvious that the variance of BFRL is the smallest among all. Particularly, the relative standard deviation of BFRL is only 52.5% of that of PSO. Although the RSD of ABC and QGA are smaller than that of others, their optimal solutions are not satisfactory.

Figures 13 and 14 are the distribution boxplots of fuel costs of generators and risk index, respectively. From top to bottom, the five horizontals are the maximum, upper quartile, median, lower quartile and minimum of convergence result obtained in 10 runs. One can readily find that the length of BFRL is the shortest, which means its variation is the smallest. Besides, its median is also located at a relatively low position, which verifies that BFRL has strong global convergence ability with stable convergence.

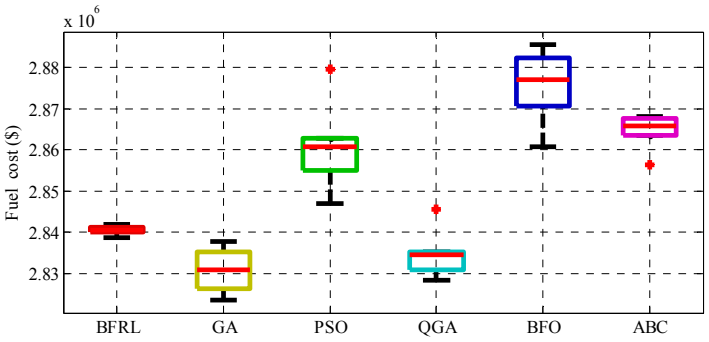

Figure 13. The boxplot of the fuel costs distribution.

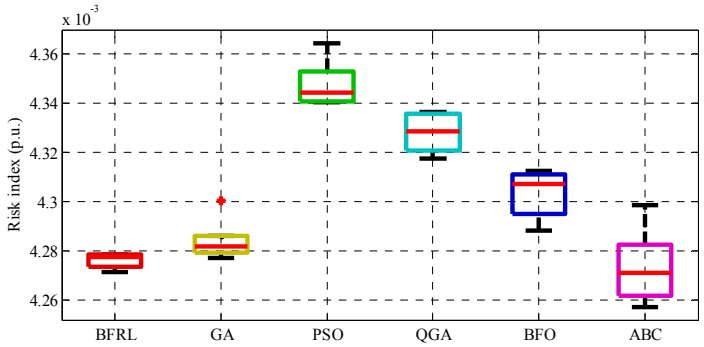

Figure 14. The boxplot of the risk index distribution.

5.5. Efficiency and Effectiveness of BFRL

From the above simulation results, it can be concluded that the comprehensive performance of BFRL is the best among all the algorithms, which includes the optimization rate, the quality of obtained optimal solution and the convergence stability.

Compared to other algorithms, BFRL can save more than 11 h in solving 96 new tasks of RBED during a day in total. Moreover, it tends to exploit the prior knowledge when initializing the knowledge matrix for a new task, i.e., the initial knowledge matrix of new task can be effectively developed from the optimal knowledge matrices of the related source task with the highest similarity, thus the knowledge matrix can converge in just a few iterations (less than 50 s). Moreover, BFRL has improved its efficiency by ten times through knowledge transfer.

The obtained integrated objective function f of BFRL is the second smallest among all the algorithms, which is only larger than that of GA. This is due to the following promising features:

- Deep local search: To approximate a high-quality local optimum with a smaller integrated objective function f, a chemotactic mode is adopted to search new solutions around the current

optimal solution, while a reproduction mode is employed to eliminate the bacteria with the low-quality local optimums.

- Wide global search: To increase the probability of obtaining a global optimum, several bacteria will be assigned for a random search in the action space with the dispersal mode.
- Proper balance between local search and global search: Each bacterium will implement a new action based on the common knowledge matrices Equations (18) and (19), a greedy action (i.e., a local search) will be selected if the random number is larger than the dispersal probability; otherwise a non-greedy action (i.e., a global search) will be chosen. As a consequence, a proper trade-off between local search and global search can be achieved.

For the convergence stability, other heuristic algorithms are incapable of knowledge transfer thus they may easily result in a low convergence stability, i.e., a significantly different optimum obtained in different runs. In contrast, BFRL can extract the optimal knowledge matrices from the sources tasks, thus the blind random search can be effectively avoided by utilizing the approximate optimal knowledge matrices, so that a convergence stability with a high-quality optimum can be realized.

6. Conclusions

In this paper, a novel model-free BFRL associated with transfer learning is proposed for RBED, which can be applied for discontinuous convex or nonconvex problems with multiple constraints. Besides, it can transform the informative information of source tasks into the state-action pair value function to accelerate the online optimization of a new task. Moreover, BFRL has a relatively simple structure and can converge with higher quality solutions in a short period of time. The main contributions are summarized as follows:

- The bacteria are regarded as multi-agent to accelerate the update of knowledge matrix in BFRL, while the high dimension of knowledge matrix can be considerably reduced by knowledge extension, such that the curse of dimension can be avoided;
- The active power deviation is defined as the similarity between source tasks and new tasks, and the online learning is accelerated significantly through transfer learning so that an online dynamic RBED can be achieved. Moreover, BFRL is adequate to handle large-scale problems;
- The multi-objective RBED is transformed into a single-objective problem via linear weighed method, and future research will investigate multi-objective algorithms associated with transfer learning for RBED. Moreover, this paper assumes the active power deviation is the only difference between source tasks and new tasks, which reduces the difficulty of transfer learning. In fact, the power gird topology, unit commitment and the fault type may vary dramatically, therefore how to extract these similarities is worth studying.
- RBED is based on the static ED while the dynamic multi-period coupled constraints are not considered. Hence, ongoing studies will also focus on the extension from single-scenario static optimization to dynamic multi-scenario optimization.

Acknowledgments: The authors gratefully acknowledge the support of National Key Basic Research Program of China (973 Program: 2013CB228205), National Natural Science Foundation of China (51477055), Yunnan Provincial Talents Training Program (KKSY201604044), and Scientific Research Foundation of Yunnan Provincial Department of Education (KKJB201704007).

Author Contributions: Chuanjia Han established the model, implemented the simulation and wrote this article; Bo Yang guided and revised the paper and refined the language; Tao Bao collected references; Tao Yu guided the research; Xiaoshun Zhang assisted in writing algorithms.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

Nomenclature

Variables

X_{f}	current condition
I_R	risk index
E_i	ith contingency
$P_{\mathrm{r}}(E_i)$	probability of E_i
$S_{\mathrm{ev}}(E_i)$	severity of E_i
$E_{\mathrm{F}i}$	fault of the ith transmission line
$E_{\mathrm{S\_F}i}$	single fault in system
L_i	ratio of actual transmission power to transmission power constraint of the ith line
ω_{Li}	overload faulty of the ith line
ω_{Vi}	voltage deviation of the ith node
U_i	voltage amplitude of the ith node
I_{RL}	total risk index of line overload
I_{RV}	total risk index of node voltage deviation
F_{C}	fuel costs of generators
C_{V}	value of total constraint violations
$P_{\mathrm{D}i}/Q_{\mathrm{D}i}$	active and reactive power of load i
U	node voltage amplitude
θ	node voltage phase angle
$P_{\mathrm{G}}/Q_{\mathrm{G}}$	active and reactive power of generator
θ_{ij}	voltage phase angle difference
$\lvert T_i \rvert$	apparent power of the ith line
P_{Gs}	generator active power on the slack bus
$\boldsymbol{Q}(s,a)$	element of knowledge matrix
s	state
a	action
$\boldsymbol{Q}$	knowledge matrix
$\boldsymbol{Q}^i$	i-th knowledge sub-matrices
r_1	random number
ρ_k^{ij}	update component of the Q-value matrix
$\psi^i(j,k,l)$	position of the ith bacteria
R	reward of a transition of state under a selected action
Δ	unit vector in the direction of swimming
C_k^i	step of swimming
$\boldsymbol{O}^i$	state-action probability matrix
r_2	random number
a_s	random global action
$P_{\mathrm{Ds}i}$	power demand of the ith load intervals
$\boldsymbol{e}^i$	transition matrix
$\boldsymbol{Q}^*$	optimal knowledge matrix

Sets

$\boldsymbol{S}_{\mathrm{G}}$	set of generators
$\boldsymbol{S}_{\mathrm{D}}$	set of load buses
$\boldsymbol{S}_{\mathrm{L}}$	set of lines
$\boldsymbol{S}_{\mathrm{pa}}$	state space
$\boldsymbol{A}$	action space
$\boldsymbol{S}_{\mathrm{UN}}$	set of all the normal operational transmission lines

Parameters

Δt	certain time interval
λ_i	failure rate of the ith transmission line
L_0	threshold of the ratio

M	penalty factor
a/b/c	positive constants used in the nonlinear utility function
μ_1	weight coefficient of line overload risk index
μ_2	weight coefficient of node voltage deviation risk index
μ_3	weight coefficient of the fuel costs of generators
μ_4	weight coefficient of the value of total constraint violations
z_1/z_2	normalization references
G_{ij}/B_{ij}	conductance and susceptance of line *i*–*j*
$P_{Gi\max}/P_{Gi\min}$	upper and lower bounds of the generator active power
$Q_{Gi\max}/Q_{Gi\min}$	upper and lower bounds of the generator reactive power
$T_{i\max}$	power limit of the *i*th line
$\zeta_{0i}/\zeta_{1i}/\zeta_{2i}$	coefficients of fuel costs
β	divergence factor to magnify the divergence of Q^i
α	learning factor
γ	discount factor
C^i_{start}	initial steps
C^i_{end}	final steps
$Iter_{\max}$	maximum iteration number
P_{r_ed}	dispersal probability
N_{v}	number of variables
N_{q}	number of controlled variables
N_{p}	number of employed bacteria
N_{c}	number of chemotactic
N_{f}	number of faults in the contingency
N_{ed}	number of dispersals
N_{re}	number of reproductions
N_{s}	number of swimming
Abbreviations	
BFRL	bacteria foraging reinforcement learning
RBED	risk-based economic dispatch
AI	artificial intelligence
SCOPF	security constrained optimal power flow
BFO	bacteria foraging optimization
OPF	optimal power flow
GA	genetic algorithm
QGA	quantum genetic algorithm
ABC	artificial bee colony
PSO	particle swarm optimization
AC	active current
RSD	relative standard deviation

References

1. Yao, W.; Jiang, L.; Wen, J.Y.; Wu, Q.H.; Cheng, S.J. Wide-area damping controller for power system inter-area oscillations: A networked predictive control approach. *IEEE Trans. Control Syst. Technol.* **2015**, *23*, 27–36. [CrossRef]
2. Yang, B.; Jiang, L.; Wang, L.; Yao, W.; Wu, Q.H. Nonlinear maximum power point tracking control and modal analysis of DFIG based wind turbine. *Int. J. Electr. Power Energy Syst.* **2016**, *74*, 429–436. [CrossRef]
3. Yang, B.; Zhang, X.S.; Yu, T.; Shu, H.C.; Fang, Z.H. Grouped grey wolf optimizer for maximum power point tracking of doubly-fed induction generator based wind turbine. *Energy Convers. Manag.* **2017**, *133*, 427–443. [CrossRef]
4. Zhou, W.; Lou, C.; Li, Z.; Lu, L.; Yang, H. Current status of research on optimum sizing of stand-alone hybrid solar-wind power generation systems. *Appl. Energy* **2010**, *87*, 380–389. [CrossRef]

5. Liu, J.; Wen, J.Y.; Yao, W.; Long, Y. Solution to short-term frequency response of wind farms by using energy storage systems. *IET Renew. Power Gener.* **2016**, *10*, 669–678.
6. Law, Y.W.; Alpcan, T.; Palaniswami, M. Security games for risk minimization in automatic generation control. *IEEE Trans. Power Syst.* **2015**, *30*, 223–232. [CrossRef]
7. Hetzer, J.; Yu, D.C.; Bhattarai, K. An economic dispatch model incorporating wind power. *IEEE Trans. Energy Convers.* **2008**, *23*, 603–611. [CrossRef]
8. Capitanescu, F.; Wehenkel, L. Improving the statement of the corrective security-constrained optimal power-flow problem. *IEEE Trans. Power Syst.* **2007**, *22*, 887–889. [CrossRef]
9. Bienstock, D.; Chertkov, M.; Harnett, S. Chance constrained optimal power flow: Risk-aware network control under uncertainty. *SIAM Rev.* **2012**, *56*, 67.
10. Alnaser, S.W.; Ochoa, L.F. Advanced network management systems: A risk-based AC OPF approach. *IEEE Trans. Power Syst.* **2015**, *30*, 409–418. [CrossRef]
11. Chiang, N.; Grothey, A. Solving security constrained optimal power flow problems by a structure exploiting interior point method. *Optim. Eng.* **2015**, *16*, 49–71. [CrossRef]
12. Fu, W.; Mccalley, J.D. Risk based optimal power flow. In Proceedings of the Porto Power Tech Conference, Porto, Portugal, 10–13 September 2001; pp. 1–6.
13. Li, Y.; Mccalley, J.D. Risk-based optimal power flow and system operation State. In Proceedings of the Power and Energy Society General Meeting, Calgary, AB, Canada, 29–30 July 2009; pp. 1–6.
14. Capitanescu, F.; Ramos, J.L.M.; Panciatici, P.; Kirschen, D.; Marcolini, A.M.; Platbrood, L.; Wehenkel, L. State-of-the-art, challenges, and future trends in security constrained optimal power flow. *Electr. Power Syst. Res.* **2011**, *81*, 1731–1741. [CrossRef]
15. Wang, Q.; Yang, A.; Wen, F.; Li, J. Risk-based security-constrained economic dispatch in power systems. *J. Mod. Power Syst. Clean Energy* **2013**, *1*, 142–149. [CrossRef]
16. Jiang, L.; Yao, W.; Wu, Q.H.; Wen, J.Y.; Cheng, S.J. Delay-dependent stability for load frequency control with constant and time-varying delays. *IEEE Trans. Power Syst.* **2012**, *27*, 932–941. [CrossRef]
17. Bertsekas, D.P. *Nonlinear Programming*; Athena Scientific: Nashua, NH, USA, 1999.
18. Zhao, H.; Wang, Y.; Guo, S.; Zhao, M.; Zhang, C. Application of a gradient descent continuous actor-critic algorithm for double-side day-ahead electricity market modeling. *Energies* **2016**, *9*, 725. [CrossRef]
19. Jiang, Q.; Geng, G.; Guo, C.; Cao, Y. An efficient implementation of automatic differentiation in interior point optimal power flow. *IEEE Trans. Power Syst.* **2010**, *25*, 147–155. [CrossRef]
20. Kazemtabrizi, B.; Acha, E. An advanced STATCOM model for optimal power flows using Newton's method. *IEEE Trans. Power Syst.* **2014**, *29*, 514–525. [CrossRef]
21. Gurobi Optimization. Gurobi Optimizer Reference Manual. Available online: http://www.gurobi.com (accessed on 13 December 2016).
22. IBM ILOG CPLEX Optimizer. Available online: http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/index.html (accessed on 23 December 2016).
23. Osman, M.S.; Abo-Sinna, M.A.; Mousa, A.A. A solution to the optimal power flow using genetic algorithm. *Appl. Math. Comput.* **2004**, *155*, 391–405. [CrossRef]
24. Vlachogiannis, J.G.; Stergaard, J. Reactive power and voltage control based on general quantum genetic algorithms. *Expert Syst. Appl.* **2009**, *36*, 6118–6126. [CrossRef]
25. Lin, W.M.; Tu, C.S.; Tsai, M.T. Energy management strategy for microgrids by using enhanced bee colony optimization. *Energies* **2015**, *9*, 5. [CrossRef]
26. Chen, Z.; Xiong, R.; Wang, K.; Jiao, B. Optimal energy management strategy of a plug-in hybrid electric vehicle based on a particle swarm optimization algorithm. *Energies* **2015**, *8*, 3661–3678. [CrossRef]
27. Liu, Y.; Passino, K.M. Biomimicry of social foraging bacteria for distributed optimization: Models, principles, and emergent behaviors. *J. Optim. Theory Appl.* **2002**, *115*, 603–628. [CrossRef]
28. Gazi, V.; Passino, K.M. *Bacteria Foraging Optimization*; Springer: Berlin/Heidelberg, Germany, 2011.
29. Pan, S.J.; Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359. [CrossRef]
30. Taylor, M.E.; Stone, P. Transfer learning for reinforcement learning domains: A survey. *J. Mach. Learn. Res.* **2009**, *10*, 1633–1685.
31. Zhang, X.S.; Yu, T.; Yang, B.; Cheng, L.F. Accelerating bio-inspired optimizer with transfer reinforcement learning for reactive power optimization. *Knowl.-Based Syst.* **2017**, *116*, 26–38. [CrossRef]

32. Zhang, X.S.; Chen, Y.X.; Yu, T.; Yang, B.; Qu, K.P.; Mao, S. Equilibrium-inspired multiagent optimizer with extreme transfer learning for decentralized optimal carbon-energy combined-flow of large-scale power systems. *Appl. Energy* **2017**, *189*, 157–176. [CrossRef]
33. Hu, D.H.; Zheng, V.W.; Yang, Q. Cross-domain activity recognition via transfer learning. *Pervasive Mob. Comput.* **2011**, *7*, 344–358. [CrossRef]
34. Cao, X.; Wang, Z.; Yan, P.; Li, X. Transfer learning for pedestrian detection. *Neurocomputing* **2013**, *100*, 51–57. [CrossRef]
35. Yu, T.; Liu, J.; Chan, K.W.; Wang, J.J. Distributed multi-step Q(λ) learning for optimal power flow of large-scale power grids. *Int. J. Electr. Power Energy Syst.* **2012**, *42*, 614–620. [CrossRef]
36. Ni, M.; Mccalley, J.D.; Vittal, V.; Tayyib, T. Online risk-based security assessment. *IEEE Trans. Power Syst.* **2002**, *22*, 59. [CrossRef]
37. Li, W. *Risk Assessment of Power Systems: Models, Methods, and Applications*; John Wiley & Sons: Hoboken, NJ, USA, 2014.
38. Zhang, X.S.; Xu, H.; Yu, T.; Yang, B.; Xu, M.X. Robust collaborative consensus algorithm for decentralized economic dispatch with a practical communication network. *Electr. Power Syst. Res.* **2016**, *140*, 597–610. [CrossRef]
39. Yu, T.; Zhou, B.; Chan, K.W.; Cheng, L. Stochastic optimal relaxed automatic generation control in non-Markov environment based on multi-step learning. *IEEE Trans. Power Syst.* **2011**, *26*, 1272–1282. [CrossRef]
40. Watkins, C.J.C.H.; Dayan, P. Q-learning. *Mach. Learn.* **1992**, *8*, 279–292. [CrossRef]
41. Lin, C.; Xu, Z. Wheel torque distribution of four-wheel-drive electric vehicles based on multi-objective optimization. *Energies* **2015**, *8*, 3815–3831. [CrossRef]
42. Leslie, D.S.; Collins, E.J. Individual Q-learning in normal form games. *Siam J. Control Optim.* **2005**, *44*, 495–514. [CrossRef]
43. Barrows, C.; Blumsack, S. Transmission switching in the RTS-96 test system. *IEEE Trans. Power Syst.* **2012**, *27*, 1134–1135. [CrossRef]
44. Fang, K.T. *Uniform Design and Design Tables*; Science: Beijing, China, 1994. (In Chinese)
45. IEEE Reliability Test System Task Force. IEEE reliability test system. *IEEE Trans. Power Appar. Syst.* **1979**, *98*, 2047–2054.
46. Holmberg, H.; Tuomaala, M.; Haikonen, T.; Ahtila, P. Allocation of fuel costs and CO_2-emissions to heat and power in an industrial CHP plant: Case integrated pulp and paper mill. *Appl. Energy* **2012**, *93*, 614–623. [CrossRef]