

Research Article

An Enriched Prediction Intervals Construction Method with Hybrid Intelligent Optimization

Jiazheng Lu,^{1,2} Guoyong Zhang ,^{1,2} Bo Li,^{1,2} Yu Liu,^{1,2} and Jun Guo^{1,2}

¹State Key Laboratory of Disaster Prevention and Reduction for Power Grid Transmission and Distribution Equipment, Changsha 410129, Hunan Province, China

²State Grid Hunan Electric Power Company Disaster Prevention and Reduction Center, Changsha 410129, Hunan Province, China

Correspondence should be addressed to Guoyong Zhang; 517086863@qq.com

Received 10 August 2018; Accepted 21 November 2018; Published 3 December 2018

Academic Editor: Giacomo Falcucci

Copyright © 2018 Jiazheng Lu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Prediction intervals (PIs), within which future observations of time series are expected to fall, are a powerful method for uncertainty modeling and forecasting. This paper presents the construction of optimal PIs using an enriched extreme learning machine (ELM)-based method. While quality evaluation indices for PIs on reliability and sharpness of prediction results have been defined in the literature, this paper proposes a new PIs evaluation index, robustness, which focuses on the forecasting error. Combined with the above three indices, a more comprehensive objective function is then formed for optimal PIs construction. The paper also proposes an efficient hybrid quantum-behaved particle swarm optimization method with bacterial foraging mechanism to optimize the parameters in the ELM model. The effectiveness of the additional robustness index and the proposed improved ELM approach in determining higher quality PIs is demonstrated by applying them to PIs constructions for the cases of prediction in different datasets.

1. Introduction

To make optimal water resource allocation, precise runoff prediction is always needed and is one of the most important issues in the field of hydrology [1]. In the literature, most of forecasting methods are focused on developing accurate deterministic point forecasting methods for runoff time series [2–4]. In actual case applications, single point prediction result is more popular than prediction interval because of its convenience to implement. However, for the point forecasting method, the predicted runoff value is provided without any information about associated uncertainties. During the process of decision making and operational planning, it is important to know how well the predictions match the real targets and how large the risk of unmatching is. Prediction Intervals (PIs) are excellent tools for the quantification of uncertainties associated with point forecasting [5, 6]. By definition, a PI is an interval consisting of lower and upper bound, in which the predicted value will fall with a certain

probability [7]. Recently, Prediction Intervals (PIs) have been widely accepted to quantify uncertainties associated with point forecasts [8, 9].

According to the existing interval forecasting methods, the probability density of the point forecasting error is always used to generate PIs [10, 11]. Without any prior knowledge of the point forecasting, a novel method in [12] adopts the two outputs of Neural Networks (NNs) to directly construct upper and lower bounds of PIs. Traditional NNs are widely used to construct PIs owing to outstanding generalization performance and approximation ability [13–15]. However, conventional NNs-based method causes inevitable limitations, such as overtraining [16]. As a new learning algorithm for training traditional feed forward neural networks, Extreme Learning Machine (ELM) [17] has been applied to construct PIs for its iterative-free learning mechanism [18, 19]. The performance of ELM has also been shown to have faster learning speed and better generalization ability than traditional NNs in [20–22]. In these ELM-based

prediction methods, PIs with associated confidence levels are generated through minimizing the PIs evaluation functions and optimizing the parameters in the ELM to produce high quality PIs. But the problem to determine the optimal PI still remains to be solved.

For the optimal PIs, higher reliability and sharpness are both expected [23]. Previously, a number of objective functions for training the prediction model have been proposed aiming to assess the overall quality of PIs with a single value. The Coverage Width-based Criterion (CWC) [12] based PIs objective function gives more weight to the property of reliability based on the exponential penalty term. Considering that the reliability is usually defined as the fundamental feature and determines the validity of PIs, the authors of [12] then modified the CWC in [24, 25] by regarding the reliability as the hard constraint. However, this method would cause the value of sharpness becomes larger than it needs to be, since the hard reliability constraint is difficult to be satisfied without sacrificing the sharpness. Hence, PIs construction still has room to be improved, with respect to PIs evaluation framework and optimization objective function.

In this paper, a new PIs evaluation index and an objective function have been formulated to comprehensively account for the properties of obtained PIs. Usually, reliability and sharpness are considered as main required properties of interval forecasting. However, the PIs error information for samples with their target values beyond the bounds is a key measure to assess the potential risk of operational planning. Regarding the actual values beyond the bounds of PIs, it will cause the overload or loss load. Hence, lower error values are also needed to be considered in the objective function of PIs optimization. An additional PIs evaluation index, which focuses on the forecasting error information, is therefore developed and defined as robustness. Combined with the reliability and sharpness indices, a more comprehensive objective function is formulated for training the ELM to obtain optimal PIs. To minimize the new constructed objective function of PIs optimization, this paper proposes the integration of the bacterial foraging mechanism into quantum-behaved particle swarm optimization method to form a hybrid intelligent optimization method for the determination of the parameters in the prediction model.

The rest of this paper is organized as follows. Section 2 describes the required PIs evaluation indices and the proposed objective function. Section 3 describes the implementation of proposed PIs construction approach based on ELM and hybrid intelligent optimization method. Case studies results and comparisons of proposed approach with benchmarks are presented in Section 4. Finally, Section 5 concludes the whole paper.

2. Evaluation Framework for Optimal PIs

Reliability and sharpness are widely considered as the main required properties of optimal PIs [12–14]. In this section, an additional PIs evaluation index that focuses on the interval forecasting error is defined as robustness. A novel objective function for PIs optimization is then formulated to

comprehensively account for the reliability, sharpness, and robustness properties of constructed PIs.

2.1. Reliability. According to the PIs definition [12], a PI with lower bound $L^\alpha(x_i)$ and upper bound $U^\alpha(x_i)$ bracket the target value t_i with nominal confidence level α can be expressed as follows:

$$P(t_i \in [L^\alpha(x_i), U^\alpha(x_i)]) = 100(1 - \alpha)\% \quad (1)$$

Reliability is referred to as the statistical consistency of PI coverage probability and nominal confidence level. To assess the reliability of constructed PIs, PI coverage probability (PICP) is proposed in [5] to describe the probability that target values will be covered by the upper and lower bounds. The PICP is defined as follows:

$$\text{PICP} = \frac{1}{N_t} \sum_{i=1}^{N_t} \xi_i \quad (2)$$

where N_t is the size of the entire test samples, and ξ_i is the indicator of PICP. If the future target value t_i is covered between the lower bound L_i and upper bound U_i , then $\xi_i=1$, otherwise $\xi_i=0$. ξ_i can be expressed as

$$\xi_i = \begin{cases} 1 & \text{if } t_i \in [L_i, U_i] \\ 0 & \text{if } t_i \notin [L_i, U_i] \end{cases} \quad (3)$$

2.2. Sharpness. Sharpness is referred to as the ability of PIs to concentrate the probabilistic forecasts information about future outcome [5]. For the optimal PIs, the target t_i is expected to lie within the bounds of PIs with higher reliability. This can be easily achieved if we just take the PICP into account, which will result in the lacking of useful forecasting information for decision-maker. The PI normalized average width (PINAW) index is applied in [12] to quantitatively measure the sharpness of PIs, which is defined as

$$\text{PINAW} = \frac{1}{N_t R} \sum_{i=1}^{N_t} (U_i - L_i) \quad (4)$$

where R is the maximum range of targets.

2.3. Robustness. Robustness is referred to as the ability of PIs to resist the probabilistic forecasts error during execution. On the basis of satisfactory reliability requirement and narrower interval width, this paper proposes that the forecasting errors for samples with target values beyond PIs bounds should also diminish towards zero as closely as possible. According to this, the Average Width Error (AWE) is proposed as a key measure to assess the potential risk of operation, and it is defined by

$$\text{AWE} = \frac{1}{\alpha N_t R} \sum_{i=1}^{N_t} E_i \quad (5)$$

$$E_i = \begin{cases} t_i - U_i & \text{if } t_i > U_i \\ 0 & \text{if } L_i \geq t_i \geq U_i \\ L_i - t_i & \text{if } t_i < L_i \end{cases} \quad (6)$$

where $\alpha \cdot N_t$ means the size of nominal error samples, and E_i is the indicator of width error information for each sample.

2.4. Objective Function. A number of objective functions have been proposed in the literature [12, 18, 24, 25], aiming to assess the overall quality of PIs based on the reliability and sharpness evaluation index. The Coverage Width-based Criterion (CWC) [12] gives more weight to the variation of PICP and if it is lower than the nominal confidence level of Eq. (1), the corresponding exponential term penalty will be accounted for. By using this objection function, it is hard to decide the coefficient for penalty term. For the Constrained CWC (CCWC) [24], the PICP is regarded as the hard constraint but it also results in large width of the PIs. For the interval score criterion (ISC) [18], the score of each prediction point is calculated according to the modified sharpness information. Different from the above objection function, the proposed objective function focuses on the best combination of reliability, sharpness, and robustness. Taking the criterion of robustness into account together with that of reliability and sharpness, a new objective function for PIs optimization can be defined as

$$\min F = (1 + \gamma_{\text{PICP}}\sigma) \cdot (\text{PINAW} + \text{AWE}) \quad (7)$$

where σ is the penalty coefficient to ensure the reliability, and γ_{PICP} is a function of PICP. If PICP is less than the PI nominal confidence level, $\gamma_{\text{PICP}}=1$ and the penalty item will be accounted for to ensure the reliability. Otherwise, $\gamma_{\text{PICP}}=0$ and the optimization process will maximize the sharpness and robustness of constructed PIs.

An examination of Eq. (7) shows that this objective function gives the reliability index a higher priority over sharpness and robustness but complemented by the other two indices. The objective function aims to obtain high-quality PIs with smaller values for PINAW and AWE based on satisfied confidence level. Compared to the existing objective functions, the new objective function provides a more comprehensive PIs evaluation for the PIs construction method presented in the next section.

3. Hybrid Intelligent Optimization Method for Optimal PIs Construction

To construct PIs for the runoff time series, the ELM-based PIs construction method can be applied. The construction of PIs based on ELM model is described in Section 3.1 and the theory and implementation of hybrid intelligent optimization method in optimizing the objective function are described in Sections 3.2 and 3.3 below, respectively.

3.1. Construction of PIs Based on Extreme Learning Machine. As a novel learning algorithm for training single hidden-layer feedforward neural networks (SLFNs), ELM generates the hidden node parameters randomly and determines the output weight matrix analytically [16]. For N distinct samples

$\{(\mathbf{x}_i, \mathbf{t}_i)\}_{i=1}^N$, SLFNs with K hidden nodes and activation function $g(x)$ are mathematically modeled as [17]:

$$\sum_{j=1}^K \beta_j g_j(\mathbf{w}_j \cdot \mathbf{x}_i + b_j) = y_i, \quad i = 1, \dots, N \quad (8)$$

where \mathbf{w}_j is the weight vector connecting the j -th hidden neuron and the input nodes, β_j is the weight vector connecting the j -th hidden neuron and the output nodes, and b_j is the threshold of the j -th hidden node. If SLFNs can approximate these N samples with zero error, it means that

$$\sum_{i=1}^N \|\mathbf{y}_i - \mathbf{t}_i\| = 0 \quad (9)$$

i.e., there exist β_j , \mathbf{w}_j and b_j such that

$$\sum_{j=1}^K \beta_j g_j(\mathbf{w}_j \cdot \mathbf{x}_i + b_j) = \mathbf{t}_i, \quad i = 1, \dots, N \quad (10)$$

The above equation can then be written as [17]

$$\mathbf{H}\beta = \mathbf{T} \quad (11)$$

$$\mathbf{H} = \begin{bmatrix} g_j(\mathbf{w}_1 \cdot \mathbf{x}_1 + b_1) & \cdots & g_j(\mathbf{w}_K \cdot \mathbf{x}_1 + b_K) \\ \vdots & \cdots & \vdots \\ g_j(\mathbf{w}_1 \cdot \mathbf{x}_N + b_1) & \cdots & g_j(\mathbf{w}_K \cdot \mathbf{x}_N + b_K) \end{bmatrix}_{N \times K} \quad (12)$$

where \mathbf{H} is the hidden layer output matrix, and \mathbf{T} is the matrix of targets. Unlike standard NNs where all parameters need to be tuned, in ELM, the input weights and hidden biases are randomly assigned and fixed. Then the hidden layer output matrix \mathbf{H} can be determined, and the approximation of w^* , b^* and β^* can be obtained such that

$$\begin{aligned} & \|H(\omega_1^*, \dots, \omega_K^*, b_1^*, \dots, b_K^*)\beta^* - T\| \\ & = \min_{\beta} \|H(\omega_1, \dots, \omega_K, b_1, \dots, b_K)\beta - T\| \end{aligned} \quad (13)$$

With fixed input weights and hidden layer biases of ELM, training an SLFN is simply equivalent to finding the smallest norm least-squares solution of the above linear system:

$$\hat{\beta} = \mathbf{H}^\dagger \mathbf{T} \quad (14)$$

where \mathbf{H}^\dagger is the Moore-Penrose generalized inverse of the hidden layer output matrix \mathbf{H} .

To construct PIs for the runoff time series, the ELM-based interval forecasting method with the PIs generated at the output layer of the neural network is illustrated in Figure 1. In which, the first and second outputs of the ELM model correspond to the upper and lower bound of the PIs separately [6].

3.2. An Efficient Hybrid Intelligent Optimization Method for PIs Optimization. The optimization for the above ELM-based PIs construction aims to obtain the minimum objective

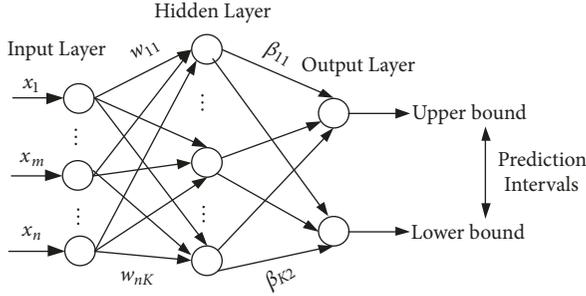


FIGURE 1: The structure of ELM model for runoff interval forecasting.

function values of Eq. (7) with respect to the best output weights β in the ELM model. To minimize the constrained optimization function and optimize the parameters of ELM, conventional optimization methods, such as Dynamic Programming (DP), Particle Swarm Optimization (PSO) algorithm, always suffer from the problem of being trapped into local optima [26–30]. Inspired by quantum mechanics, a new version of PSO named Quantum-behaved Particle Swarm Optimization (QPSO) [31] was proposed due to its guaranteed characteristic of global convergence. However, it still needs a local search mechanism to make a balance between the exploitation and exploration [32]. This paper proposes the integration of the bacterial foraging mechanism [33] into QPSO to form an efficient hybrid intelligent optimization method for the determination of the parameters in the ELM prediction model based on the above objective function.

3.2.1. The Description for Quantum-Behaved Particle Swarm Optimization Method. The QPSO algorithm assumes that there is a quantum delta potential well on each dimension at the local attractor point P_d [31]

$$P_d = \frac{(\varphi_{1d}P_{id} + \varphi_{2d}P_{gd})}{(\varphi_{1d} + \varphi_{2d})} \quad (15)$$

where $i = 1, 2, \dots, M$ indicates the particles of population, $d = 1, 2, \dots, D$ indicates the dimensions of each particle, P_{id} is the best position of i th particle, P_{gd} is the position of global best particle, and φ_{1d} and φ_{2d} are random vectors with range of (0, 1).

In QPSO, every particle has quantum behavior with its state formulated by wave function [34]. The probability density functions of particle's position can be deduced from the Schrödinger equation, and then the measurement of particle's position from the quantum state to the classical one can be implemented by using the Monte Carlo simulation method. The position of the i -th particle is updated as follows:

$$X_{id} = \begin{cases} P_d + \delta \cdot |\mathbf{mbest} - X_{id}| \cdot \ln\left(\frac{1}{u}\right), & 1 > h > 0.5 \\ P_d - \delta \cdot |\mathbf{mbest} - X_{id}| \cdot \ln\left(\frac{1}{u}\right), & 0 < h \leq 0.5 \end{cases} \quad (16)$$

where h and u are two random numbers distributed uniformly with range of (0, 1), and δ is suggested to decrease the value of δ linearly from δ_{\max} to δ_{\min} generally; \mathbf{mbest} is the mean best position, which is defined as the mean of P_{id} positions of all particles.

$$\mathbf{mbest} = \sum_{i=1}^M \frac{P_i}{M} = \left[\sum_{i=1}^M \frac{P_{i1}}{M}, \sum_{i=1}^M \frac{P_{i2}}{M}, \dots, \sum_{i=1}^M \frac{P_{iD}}{M} \right] \quad (17)$$

3.2.2. The Description for Hybrid Intelligent Optimization Method. It is evident from Eq. (16) that each component of the particle's updated position is determined by the local attractor P_d and the disturbing part. Like the original PSO algorithm, convergence of the particles' positions to their local attractors can guarantee the convergence of particles [28]. As a result, the local attractors in Eq. (16) will gather toward the global best value, which in turn makes the particle's current position converge to the global best position. The local attractors in Eq. (15) can be translated into

$$P_d = \frac{\varphi_{1d}}{\varphi_{1d} + \varphi_{2d}} (P_{id} - P_{gd}) + P_{gd} \quad (18)$$

It can be seen that the global best position guides the movement of local attractors. During the process of iteration, if the global best position is trapped into a local optimal point, it will mislead the particle's current position convergence, resulting in premature convergence. Aiming at this problem, this paper develops a Hybrid Quantum-behaved Particle Swarm Optimization (HQPSO) algorithm by introducing the bacterial foraging mechanism to update the position of global best position. As investigated in [35, 36], one of the major driving forces of Bacterial Foraging Optimization (BFO) is the *chemotactic* movement. However, the chemotaxis employed by BFO usually results in sustained oscillation, especially on flat fitness landscapes, due to the fixed chemotactic step size.

To make a balance between the global searching capability and local searching capability, this paper proposes a *dynamic approximation control strategy* to update the chemotactic step size. For the particles, the loss of global searching capability means it only flies within a small space, which will result in premature convergence, while the loss of local searching capability means that the possible flying cannot lead to perceptible effect on its fitness, which will result in a slow convergence speed [37]. To overcome these shortages, the proposed dynamic approximation control strategy aims to leave a large search area in the early iteration process and shrink the search range adaptively in the later iterations.

Suppose $P_{g,d}(s)$ represents the global best particle at s -th bacterial foraging operations, $C(s)$ is the size of the chemotactic step taken in the random direction specified by the tumble, the update of global best position based on the

bacterial foraging mechanism and dynamic approximation step control can be expressed as

$$P_{g,d}(s+1) = P_{g,d}(s) + C(s) \frac{\Delta(s)}{\sqrt{\Delta^T(s) \Delta(s)}} \quad (19)$$

$$s = 1, 2, \dots, N_c$$

$$C(0) = X_d^{\max} - X_d^{\min} \quad (20)$$

$$C(s+1) = \lambda \cdot C(s) \quad (21)$$

where Δ indicates a vector in the random direction whose elements lie in $[-1, 1]$, N_c indicates the number of chemotactic steps, X_d is the vector of global best position with respect to the d -th dimension, X_d^{\max} and X_d^{\min} are the bounds of the d -th dimension, and λ is a range parameter that used to control the decrease rate of chemotactic step size.

3.3. Implementation of the HQPSO-Based PIs Construction.

In this section, the proposed HQPSO algorithm is applied to minimize the objective function of optimal PIs construction. During the optimization process, the output weight β_j ($j = 1, 2, \dots, K$) of ELM is designed as the decision variable and the objective function value of each particle is designed to evaluate the quality of obtained PIs. The dimensions of each particle $[x_{i1}, x_{i2}, \dots, x_{iK}]$ in HQPSO algorithm are K , and the value of each particle X_{id} represents the value of output weight β_j ($j = 1, 2, \dots, K$). The steps of optimal PIs construction are given as follows:

- (1) *Normalization.* Normalize the runoff database to $[-1, 1]$, and then construct the PIs optimal model.
- (2) *Initialization.* The particle value $[x_{i1}, x_{i2}, \dots, x_{iK}]$ is randomly initialized and the initial bound is $[-1, 1]$.
- (3) *Construct PIs and calculate fitness F .* Construct PIs based on the parameters β_j ($j = 1, 2, \dots, K$) and then calculate the objective function value F for each particle.
- (4) *Update the value of particles.* The parameters β_j of each particle are updated by (16).
- (5) *Update the value of local attractor point.* Construct a new ELM-based model by using the updated β , and then calculate the objective function value for new particles. If this new objective function value is smaller than that of P_i , then P_i is updated by this new particle. What is more, if it is smaller than P_g , then P_g is updated.
- (6) *Update the value of global best particle.* By using the bacterial foraging mechanism, the position of global best particle can be updated as in Algorithm 1.
- (7) *Loop.* If the maximum iterations have not been reached, go to step (4). Otherwise, the iteration process is finished and the optimal output weight matrix of the ELM is obtained.

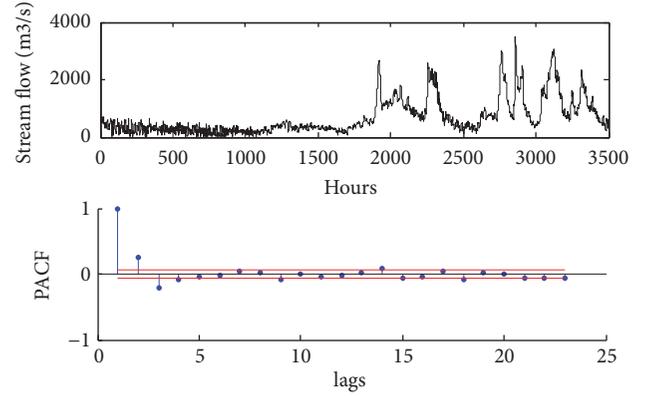


FIGURE 2: PACF analysis of original data. (a) Original runoff data, (b) PACF values of runoff data.

4. Results and Discussion

4.1. Datasets. The effectiveness of the additional robustness index and the proposed ELM optimization approach in determining higher quality PIs is demonstrated by applying them to PIs constructions for the predictions of runoff time series. The runoff datasets are from Zhexi hydropower station, which is located in Hunan province. The hourly inflow runoff data measured in 2015 is adopted in this research, out of which the first six months is used for training and the rest of the data are used for testing. All these data are normalized to $[-1, 1]$ before constructing interval forecasting model.

For training of the proposed prediction intervals construction method, as described in Section 3.1, ELM generates the hidden node parameters \mathbf{w} and \mathbf{b} randomly. The only job left for the users is to determine the output weights β and the inputs of ELM model. In this paper, the inputs of ELM are chosen based on the partial autocorrelation function (PACF) analyses [38], which is a useful tool to analyze the correlation between the candidate variables and the historical datasets. After normalizing the training data, the PACF values of inflow runoff data are calculated and shown in Figure 2. It can be seen that the lag orders of the inflow runoff time series are three and the inputs of ELM model are $[x_{i-1}, x_{i-2}, x_{i-3}]$. The number of neurons in hidden layers is set as seven according to the Kolmogorov theorem (i.e. $K = 2n + 1$).

The training process during the PIs optimization shows the convergence behavior of the constructed PIs for global best particle, which aims to find the best values for the output weights of the ELM. To evaluate the performance of proposed algorithm, three methods including PSO, QPSO, and HQPSO are applied to minimize the objective function in (7). In the case studies, the PINC is set to 90%, and the coverage probability for training phase is set 1-2% higher than the nominal confidence level as pointed out in [11]. The penalty coefficient σ is set to 10. The major parameters for the three optimization methods are given in Table 1. The number for iterations and population size in three methods are set to 500 and 100, respectively.

TABLE 1: Parameters for optimization methods.

| Method | Parameter | Value |
|--------|-----------------|-------|
| PSO | $c1$ | 2 |
| | $c2$ | 2 |
| | w_{\max} | 0.9 |
| | w_{\min} | 0.5 |
| QPSO | δ_{\max} | 0.9 |
| | δ_{\min} | 0.5 |
| HQPSO | N_c | 15 |
| | N_s | 5 |
| | λ | 0.5 |

4.2. Performance of HQPSO-Based PIs Optimization. For the training process of PIs optimization, convergence properties for all the case studies are shown in Figure 3. The y-coordinate of Figure 2 means the optimization results of PIs construction according to (7). It is shown that the proposed HQPSO algorithm has much better global searching ability and convergence property than PSO and QPSO method. For the beginning of iteration, owing to the introduction of local searching mechanism, the fitness of global best particle with HQPSO decreases rapidly, which ensures the efficiency of the swam search. For the later evolution process, owing to the decline of population diversity, there will appear premature convergence. Among three optimization methods, the PSO performs worst since it traps into a local optimum easily. Compared to PSO, QPSO has a better global searching ability, but it makes only a little improvement from 300 to 500 iterations due to the poor local searching ability. Conversely, the HQPSO in the same iteration range still has the global best particle decreasing in the fitness value as shown in Figure 3.

The original QPSO and HQPSO method employ the same parametric setup, except with the difference of the chemotactic step size and swimming length in the bacterial foraging mechanism. The chemotactic step size $C(s)$ was kept at 0.1 in the classical BFO [29]. For the dynamic approximation control strategy, the chemotactic step size shows exponential decline as the bacterial foraging iterative process advances. For PIs optimization, the optimization range for output weight β is set to be $[-100, 100]$. To obtain optimum accuracy, the chemotactic steps in the later iterations are required to be reduced to nearly 0.001. Hence, the value of λ can be calculated by a given value of N_c . Figure 4 illustrates the relationship between the objective function and the number of generations for different chemotactic steps N_c . As evident from the results, when the chemotactic step size is fixed at 0.1, the objective function decreases rapidly at the beginning, but it suffers from premature convergence. In the results of dynamic step size control strategy, for larger chemotactic steps N_c , the objective function converges faster.

Figure 5 illustrates the characteristics between PIs optimization objective function and the number of generations for different swimming lengths N_s of the bacteria. As evident, when the swimming length is bigger, the objective function decreases faster and converges faster. However, the larger chemotactic steps and bigger swimming lengths will lead to higher computational time. From Figure 5, it can also be observed that, for longer iteration process, e.g., 500

generations and above, the small change of the values for chemotactic steps and swimming lengths does not affect the optimization results, and therefore the values for N_s and N_c can be set at smaller values to keep the computational time down. Conversely, the values can be at larger for a shorter iteration process, e.g., 200 generations.

4.3. Discussions on Different Objective Functions. The effectiveness of ELM has already been demonstrated to have faster learning speed and better generalization ability than traditional NNs in [20–22]. In this section, the effectiveness of proposed PIs objective function in (7) is demonstrated by comparing it with CWC, constrained CWC (CCWC), and the interval score-based criterion (ISC). The merits of using CWC or ISC have previously been compared with conventional interval forecasts methods, e.g., exponential smoothing and quantile regression. Hence, this paper focuses on the effectiveness of proposed PIs objective function. The PIs are constructed with 90% and 80% nominal confidence level, respectively. The numerical results of different case studies are given in Table 2 including the reliability index PICP, the sharpness index PINAW, and the robustness index AWE. At two different confidence levels, it can be seen from Table 2 that all these methods with different objection functions can provide fairly satisfactory coverage probability. The reason for this is obvious since all of these cost functions take the reliability index as the primary requirement of forecasting. For the performance of sharpness and robustness, it can be seen that the PINAWs of the proposed objective function are the smallest for all the case studies, indicating the highest sharpness of the obtained PIs. For CWC and CCWC, the penalty function is only related to the value of PICP, and the hard PICP constraint in CCWC is difficult to be satisfied without sacrificing the width of the PIs, which makes the values of PINAW become larger than they need to be. Besides, the CWC and CWCC objective functions do not take the error information into account and they just focus on the narrower width of PIs, which will lead to much larger values for AWE. For ISC, the generated PIs demonstrate fair robustness due to the introduction of interval score, but the interval score cannot quantitatively distinguish the contributions of robustness and sharpness, and they generally have larger width than the present proposed approach indicating a lower sharpness.

The obtained PIs for runoff datasets with PINC=90% and PINC=80% are displayed in Figure 6, where the actual measured data are covered by the constructed PIs in a great percentage. According to Figure 6, it can be observed that the lower and upper bounds have a good performance in following the change of real test samples, which demonstrates that the proposed approach is effective in construction of optimal PIs.

4.4. Performance Variation for Different Application Dataset. The effectiveness of the proposed ELM optimization approach in determining higher quality PIs is demonstrated by applying it to PIs constructions for the predictions of load demand. The load demand data are from Tasmania regional market in Australian National Electricity Market

- (a) Initialize parameters: $s=1, d=1$.
- (b) Tumble: Generate a random vector Δ , with each element Δ_d belongs to a random number on $[-1, 1]$.
- (c) Move: Update $P_{g,d}(s+1)$ according to (19) and calculate its new fitness F .
- (d) Swim: Let $m=0$ (counter for swimming length), while m is smaller than the maximum swimming length N_s , let $m=m+1$. If F is smaller than $f(p_g)$, then update p_g . Else, set m to be the maximum swimming length.
- (e) Loop: If $d < D, d=d+1$, go to step (b). In this case, continue the foraging operation for next dimension variable.
- (f) Loop: If $s < N_c, s=s+1$, update the chemotactic step size $C(s)$ according to (21) and go to step (b). Else, terminate the life of bacterial foraging operation.

ALGORITHM 1: Bacterial Foraging Searching Mechanism.

TABLE 2: PIs evaluation results with different confidence levels.

| Methods | 90% confidence level | | | 80% confidence level | | |
|-------------------|----------------------|----------|--------|----------------------|----------|--------|
| | PICP(%) | PINAW(%) | AWE(%) | PICP(%) | PINAW(%) | AWE(%) |
| ELM-CWC | 90.63 | 10.65 | 1.98 | 81.28 | 7.53 | 3.48 |
| ELM-CCWC | 91.23 | 11.30 | 1.17 | 83.72 | 8.51 | 3.29 |
| ELM-ISC | 90.80 | 9.67 | 1.30 | 80.45 | 6.96 | 2.57 |
| Proposed approach | 90.15 | 7.88 | 1.02 | 80.16 | 5.01 | 2.03 |

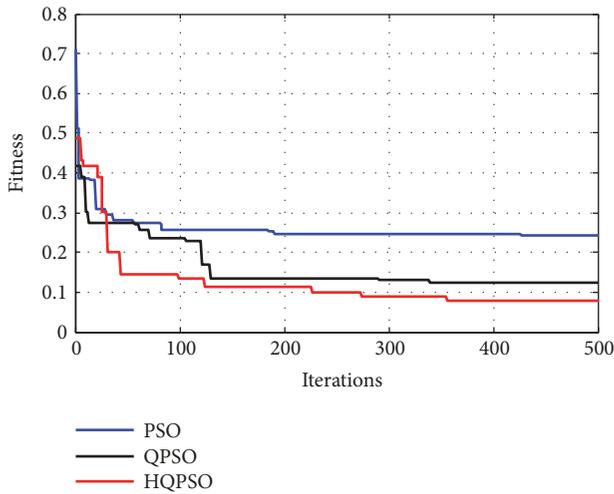


FIGURE 3: Convergence curves of different algorithms for runoff interval forecasting.

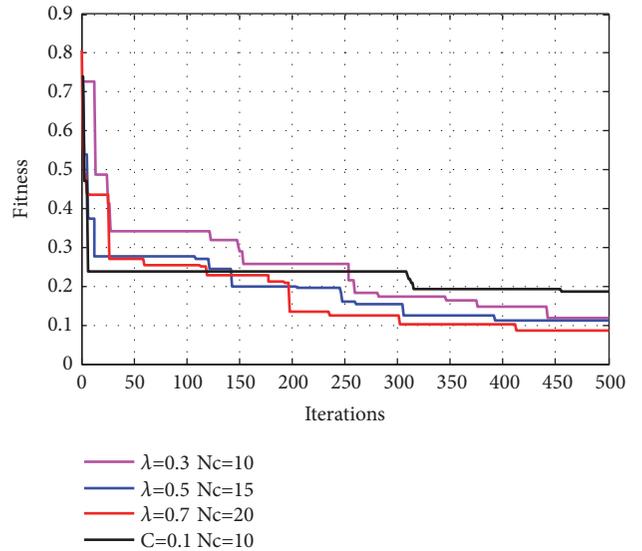


FIGURE 4: Performance value for different chemotactic steps.

(ANEM) [39]. The chosen time periods are from Jan 2016 to June 2016 with half an hour trading interval, out of which the first three months is used for training the ELM model and the rest of the data are used for testing the prediction performance of the proposed algorithm. Half-hour ahead PIs are implemented for load demand. The numerical results of different case studies are given in Table 3 in terms of the reliability index PICP, the sharpness index PINAW, and the robustness index AWE. At two different confidence levels, it can be seen from Table 3 that the proposed method generally has fairly satisfactory coverage probability and lower values for PINAW index and AWE index (i.e., higher sharpness and robustness) than traditional methods.

The obtained PIs for load demand data with PINC=90% are displayed in Figure 7, where the actual measured data are covered by the constructed PIs in a great percentage.

According to Figure 7, it can also be observed that the lower and upper bounds have a good performance in following the change of real test samples for all case studies. Therefore, experimental testing results demonstrate the effectiveness of the proposed ELM-based interval forecasting approach in improving the quality of obtained PIs with a combination of higher reliability, sharpness and robustness.

5. Conclusions

The ELM based PIs construction method has been applied and extended for PIs optimization in this paper. For the training of ELM model, a new evaluation index that focuses on the PIs error information has been developed to evaluate the robustness of PIs. A novel objective function for PIs

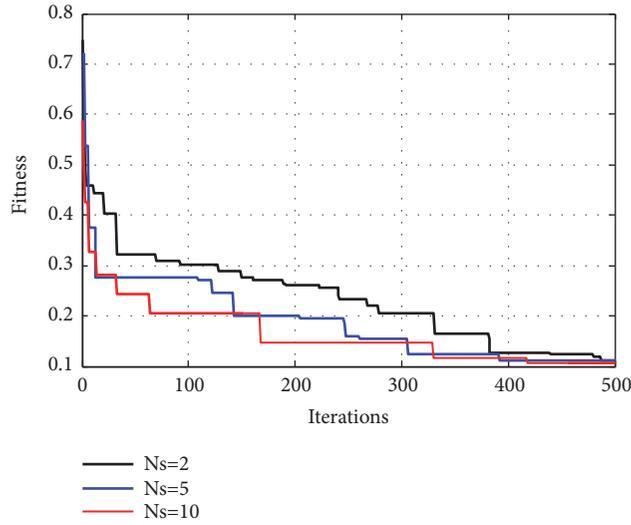


FIGURE 5: Performance value for different swimming lengths.

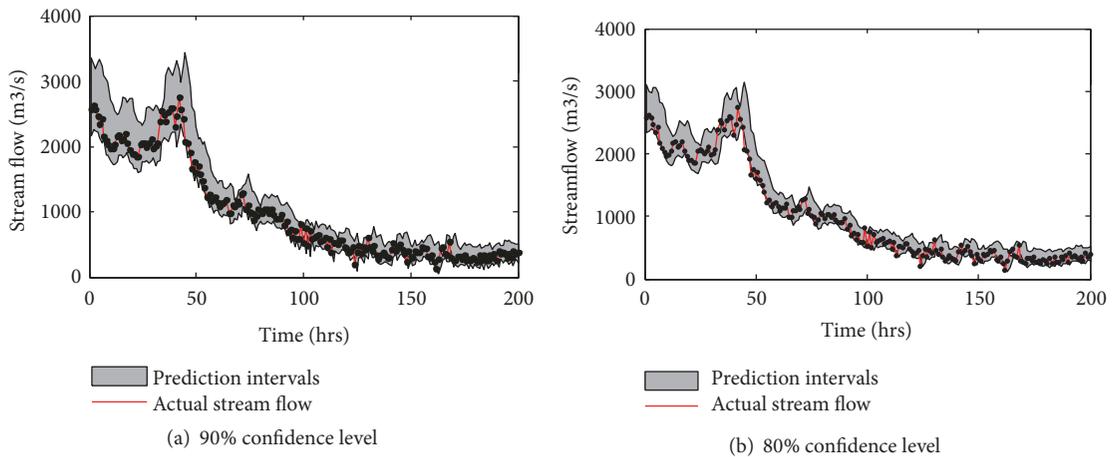


FIGURE 6: PIs with PINC with different confidence level for runoff datasets.

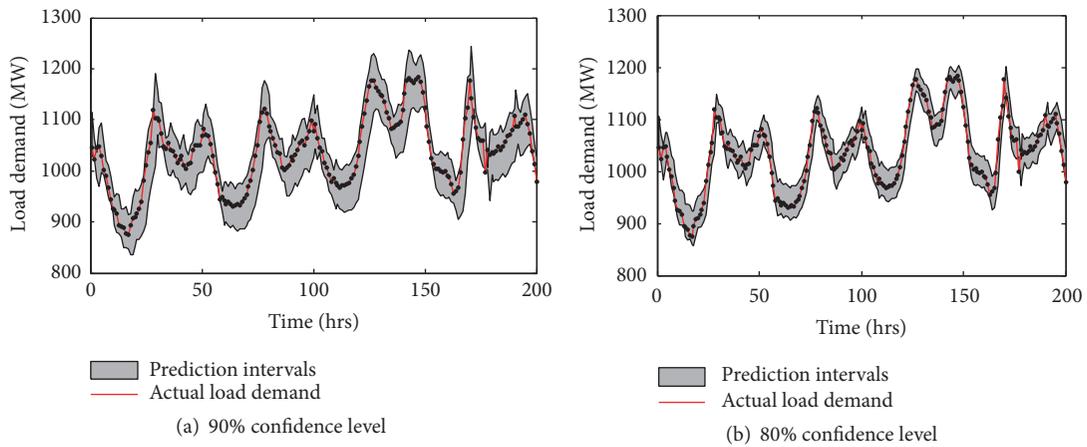


FIGURE 7: PIs with PINC with different confidence level for load demand data.

TABLE 3: PIs evaluation results with different confidence levels for load demand data.

| Methods | 90% confidence level | | | 80% confidence level | | |
|-------------------|----------------------|----------|--------|----------------------|----------|--------|
| | PICP(%) | PINAW(%) | AWE(%) | PICP(%) | PINAW(%) | AWE(%) |
| ELM-CWC | 90.65 | 15.07 | 3.28 | 80.65 | 11.42 | 4.61 |
| ELM-CCWC | 91.19 | 16.23 | 3.15 | 79.26 | 13.92 | 4.42 |
| ELM-ISC | 89.88 | 13.95 | 1.83 | 80.42 | 12.08 | 3.01 |
| Proposed approach | 90.56 | 13.28 | 1.66 | 80.93 | 10.62 | 2.56 |

optimization has also been formulated to comprehensively account for the properties of reliability, sharpness, and robustness. To solve the proposed nonlinear objective function in the training phase of the ELM, an improved QPSO with the bacterial foraging mechanism has been proposed to minimize the proposed objective function and optimize the parameters of the ELM model. To make a balance between the exploitation and exploration of the search space, this paper also develops a dynamic approximation control strategy to update the chemotactic step size. The effectiveness of proposed method has been validated by using it to construct optimal PIs for different datasets. The results have illustrated that the proposed ELM-based method can provide much higher quality interval forecasting information.

Data Availability

Data generated by the authors or analyzed during the study are available from the following options: (1) the data related to this research is available upon request by contacting the corresponding author. (2) All data generated or analyzed during the study are included in the published paper.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The authors are grateful to Qingpu Li and Jing Luo for their contribution in proofreading this manuscript and giving advise on English writing. This work was supported by the Key Science and Technology Foundation of SGCC (Grants no.5216A015001M).

References

- [1] D. Peng, L. Qiu, J. Fang, and Z. Zhang, "Quantification of climate changes and human activities that impact runoff in the Taihu Lake basin, China," in *Mathematical Problems in Engineering*, vol. 2016, Mathematical Problems in Engineering, 2016.
- [2] M. Jakubcová, P. Máca, and P. Pech, "Parameter estimation in rainfall-runoff modelling using distributed versions of particle swarm optimization algorithm," *Mathematical Problems in Engineering*, vol. 2015, Article ID 968067, 2015.
- [3] S. Abdollahi, J. Raeisi, M. Khalilianpour, F. Ahmadi, and O. Kisi, "Daily mean streamflow prediction in perennial and non-perennial rivers using four data driven techniques," *Water Resources Management*, vol. 31, no. 15, pp. 4855–4874, 2017.
- [4] C. Huang, A. J. Newman, M. P. Clark, A. W. Wood, and X. Zheng, "Evaluation of snow data assimilation using the ensemble Kalman filter for seasonal streamflow prediction in the western United States," *Hydrology and Earth System Sciences*, vol. 21, no. 1, pp. 635–650, 2017.
- [5] J. T. G. Hwang and A. A. Ding, "Prediction Intervals for Artificial Neural Networks," *Journal of the American Statistical Association*, vol. 92, no. 438, pp. 748–757, 1997.
- [6] G. Zhang, Y. Wu, K. P. Wong, Z. Xu, Z. Y. Dong, and H. H.-C. Iu, "An Advanced Approach for Construction of Optimal Wind Power Prediction Intervals," *IEEE Transactions on Power Systems*, vol. 30, no. 5, pp. 2706–2715, 2015.
- [7] C. Chatfield, "Calculating interval forecasts," *Journal of Business and Economic Statistics*, vol. 11, no. 2, pp. 121–135, 1993.
- [8] N. A. Shrivastava, A. Khosravi, and B. K. Panigrahi, "Prediction interval estimation of electricity prices using pso-tuned support vector machines," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 2, pp. 322–331, 2015.
- [9] Q. Ni, S. Zhuang, H. Sheng, S. Wang, and J. Xiao, "An Optimized Prediction Intervals Approach for Short Term PV Power Forecasting," *Energies*, vol. 10, no. 10, p. 1669, 2017.
- [10] L. Chen, Y. Zhang, J. Zhou, V. P. Singh, S. Guo, and J. Zhang, "Real-time error correction method combined with combination flood forecasting technique for improving the accuracy of flood forecasting," *Journal of Hydrology*, vol. 521, pp. 157–169, 2015.
- [11] J. Zhang, L. Chen, V. P. Singh, H. Cao, and D. Wang, "Determination of the distribution of flood forecasting error," *Natural Hazards*, vol. 75, no. 2, pp. 1389–1402, 2015.
- [12] A. Khosravi, S. Nahavandi, D. Creighton, and A. F. Atiya, "Lower upper bound estimation method for construction of neural network-based prediction intervals," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 22, no. 3, pp. 337–346, 2011.
- [13] R. Taormina and K.-W. Chau, "ANN-based interval forecasting of streamflow discharges using the LUBE method and MOFIPS," *Engineering Applications of Artificial Intelligence*, vol. 45, pp. 429–440, 2015.
- [14] K. S. Kasiviswanathan and K. P. Sudheer, "Methods used for quantifying the prediction uncertainty of artificial neural network based hydrologic models," *Stochastic Environmental Research and Risk Assessment*, vol. 31, no. 7, pp. 1659–1670, 2017.
- [15] L. Ye, J. Zhou, X. Zeng, J. Guo, and X. Zhang, "Multi-objective optimization for construction of prediction interval of hydrological models based on ensemble simulations," *Journal of Hydrology*, vol. 519, pp. 925–933, 2014.

- [16] C. Wan, Z. Xu, P. Pinson, Z. Y. Dong, and K. P. Wong, "Probabilistic forecasting of wind power generation using extreme learning machine," *IEEE Transactions on Power Systems*, vol. 29, no. 3, pp. 1033–1044, 2014.
- [17] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, 2006.
- [18] C. Wan, Z. Xu, P. Pinson, Z. Y. Dong, and K. P. Wong, "Optimal prediction intervals of wind power generation," *IEEE Transactions on Power Systems*, vol. 29, no. 3, pp. 1166–1174, 2014.
- [19] X. Chen, Z. Y. Dong, K. Meng, Y. Xu, K. P. Wong, and H. W. Ngan, "Electricity price forecasting with extreme learning machine and bootstrapping," *IEEE Transactions on Power Systems*, vol. 27, no. 4, pp. 2055–2062, 2012.
- [20] F. Zhao, Y. Liu, K. Huo, and Z. Zhang, "Radar Target Classification Using an Evolutionary Extreme Learning Machine Based on Improved Quantum-Behaved Particle Swarm Optimization," *Mathematical Problems in Engineering*, vol. 2017, Article ID 7273061, 13 pages, 2017.
- [21] J. Cao and Z. Lin, "Extreme learning machines on high dimensional and large data applications: a survey," *Mathematical Problems in Engineering*, vol. 2015, Article ID 103796, 13 pages, 2015.
- [22] C.-U. Yeom and K.-C. Kwak, "Short-term electricity-load forecasting using a task-based extreme learning machine with knowledge representation," *Energies*, vol. 10, no. 10, 2017.
- [23] P. Pinson, H. A. Nielsen, J. K. Møller, H. Madsen, and G. N. Kariniotakis, "Non-parametric probabilistic forecasts of wind power: required properties and evaluation," *Wind Energy*, vol. 10, no. 6, pp. 497–516, 2007.
- [24] H. Quan, D. Srinivasan, and A. Khosravi, "Short-term load and wind power forecasting using neural network-based prediction intervals," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 2, pp. 303–315, 2014.
- [25] H. Quan, D. Srinivasan, and A. Khosravi, "Particle swarm optimization for construction of neural network-based prediction intervals," *Neurocomputing*, vol. 127, pp. 172–180, 2014.
- [26] Z.-K. Feng, W.-J. Niu, C.-T. Cheng, and S.-L. Liao, "Hydropower system operation optimization by discrete differential dynamic programming based on orthogonal experiment design," *Energy*, vol. 126, pp. 720–732, 2017.
- [27] X. Li, K. Xing, M. Zhou, X. Wang, and Y. Wu, "Modified Dynamic Programming Algorithm for Optimization of Total Energy Consumption in Flexible Manufacturing Systems," in *IEEE Transactions on Automation Science and Engineering*, vol. 99, p. 15, 1–15, 99, 2018.
- [28] F. van den Bergh and A. P. Engelbrecht, "A study of particle swarm optimization particle trajectories," *Information Sciences*, vol. 176, no. 8, pp. 937–971, 2006.
- [29] L. Nan, X. Zeng, Y. Du, Z. Dai, and L. Chen, "Shared Variable Extraction and Hardware Implementation for Nonlinear Boolean Functions Based on Swarm Intelligence," *Mathematical Problems in Engineering*, vol. 2018, Article ID 7104764, 2018.
- [30] B. Mao, Z. Xie, Y. Wang, H. Handroos, and H. Wu, "A Hybrid Strategy of Differential Evolution and Modified Particle Swarm Optimization for Numerical Solution of a Parallel Manipulator," *Mathematical Problems in Engineering*, vol. 2018, Article ID 9815469, 2018.
- [31] J. Sun, B. Feng, and W. Xu, "Particle swarm optimization with particles having quantum behavior," in *Proceedings of the Congress on Evolutionary Computation*, pp. 325–331, Portland, Ore, USA, 2004.
- [32] J. Sun, W. Fang, V. Palade, X. Wu, and W. Xu, "Quantum-behaved particle swarm optimization with Gaussian distributed local attractor point," *Applied Mathematics and Computation*, vol. 218, no. 7, pp. 3763–3775, 2011.
- [33] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," *IEEE Control Systems Magazine*, vol. 22, no. 3, pp. 52–67, 2002.
- [34] Z.-K. Feng, W.-J. Niu, and C.-T. Cheng, "Multi-objective quantum-behaved particle swarm optimization for economic environmental hydrothermal energy system scheduling," *Energy*, vol. 131, pp. 165–178, 2017.
- [35] V. Ravikumar Pandi, A. Biswas, S. Dasgupta, and B. K. Panigrahi, "A hybrid bacterial foraging and differential evolution algorithm for congestion management," *European Transactions on Electrical Power*, vol. 20, no. 7, pp. 862–871, 2010.
- [36] Y. P. Chen, Y. Li, G. Wang et al., "A novel bacterial foraging optimization algorithm for feature selection," *Expert Systems with Applications*, vol. 83, 1–17, 83, 2017.
- [37] X. F. Xie, W. J. Zhang, and Z. L. Yang, "Adaptive particle swarm optimization on individual level," in *Proceedings of the 6th International Conference on Signal Processing*, vol. 2, pp. 1215–1218, IEEE, Beijing, China, 2002.
- [38] M. T. Hagan and S. M. Behr, "The time series approach to short term load forecasting," *IEEE Transactions on Power Systems*, vol. 2, no. 3, pp. 785–791, 1987.
- [39] "Australian Energy Market Operator (AEMO) Information," <http://data.wa.aemo.com.au/#load-summary>.



Hindawi

Submit your manuscripts at
www.hindawi.com

