Hindawi
Shock and Vibration
Volume 2019, Article ID 3253280, 10 pages
https://doi.org/10.1155/2019/3253280

Research Article

Bridge Critical State Search by Using Quantum Genetic Firefly Algorithm

Shi-bo Tao,[1,2] **Dian-zhong Liu**,[1,2] **and Ai-ping Tang**[3,4]

[1]*School of Civil Engineering, Jilin Jianzhu University, Changchun, Jilin 130118, China*
[2]*Science and Technology Innovation Center for Structure and Earthquake Resistance of Jilin Province, Changchun, China*
[3]*Key Lab of Structures Dynamic Behavior and Control, Harbin Institute of Technology, Ministry of Education, Heilongjiang, Harbin 150090, China*
[4]*School of Civil Engineering, Harbin Institute of Technology, Harbin 150090, China*

Correspondence should be addressed to Shi-bo Tao; taoshibo1985@163.com

Received 28 September 2018; Revised 20 December 2018; Accepted 12 February 2019; Published 13 March 2019

Academic Editor: Huu-Tai Thai

When performing flutter analysis through the traditional methods, it is difficult to solve high-order strong nonlinear equations. For overcoming this difficulty, this paper establishes a double-parameter optimization model for searching the flutter critical wind speed and frequency. A new hybrid firefly algorithm called the quantum genetic firefly algorithm is presented to search the optimal solution to the optimization model. The proposed algorithm is the combination of the firefly algorithm and the quantum genetic algorithm. The results of the quantum genetic firefly algorithm are compared with the results shown by the firefly algorithm and quantum genetic algorithm. Numerical and experimental results of the proposed algorithm are competitive and in most cases are better than that of the firefly algorithm and quantum genetic algorithm.

1. Introduction

In 1940, the Tacoma Narrows Bridge collapsed due to the phenomenon called flutter. The flutter is a kind of aeroelastic divergence phenomena that can induce structural failure. Therefore, in the design of the long-span bridges, the critical flutter state has to be carefully investigated not to excite the flutter below the designed wind speed [1]. The semi-retro-solution method is the most common way to analyze the critical flutter state. The traditional semi-retro-solution methods are required to compare the roots of the flutter state equations repeatedly [2], as it has high complexity and large computation amount. It also requires frequent human intervention to judge the rationality of the calculation results. Too much intervention not only affects the efficiency of computation but also affects the accuracy and stability of the algorithm that may lead to the failure of the analysis process. Therefore, it is necessary to evaluate more efficient and automated method. For example, Lee et al. [3] calculated the onset flutter of the aeroelastic bridge system by using the quasi-steady approach and approximated formula. In these methods, it is not necessary to resolve highly nonlinear equations directly, but it needs to set the initial value of the proposed equation.

To avoid the difficulty of solving highly nonlinear equations, we transformed it into optimized problem solution and then established the model to research the optimal solution. Recently, the firefly algorithm (FA) has proved that this algorithm can tackle the optimization problems effectively. However, the performance of the traditional firefly algorithm depends on its control parameters. When the selection of the algorithm is unsuitable, the search can easily get stuck in a local optimum [4].

In this paper, the solution to the critical flutter state problem is converted to an optimization problem, and a double-parameter optimization model is established. A new optimization method for searching flutter critical wind velocity and frequency is presented based on the firefly algorithm combined with the quantum genetic algorithm.

The simulations were performed with three case studies to observe the quality of the quantum genetic firefly algorithm and compared with both the traditional firefly algorithm and the quantum genetic algorithm.

2. Theoretical Background

2.1. Flutter Equation. The uniformly self-existed aerodynamic force and moment of the bridge can be determined by the following expression with 18 derivative parameters, in which the first and second portions of the expression represent the lift and drag forces, and the final portion represents the corresponding pitching moment [5]:

$$\begin{aligned}
L &= \rho U^2 B\left[KH_1^*\frac{\dot{h}}{U} + KH_2^*\frac{B\dot{\alpha}}{U} + K^2H_3^*\alpha + K^2H_4^*\frac{h}{B}\right.\\
&\quad \left. + KH_5^*\frac{\dot{p}}{U} + K^2H_6^*\frac{p}{B}\right],\\
D &= \rho U^2 B\left[KP_1^*\frac{\dot{h}}{U} + KP_2^*\frac{B\dot{\alpha}}{U} + K^2P_3^*\alpha + K^2P_4^*\frac{h}{B}\right.\\
&\quad \left. + KP_5^*\frac{\dot{p}}{U} + K^2P_6^*\frac{p}{B}\right],\\
M &= \rho U^2 B^2\left[KA_1^*\frac{\dot{h}}{U} + KA_2^*\frac{B\dot{\alpha}}{U} + K^2A_3^*\alpha + K^2A_4^*\frac{h}{B}\right.\\
&\quad \left. + KA_5^*\frac{\dot{p}}{U} + K^2A_6^*\frac{p}{B}\right],
\end{aligned} \tag{1}$$

where h is the vertical displacement (m); α is the torsional displacement (nose-up positive) (deg); ρ is the air density (kg/m^3); $K = \omega B/U$ is the reduced frequency, in which ω is the circle frequency; B is the width of the bridge deck; U is the mean wind speed (m/s); and the parameters of H_i^*, P_i^*, and A_i^* ($i = 1\sim6$) are flutter derivatives for the lift, drag, and moment actions, respectively. These three parameters are related to K and can be obtained from the wind tunnel test; the dot $(\cdot)$ represents the derivation in relation to time t.

The motion differential equations of 3-DOFs under self-excited forces are as follows:

$$\begin{aligned}
m\left(\ddot{h} + 2\xi_h\omega_h\dot{h} + \omega_h^2 h\right) &= L,\\
m\left(\ddot{p} + 2\xi_p\omega_p\dot{p} + \omega_h^2 p\right) &= D,\\
I\left(\ddot{\alpha} + 2\xi_\alpha\omega_\alpha\dot{\alpha} + \omega_\alpha^2\alpha\right) &= M.
\end{aligned} \tag{2}$$

On substituting equation (1) in equation (2) and setting $s = tU/B$ and $K = B\omega/U$, the motion equations can be obtained as follows:

$$\begin{aligned}
\frac{\ddot{h}}{B} + 2\xi_h K_h\frac{\dot{h}}{B} + K_h^2\frac{h}{B} &= \frac{\rho B^2}{m}\left[KH_1^*\frac{\dot{h}}{B} + KH_2^*\dot{\alpha} + K^2H_3^*\alpha\right.\\
&\quad \left. + K^2H_4^*\frac{h}{B} + KH_5^*\frac{\dot{p}}{U} + K^2H_6^*\frac{p}{B}\right],\\
\frac{\ddot{p}}{B} + 2\xi_p K_p\frac{\dot{p}}{B} + K_p^2\frac{p}{B} &= \frac{\rho B^2}{m}\left[KP_1^*\frac{\dot{h}}{B} + KP_2^*\dot{\alpha} + K^2P_3^*\alpha\right.\\
&\quad \left. + K^2P_4^*\frac{h}{B} + KP_5^*\frac{\dot{p}}{U} + K^2P_6^*\frac{p}{B}\right],\\
\ddot{\alpha} + 2\xi_\alpha K_\alpha\dot{\alpha} + K_\alpha^2\alpha &= \frac{\rho B^2}{I}\left[KA_1^*\frac{\dot{h}}{B} + KA_2^*\dot{\alpha} + K^2A_3^*\alpha\right.\\
&\quad \left. + K^2A_4^*\frac{h}{B} + KA_5^*\frac{\dot{p}}{U} + K^2A_6^*\frac{p}{B}\right],
\end{aligned} \tag{3}$$

where m and I are the mass and moment of inertia, respectively; ξ_h, ξ_p, and ξ_α are the damping ratios of the vertical bending motion, lateral bending motion, and torsional motion, respectively; and ω_h, ω_p, and ω_α are, respectively, the circular frequency of the vertical bending motion, lateral bending motion, and torsional motion; Finally, h, p, and α are generalized displacements.

2.2. Determination of the Flutter Optimization Model. Solving equation (3), we get $h = h_0e^{i\omega t} = h_0e^{iKs}$, $p = p_0e^{i\omega t} = p_0e^{iKs}$, and $\alpha = \alpha_0e^{i\omega t} = \alpha_0e^{iKs}$. On substituting these results in equation (2), in which $X = \omega_h/\omega$, the following expressions with h_0/B, p_0/B, and α_0 can be obtained as

$$\begin{cases}
\left[-1 + X^2 - \dfrac{\rho B^2}{m}H_4^* + \left(2\xi_h X - \dfrac{\rho B^2}{m}H_1^*\right)i\right]\dfrac{h_0}{B} + \left[-\dfrac{\rho B^2}{m}H_6^* - \dfrac{\rho B^2}{m}H_5^*i\right]\dfrac{p_0}{B} + \left[-\dfrac{\rho B^2}{m}H_3^* - \dfrac{\rho B^2}{m}H_2^*\alpha_0\right]\alpha_0 = 0,\\
\left[-\dfrac{\rho B^2}{m}P_4^* - \dfrac{\rho B^2}{m}P_1^*i\right]\dfrac{h_0}{B} + \left[-1 + \left(\dfrac{\omega_p}{\omega_h}\right)^2 X^2 - \dfrac{\rho B^2}{m}P_6^* + \left(2\xi_p\dfrac{\omega_p}{\omega_h}X - \dfrac{\rho B^2}{m}P_5^*\right)i\right]\dfrac{p_0}{B} + \left[-\dfrac{\rho B^2}{m}P_3^* - \dfrac{\rho B^2}{m}P_2^*i\right]\alpha_0 = 0,\\
\left[-\dfrac{\rho B^4}{I}A_4^* - \dfrac{\rho B^4}{I}A_1^*i\right]\dfrac{h_0}{B} + \left[-\dfrac{\rho B^4}{I}A_6^* - \dfrac{\rho B^4}{I}A_5^*i\right]\dfrac{p_0}{B} + \left[-1 + \left(\dfrac{\omega_\alpha}{\omega_h}\right)^2 X^2 - \dfrac{\rho B^4}{I}A_3^* + \left(2\xi_\alpha\dfrac{\omega_\alpha}{\omega_h}X - \dfrac{\rho B^4}{I}A_2^*\right)i\right] = 0.
\end{cases} \tag{4}$$

In case if the above expressions have nonzero solution, the corresponding coefficients are as follows:

$$\begin{vmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{vmatrix} = 0, \tag{5}$$

where A_{ij} is the corresponding coefficient resulted from equation (4). An example is provided as follows:

$$A_{11} = \left(-1 + X^2 - \frac{\rho B^2}{m} H_4^*\right) + \left(2\xi_h X - \frac{\rho B^2}{m} H_1^*\right) i. \tag{6}$$

By solving equation (5), the polynomial equation which represents the parameter X is obtained, which is a real number in the critical flutter state. Therefore, the real and imaginary components of the polynomial equation should be zero. Equation (5) can be resulted as

$$\begin{cases} f_1 = A_{R6}^* X^6 + A_{R5}^* X^5 + A_{R4}^* X^4 + A_{R3}^* X^3 + A_{R2}^2 X^2 \\ \qquad + A_{R1}^* X + A_{R0}^* = 0, \\ f_2 = A_{I5}^* X^5 + A_{I4}^* X^4 + A_{I3}^* X^3 + A_{I2}^* X^2 + A_{I1}^* X + A_{I0}^* = 0, \end{cases} \tag{7}$$

where A_{Ri}^* and A_{Ii}^* are the real and imaginary coefficients which correspond to X^i. From equation (7), ω_{cr} and U_{cr} can be obtained for the purpose of the verification of the above equation. The parameters of ω_{cr} and U_{cr} are the critical circular frequency and critical wind speed, respectively. Considering a realistic bridge, ω_{cr} ranges between $\max\{\omega_\alpha, \omega_h, \omega_p\}$ and $\min\{\omega_\alpha, \omega_h, \omega_p\}$, and U_{cr} can be approximately estimated using equation (8) [6]:

$$\widehat{U}_{\text{cr}} = \left[1 + (\varepsilon - 0.5)\sqrt{\left(\frac{r}{b}\right)0.72\mu}\right]\omega_h b, \tag{8}$$

where $\varepsilon = \omega_\alpha/\omega_h$ (defined as the torsional frequency divided by the vertical frequency), r is the radius of gyration, μ is the mass ratio of structure and air, and b is the half width of the bridge. On the basis of equation (8), the provided results can be found in equation (9).

Considering the parameter $X = [\omega, U]$, equation (7) can be solved as $f(X) = \sqrt{f_1^2 + f_2^2} = 0$, where the constraint condition that is approximately close to zero can be found by equation (9), which is used to find the ω_{cr} and U_{cr} values:

$$\begin{aligned} &\text{minimize:} \quad f(X) = \sqrt{f_1^2 + f_2^2}, \\ &\text{s.t.} \quad \begin{cases} \min\{\omega_\alpha, \omega_h, \omega_p\} \le \omega_{\text{cr}} \le \max\{\omega_\alpha, \omega_h, \omega_p\}, \\ 0.75\widehat{U}_{\text{cr}} \le U_{\text{cr}} \le 1.5\widehat{U}_{\text{cr}}. \end{cases} \end{aligned} \tag{9}$$

Equation (9) is the flutter optimization model for searching critical circular frequency and wind speed.

3. Solution Method of Flutter Optimization Model

In equation (3), parameter ω is semi-implicit, and U is implicit in A_i^*, P_i^*, and H_i^*. Therefore, the objective function of the optimization model equation (9) is the implicit function of $X = [\omega, U]$. For this optimization problem, the traditional optimization algorithm is not very useful because the gradient information at the search point is too difficult to be calculated. The firefly algorithm is fit for dealing with the optimization problems of the implicit objective function for its strong ability for nonlinear mapping. Considering that the traditional firefly algorithm depends on its control parameters, sometimes suffer from being trapped in local optima and low convergence speed in the later period, we adopted the quantum genetic firefly algorithm to solve the flutter optimization model to obtain critical circular frequency and critical wind speed.

3.1. Firefly Algorithm. FA is a relatively new method which is developed by Yang [7]. This approach is on the basis of the certain behavioral pattern, particularly the flashing characteristic of fireflies in the tropical summer sky. Fireflies are beetles members of the family Lampyridae. A firefly is a kind of insect that utilizes the principle of bioluminescence to attract mates or prey. The luminance produced by a firefly enables other fireflies to trail its path in search of their prey.

Some flashing characteristics of fireflies were idealized so as to develop a firefly algorithm. For simplicity, only three rules were followed [7]:

(1) All fireflies are assumed unisex so that one firefly will be attracted to other fireflies regardless.

(2) The attractiveness of one firefly to another is proportional to their brightness, which is declined with increase in the distance between them; consequently, the ones with less brightness will always move toward the ones with higher brightness. If there is no firefly brighter than a given firefly, it will move randomly.

(3) The brightness or light intensity of a firefly is affected or determined by the landscape of the objective function to be optimized. For a maximization problem, the brightness can be proportional to the value of the objective function. Other forms of brightness can be defined similar to the fitness function in genetic algorithms.

The brightness of the individual firefly is affected by the nature of the encoded cost function, simply say, the brightness of proportion to the value of the fitness or objective function. The major issues in FA development are the formulation of the objective function (attractiveness) and the variation of the light intensity. As an instance, in the optimal design problem involving the maximization of the objective function, the fitness function is proportional to the brightness or the amount of light emitted by the firefly. Therefore, decrement of the light intensity due to more distance between the fireflies will lead to variations of intensity and thereby lessen the attractiveness among them. Equation (10) can be used to represent the light intensity with varying distance:

$$I(r) = I_0 \exp\left(-\gamma \cdot r^2\right), \tag{10}$$

where I is the light intensity of the source at distance r from a firefly; I_0 represents the initial light intensity when $r = 0$; and γ is the light absorption coefficient, which characterizes the variation of attractiveness and influences the convergence speed and the overall behavior of FA. γ typically varies from 0.1 to 10 [8]. As a firefly's attractiveness is proportional to the light intensity observed by adjacent fireflies, we can represent the attractiveness β at a Cartesian distance r from the firefly as follows:

$$\beta = \beta_0 \exp\left(-\gamma \cdot r^2\right), \tag{11}$$

where β_0 is the attractiveness at distance $r = 0$. The light intensity I and attractiveness β are in the same way synonymous. While the intensity is referred to as an absolute measure of emitted light by the firefly, the attractiveness is a relative measure of the light that should be seen in the eyes of the beholders and judged by other fireflies. The distance between any two fireflies i and j at x_i and x_j can be the Cartesian distance:

$$r_{ij} = \left\| x_i - x_j \right\|. \tag{12}$$

The movement of firefly i as attracted to another brighter firefly j can be represented as follows:

$$\Delta x_i = \beta_0 e^{-\gamma \cdot r_{ij}^2}\left(x_j^t - x_i^t\right) + \alpha\left(N_{\text{rand}} - 0.5\right), \tag{13}$$

where t is the iteration. The first term that appeared in equation (13) is because of the attraction. The second term $\alpha(N_{\text{rand}} - 0.5)$ represents the randomization, and it can enlarge the search space. α is a randomization coefficient, and it is the random number vector derived from a Gaussian distribution $\alpha \in [0, 1]$. The value of N_{rand} is a random number generator uniformly distributed in $[0, 1]$. The next movement of firefly i is updated as follows:

$$x_i^{t+1} = x_i^t + \Delta x_i. \tag{14}$$

The movements of fireflies consist of three terms: the current position of ith firefly, attraction to another more attractive firefly, and a random walk that consists of a randomization parameter α and the random generated number from interval $[0, 1]$. When $\beta_0 = 0$, the movement depends on the random walk only. On the contrary, the parameter γ has a crucial impact on the convergence speed.

Apart from the first step, all performances are repeatedly carried out until the optimization process ends. The flowchart of FA is shown in Figure 1.

3.2. Quantum Genetic Firefly Algorithm. The quantum genetic algorithm (QGA) is the combination of the genetic algorithm (GA) and the quantum computing [9]. In quantum computing, the smallest unit of information in a two-state quantum computer is called a quantum bit. Quantum bits (Q-bits) replace general genes, and each individual is constructed with a string of Q-bits. The quantum gate (Q-gate) is used to update individuals. This algorithm has good stability robustness and extensive adaptability. The quantum genetic firefly algorithm (QGFA) is the combination of the firefly algorithm (FA) and the quantum genetic algorithm (QGA) (Algorithm 1).

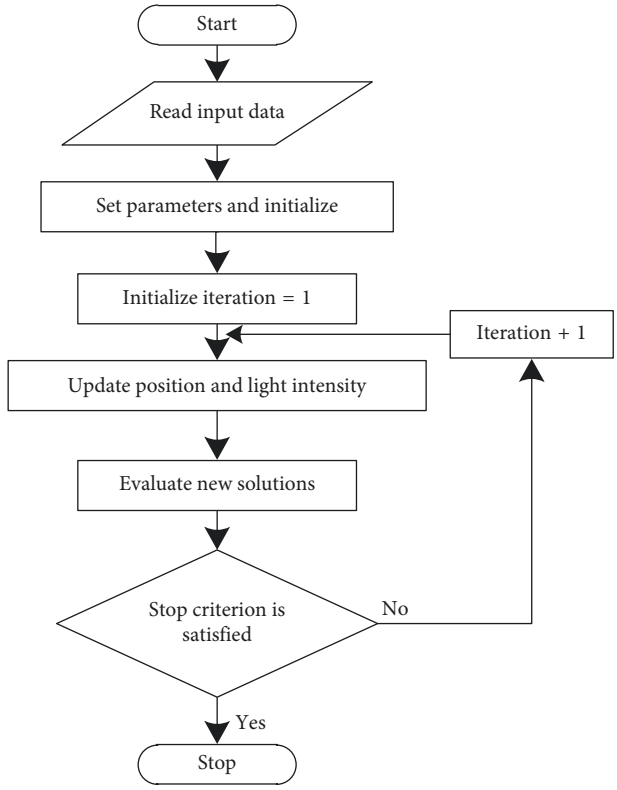

Figure 1: Flowchart of FA.

The details of the QGFA are as follows:

(1) Quantum representation of fireflies: the positions of the fireflies are represented by binary encoding based on the Q-bits. The Q-bits can be in a "0" state, a "1" state, or any linear superposition of the two. The state of a Q-bit can be represented as follows:

$$|\varphi\rangle = \cos\theta|0\rangle + \sin\theta|1\rangle, \tag{15}$$

where $\cos\theta$ and $\sin\theta$ represent state probability amplitudes, respectively. $\cos^2\theta$ is the probability that a Q-bit is observed as $|0\rangle$, and similarly, $\sin^2\theta$ is the probability that a Q-bit is observed as $|1\rangle$. The use of a Q-bit to represent an individual in this study was inspired by quantum computing concepts. QGFA operates on a population composed of multiple feasible solutions. Each feasible solution of the QGFA, which is made up of multiple Q-bits, is the element chromosome of the population. The advantage of the representation is the capability to use the linear superposition method to generate any possible solution. As described in equation (15), a Q-bit can be defined as $(\cos\theta, \sin\theta)^T$, where each Q-bit is regarded as a pair of coordinates. The quantum bits of the firefly i position can be represented as follows:

$$Qf_i = \begin{bmatrix} \cos\left(\theta_{i1}\right) \mid \cos\left(\theta_{i2}\right) \mid \cdots \mid \cos\left(\theta_{im}\right) \\ \sin\left(\theta_{i1}\right) \mid \sin\left(\theta_{i2}\right) \mid \cdots \mid \sin\left(\theta_{im}\right) \end{bmatrix}, \tag{16}$$

(1) Initialize all the population $Q(t)$ and $t = 0$
(2) Measure each individual of the initial population $Q(t)$ to obtain the binary solution $P(t)$
(3) Evaluate the binary solution $P(t)$ and calculate individual light intensity of $P(t)$
(4) Determine whether they meet the terminating conditions: yes, stop; no, continue
(5) $t = t + 1$; change the position of the populations by the strategy of quantum rotation gate equation (20) to obtain new populations $Q'(t)$
(6) Apply quantum crossover and mutation operation
(7) Calculate individual light intensity
(8) Go to (4)

Algorithm 1: Process of quantum genetic firefly algorithm (QGFA).

where $\theta_{ij} \in [0, \pi/2]$ is the rotation angle. In the initial search of the algorithm, all states appear with the same probability, so we set $\theta_{ij}^{t=0} = \pi/4$ and obtain

$$Qf_i^0 = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \cdots & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \cdots & \frac{1}{\sqrt{2}} \end{bmatrix}. \quad (17)$$

(2) Quantum movement according to the firefly algorithm strategy: the concept of the quantum rotation gate is introduced into the firefly algorithm, thus improving the global search capability of the firefly algorithm further. The group $Q(t)$ is updated by rotating the Q-bit toward the direction of the corresponding Q-bit to obtain a better value. The state probability amplitudes of the ith Q-bit are updated as follows:

$$\begin{bmatrix} \cos\chi \\ \sin\chi \end{bmatrix} = \begin{bmatrix} \cos(\Delta\theta_i) & -\sin(\Delta\theta_i) \\ \sin(\Delta\theta_i) & \cos(\Delta\theta_i) \end{bmatrix} \begin{bmatrix} \cos(\theta_i) \\ \sin(\theta_i) \end{bmatrix}. \quad (18)$$

The update strategy of the quantum rotation gate is defined as follows:

$$\theta_{ik}^{t+1} = \begin{cases} \theta_{ik}^t + \beta_0 e^{-\gamma r_{ij}^2}\left(\theta_{jk}^t - \theta_{ik}^t\right) + \alpha\left(N_{\text{rand}} - 0.5\right), & \text{if } B_{ik}^t \neq B_{ik}^t, \\ \theta_{ik}^t + \alpha\left(N_{\text{rand}} - 0.5\right), & \text{otherwise}, \end{cases} \quad (19)$$

where θ_{ik}^t is the rotation angle magnitude the ith firefly's kth Q-bit after the tth iteration and r_{ij} is the hamming distance [10]. B is the binary solution of the firefly algorithm.

(3) Introduced crossover and mutation: the goal of crossover and mutation is to acquire new information to maintain population diversity. Crossover is realized through exchanging part of the Q-bit encoding. Mutation is realized through the changing part of Q-bit encoding.

The flowchart of QGFA is shown in Figure 2.

According to operational processes above, QGFA is adopted to search the flutter critical point; key steps are explained as follows:

(1) Initialize all the population $Q(t)$ and $t = 0$. The positions of the fireflies are the solution vectors $[\omega, U]$ of the flutter optimization model.
(2) Select $f(X)$ to be the fitness function in equation (9) and calculate the light intensity of the population $Q(t)$.
(3) Update the population $Q(t)$ through equation (19).

3.3. Validation of QGFA. We applied a commonly employed test function and compared results to those obtained by QGFA. Shekel's Foxholes [11], introduced by Shekel, is a 2D function with 25 peaks with different heights, ranging from 476.19 to 499.00. The global optimum is located at (−32, −32). Shekel's Foxholes function is defined as follows:

$$Z(x, y) = 500 - \frac{1}{0.002 + \sum_{i=0}^{24} 1/\left(1 + i + (x - a(i))^6 + (y - b(i))^6\right)}, \quad -65.536 \leq x, y \leq 65.536, \quad (20)$$

where $a(i) = 16[(i \bmod e5) - 2]$ and $b(i) = 16[(i/5) - 2]$.

Due to high modality of Shekel's Foxholes function, it is a difficult task to determine its points of local and global maximum, making it a great challenge for optimization algorithms. The QGFA was applied for obtaining the maximized value of equation (20), by selecting 200 fireflies as the particle population and also α and γ parameters which was considered as 0.5 and 0.1, respectively. However, the results of the QGFA algorithm including 10 of best and worst maximum values with their iteration number are presented in Table 1. The global optimal obtained by QGFA along the number of iteration for the successive independent 10 runs.

As it can be seen from Table 1, results present that the QGFA can reach to the maximum value in all experiments. Indeed, using various iteration numbers, the QGFA successes to obtain the optimal solution and rejected local optimal values (local maximums).

4. Case Studies

To further analyze the quantum genetic firefly algorithm (QGFA) in the next subsections, a brief introduction to three cases studies is presented. Three cases studies are related to the ideal flat plate flutter analysis, the free vibration bridge flutter analysis, and the force vibration bridge flutter analysis. The optimization algorithms were implemented in the MATLAB (R2014a) program. The model was performed

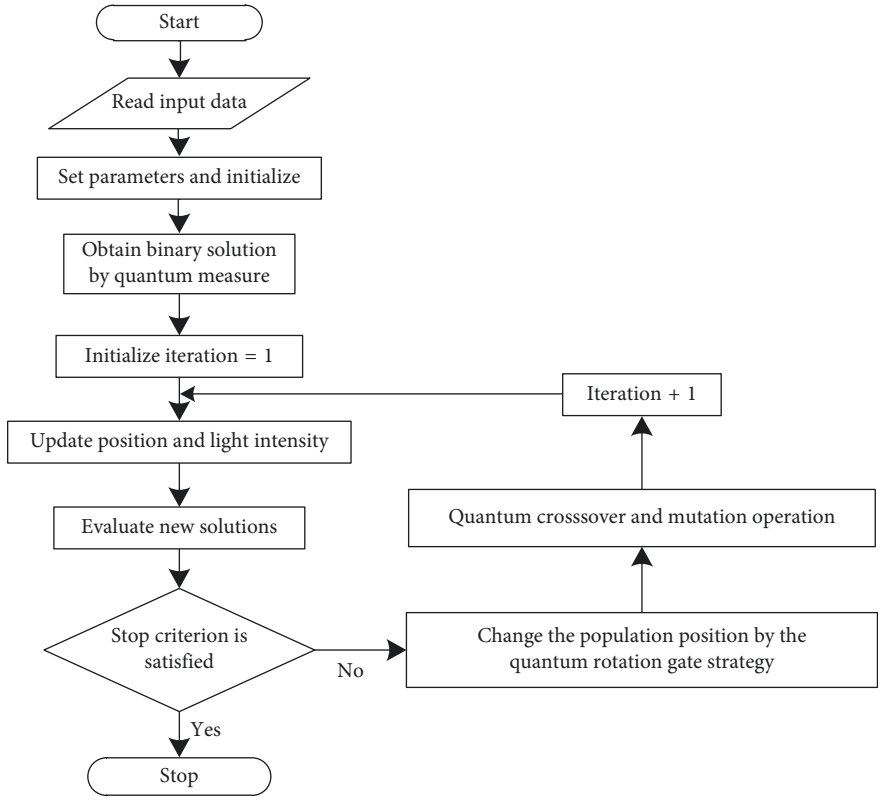

Figure 2: Flowchart of QGFA.

Table 1: QGFA optimization results for Shekel's Foxholes.

Experiment	Fitness	Iteration number
Best results		
1	499.01	21
2	499.01	32
3	499.01	24
4	499.01	13
5	499.01	12
6	499.01	14
7	499.01	25
8	499.01	22
9	499.01	25
10	499.01	21
Average	499.01	20.9
Worst results		
1	499.01	37
2	499.01	32
3	499.01	51
4	499.01	43
5	499.01	36
6	499.01	43
7	499.01	49
8	499.01	44
9	499.01	40
10	499.01	35
Average	499.01	41

using an AMD A8-4500M APU with the Radeon (tm) HD Graphics 1.90 GHz processor computer with 4 GB of RAM memory PC.

The parameters used in the cases of QGA, FA, and QGFA are listed in Table 2.

4.1. Case I: Flutter Analysis of a Flat Plate. Consider a uniform wind applied on a plate with a half-chord length $b = 0.225$ m, mass $m = 11.25$ kg/m, moment of inertia $I = 0.2828$ kg·m^2, vertical bending frequency $\omega_h = 12.11$ rad/s, torsional frequency $\omega_\alpha = 19.0$ rad/s, vertical bending damping ratio $\xi_h = 0.005$, and torsional damping ratio $\xi_\alpha = 0.008$. The aerodynamic derivatives are obtained from Theodorsen's theoretic solutions [14]. The critical circular frequency and critical wind speed are in the range of $\omega_{cr} \in [12, 19]$ and $U_{cr} \in [13, 28]$, respectively. The results that were obtained using QGA, FA, and QGFA in such a case are provided in Table 3.

Reference [15] obtained the calculation results: $\omega = 15.186$ rad/s and $U = 16.91$ m/s through the pursuit method. As shown in Table 3, the calculation results of QGFA are more agreement with [15], compared with FA and QGA. Based on the mean values, QGFA is the best in comparison with that of QGA and FA. For average CPU time, QGFA needs a little more time than that of FA and QGA.

4.2. Case II: Free Vibration Bridge Flutter Analysis

4.2.1. Experimental Work and Model Description. The shape of the deck is considered to represent a realistic highway bridge in China, but without barriers on the upper surface. The deck sectional length is 1.2 m. The geometry and

Table 2: FA, QGA, and QGFA parameter settings [12, 13].

Parameter	FA	QGA	QGFA
Population size	50 (75)	50 (75)	50 (75)
Max. iteration	100 (150)	100 (150)	100 (150)
Runs number	50	50	50
Crossover probability	—	—	0.2
Mutation probability	—	—	0.1
α	0.2	—	0.2
β_0	1	—	1
γ	0.1	—	0.1
Convergence indicator	10^{-5}	10^{-5}	10^{-5}

Note: () is for Case III.

Table 3: Comparison of computational results (flat plate).

Algorithm	Average CPU time (s)	Critical frequency (rad/s)		Critical wind speed (m/s)	
		Mean	Standard deviation	Mean	Standard deviation
QGA	5.6092	15.16	0.078	16.30	0.065
FA	5.5180	15.17	0.084	16.73	0.077
QGFA	5.7575	15.18	0.050	16.89	0.053

dimension of the model are shown in Figure 3. The experimental work was performed at Harbin Institute of Technology, Heilongjiang Province, Harbin, China [16]. The wind tunnel is a closed-circuit tunnel having a rectangular section with a width of 4 m wide, high of 3 m, and length of 25 m. The wind velocity (U) continuously varies from 2 m/s to 45 m/s. The longitudinal turbulence intensity is less than 0.46%. The tested model was fixed by a free vibration device, and the geometric blockage ratio is approximately equal to 2%, as shown in Figure 4.

4.2.2. Measurement Uncertainty. The uncertainty of the Re number (u_{Re}) can be written as follows [17]:

$$u_{\mathrm{Re}}^2 = \left(\frac{U}{\mathrm{Re}}\frac{\partial \mathrm{Re}}{\partial U}u_U\right)^2 + \left(\frac{\nu}{\mathrm{Re}}\frac{\partial \mathrm{Re}}{\partial \nu}u_\nu\right)^2 + \left(\frac{H}{\mathrm{Re}}\frac{\partial \mathrm{Re}}{\partial H}u_H\right)^2, \tag{21}$$

where u is the uncertainty, ν is the kinematic viscosity, and H is the height of the bridge model. The uncertain free stream velocity u_U was about 1.5%. The uncertain model size u_H was 7%. The uncertain kinematic viscosity of the air u_H was 4%. As a result, the calculated U_{Re} give in equation (21) was 2.9%. The uncertainty of the accelerometer was obtained from the calibration chart for the type 4507B accelerometer. The expanded uncertainly was 1.0%, as determined in accordance with [18].

4.2.3. Experiment Results and Analysis. In the experiments, the wind speed ranged from 4 m/s to 16 m/s. The correspondingly reduced wind speed (Ur) was U/fB, in which U, f, and B were defined in the previous subsections. The MITD method was adopted herein to identify the flutter derivatives [19]. The aerodynamic derivatives of the bridge are provided in Figure 5. The range of $\omega_{\mathrm{cr}} \in [0.6, 1.3]$ and $U_{\mathrm{cr}} \in [60, 120]$ was considered in the experiment.

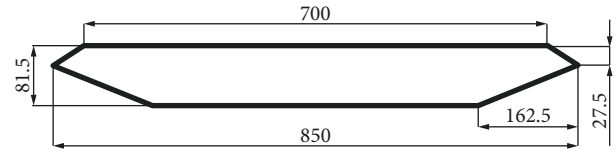

Figure 3: Deck section dimensions and shape (unit: mm).

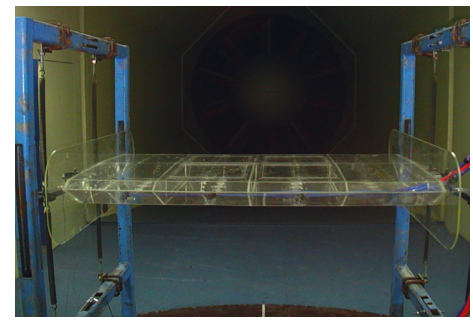

Figure 4: The tested model and the free vibration device.

The results obtained through the traditional method (Scanlan method) [20], QGA, FA, and QGFA for the Case II are presented in Table 4.

As it can be seen from Table 4, the mean of the three algorithms is almost equivalent. The standard deviation of the QGFA is the lowest of the three algorithms. The average CPU time of QGFA, however, is a little bit longer than QGA and FA. Table 5 provides comparison of the critical wind speed obtained from QGFA with that of other papers.

From Tables 4 and 5, we can observe that, based on average values, QGFA is the best in comparison with that of QGA and FA. The results of QGFA are very similar with the others, and it shows that the results of the QGFA can be accepted. In [24], the damping rate is 0.7%, so the critical wind velocity is a little higher than that of QGFA.

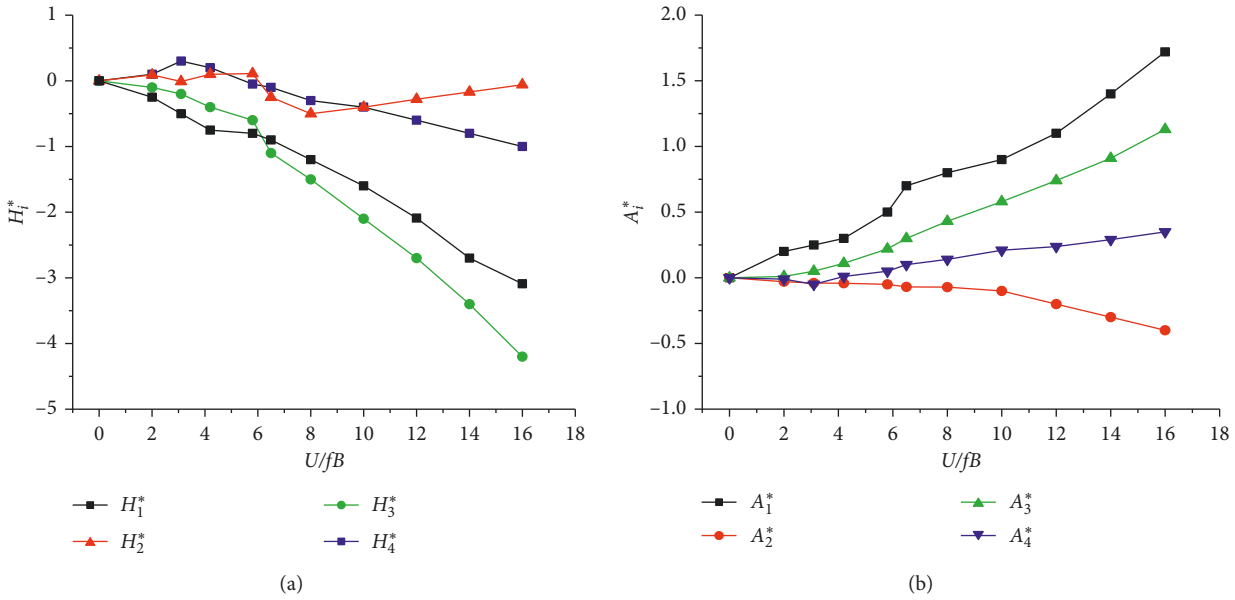

Figure 5: Flutter derivatives in Case II.

Table 4: Comparison of computational results (Case II).

Algorithm	Average CPU time (s)	Critical frequency (rad/s)		Critical wind speed (m/s)	
		Mean	Standard deviation	Mean	Standard deviation
Scanlan		1.39	—	86.9	—
QGA	57.0438	1.32	0.035	86.6	1.095
FA	55.1897	1.38	0.043	86.1	1.101
QGFA	57.6702	1.32	0.028	88.1	1.066

Table 5: Critical wind velocity U_{cr} (m/s) (Case II).

Parameter	QGFA	Hua simulation [21]	Xin simulation [22]	Huang simulation [23]	Gu experiment [24]
U_{cr}	88.1	88.9	94.1	88	96

4.3. Case III: Forced Vibration Bridge Flutter Analysis. A European bridge section model is adopted for this case. The size of the model is shown in Figure 6. In the test, no appendages are considered. The forced vibration method [25] is adopted in the wind tunnel experiment, as shown in Figure 7. MITD method was still adopted to identify the flutter derivatives. The aerodynamic derivatives of the bridge are summarized in Figure 8. The search scope is $\omega_{cr} \in [0.5, 1.2]$ and $U_{cr} \in [35, 80]$.

The results obtained by tthe raditional method (Scanlan method), QGA, FA, and QGFA to Case III are presented in Table 6.

As it can be seen from Table 6, on average CPU time, QGFA needs a little more time. However, QGFA has better numerical stability. The standard deviation of the QGFA is the lowest of the three algorithms. The result of QGFA and other people's results are listed in Table 7.

In Table 7, the barriers are considered in the wind tunnel experiment of Poulsen et al. [26], so the critical wind speed is less than those of the others. From Tables 6 and 7, once again, we can observe that, based on average values, QGFA is the best in comparison with that of QGA and FA. The results of QGFA are very similar with the other references in Table 7, and it shows that the results of the QGFA can be accepted.

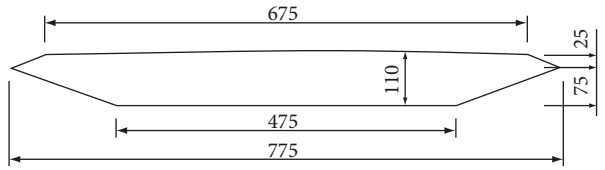

Figure 6: The size of the model in the forced vibration tests.

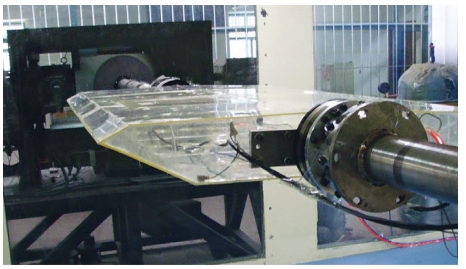

Figure 7: Force vibration system.

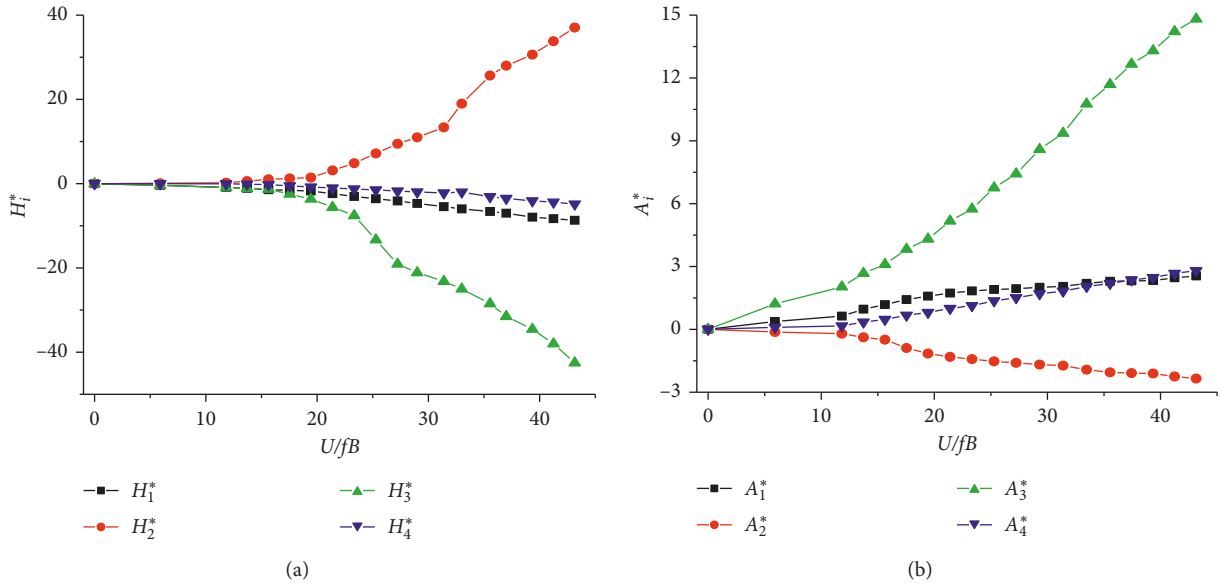

Figure 8: Flutter derivatives in Case III.

Table 6: Comparison of computational results (Case III).

Algorithm	Average CPU time (s)	Critical frequency (rad/s)		Critical wind speed (m/s)	
		Mean	Standard deviation	Mean	Standard deviation
Scanlan		1.05	—	41.7	—
QGA	148.0438	1.00	0.041	41.0	0.14
FA	137.2471	1.07	0.044	40.8	0.16
QGFA	170.6702	1.00	0.035	40.6	0.13

Table 7: Critical wind velocity and critical frequency (Case III).

Parameters	QGFA	Poulsen experiment [26]	Discrete vortex method [27]	Zhu simulation [28]	Cui simulation [29]
U_{cr} (m/s)	40.6	36	37.6	40.5	38.0
ω (Hz)	0.162	0.163	0.165	0.160	0.160

5. Conclusions

In this work, the firefly algorithm (FA) is improved by quantum theory integration and is used as an optimization tool for bridge flutter analysis. The proposed algorithm is based on the integration into the firefly algorithm of the quantum genetic algorithm (QGA). The results obtained by using the quantum genetic firefly algorithm (QGFA) are compared with FA and QGA for validation.

(1) The population of the firefly algorithm is quantized through the binary encoding format. We define update strategy of the quantum rotation gate, get a kind of the hybrid firefly algorithm, and give the steps of searching the optimal solution of the flutter optimization model with the hybrid firefly algorithm. The update strategy of the quantum rotation gate is defined to update the population position, and the quantum crossover and mutation operation are introduced to improve the global search capability of the firefly algorithm.

(2) Use the quantum genetic firefly algorithm, firefly algorithm, and quantum genetic algorithm to analyze three cases. The results show that the quantum genetic firefly algorithm has better optimization results and stability. The CPU time of the QGFA is more than FA and QGA.

(3) The entire computing process of the quantum genetic firefly algorithm is without manual intervention. The algorithm has good numerical stability, high precision, and good practicality for the bridge critical state search.

From the results obtained, we can conclude that the QGFA is an efficient, reliable, and robust method, which can be applied successfully to bridge flutter analysis.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

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

Acknowledgments

The authors extend their deepest gratitude to Mr. Isleem Haytham and Miss Cho Mya Darli for their comments on this paper. Special thanks are conveyed to BIAN Xiao-xian and Engineer ZHAO Peng for their help during the preparation of the tests and for their efforts during the tests. This work was supported by the National Natural Science Foundation (41672287).

References

[1] N. Li, M. J. Balas, P. Nikoueeyan, H. Yang, and J. W. Naughton, "Stall flutter control of a smart blade section undergoing asymmetric limit oscillations," *Shock and Vibration*, vol. 2016, no. 2, Article ID 5096128, 14 pages, 2016.

[2] Y. Gu and Z. Yang, "Modified p-k method for flutter solution with damping iteration," *AIAA Journal*, vol. 50, no. 2, pp. 507–510, 2015.

[3] H. E. Lee, T. V. Vu, S. Y. Yoo, and H. Y. Lee, "A simplified evaluation in critical frequency and wind speed to bridge deck flutter," *Procedia Engineering*, vol. 14, no. 12, pp. 1784–1790, 2011.

[4] L. D. S. Coelho and V. C. Mariani, "Improved firefly algorithm approach applied to chiller loading for energy conservation," *Energy and Buildings*, vol. 59, no. 2, pp. 273–278, 2013.

[5] M. S. Andersen, J. Johansson, A. Brandt, and S. O. Hansen, "Aerodynamic stability of long span suspension bridges with low torsional natural frequencies," *Engineering Structures*, vol. 120, pp. 82–91, 2016.

[6] V. D. Put, "Rigidity of structures against aerodynamic forces," *International Association for Bridge and Structural Engineering*, vol. 36, no. 1, pp. 189–196, 1976.

[7] M. B. Imamoglu, M. Ulutas, and G. Ulutas, "A new reversible database watermarking approach with firefly optimization algorithm," *Mathematical Problems in Engineering*, vol. 2017, no. 2, Article ID 1387375, 14 pages, 2017.

[8] N. Poursalehi, A. Zolfaghari, A. Minuchehr, and H. K. Moghaddam, "Continuous firefly algorithm applied to PWR core pattern enhancement," *Nuclear Engineering and Design*, vol. 258, pp. 107–115, 2013.

[9] S. Prakash and D. P. Vidyarthi, "A novel scheduling model for computational grid using quantum genetic algorithm," *Journal of Supercomputing*, vol. 65, no. 2, pp. 742–770, 2013.

[10] R. Yue, X. Xue, H. Liu, J. Tan, and X. Li, "Quantum algorithm for K-nearest neighbors classification based on the metric of hamming distance," *International Journal of Theoretical Physics*, vol. 56, no. 11, pp. 3496–3507, 2017.

[11] J. Shekel, "Test functions for multimodal search techniques," in *Proceeding of the Fifth Princeton Conference on Information Science and System*, pp. 321–333, Princeton, NJ, USA, March 1971.

[12] S. E. K. Fateen, A. Bonilla-Petriciolet, and G. P. Rangaiah, "Evaluation of covariance matrix adaptation evolution strategy, shuffled complex evolution and firefly algorithms for phase stability, phase equilibrium and chemical equilibrium problems," *Chemical Engineering Research and Design*, vol. 90, no. 12, pp. 2051–2071, 2012.

[13] J. Senthilnath, S. N. Omkar, and V. Mani, "Clustering using firefly algorithm: performance study," *Swarm and Evolutionary Computation*, vol. 1, no. 3, pp. 164–171, 2011.

[14] T. Theodorsen, "General theory of aerodynamic instability and the mechanism of flutter," Technical Report Archive & Image Library, University of Arizona Libraries Main Library, Tucson, AZ, USA, 1935.

[15] F. Y. Xu, A. R. Chen, and D. L. Wang, "Pursuit method for searching critical flutter wind velocity of flat plates," *Engineering Mechanics*, vol. 22, no. 5, pp. 48–53, 2005, in Chinese.

[16] S. B. Tao, A. P. Tang, D. B. Xin, K. T. Liu, and H. F. Zhang, "Vortex-induced vibration suppression of a circular cylinder with vortex generators," *Shock and Vibration*, vol. 2016, Article ID 5298687, 10 pages, 2016.

[17] B. Çuhadaroğlu, Y. E. Akansu, and A. Ö. Turhal, "An experimental study on the effects of uniform injection through one perforated surface of a square cylinder on some aerodynamic parameters," *Experimental Thermal and Fluid Science*, vol. 31, no. 8, pp. 909–915, 2007.

[18] W. Kessel, "European and international standards for statements of uncertainty," *Engineering Science & Education Journal*, vol. 7, no. 5, pp. 201–207, 1998.

[19] C. Liu and J. Teng, "Modal analysis of a cable-stayed bridge model using a modified ibrahim time domain algorithm," *Journal of Vibroengineering*, vol. 16, no. 6, pp. 3033–3044, 2014.

[20] R. H. Scanlan, "Airfoil and bridge deck flutter derivatives," *Journal of ASCE*, vol. 6, no. 6, pp. 1717–1737, 1971.

[21] X. Hua and Z. Chen, "Wind-induced flutter analysis of long-span bridges with fully automatic," *Engineering Mechanics*, vol. 19, no. 2, pp. 68–72, 2002.

[22] D. B. Xin, J. P. Ou, and H. Li, "Suppression method for wind-induced flutter of long-span bridge based on steady air-suction," *Journal of Jilin University*, vol. 41, no. 5, 2011.

[23] H. Huang, M. Li, and D. He, "Aerodynamic design method based on control theory and Navier-Stokes equations," *Acta Aerodynamica Sinica*, vol. 23, no. 1, pp. 10–15, 2005.

[24] M. Gu, W. Wu, and H. Xiang, "Testing study on flutter control of long-span bridges by using tuned mass dampers," *Journal of Tongji University*, vol. 24, no. 2, pp. 124–129, 1996.

[25] F. Y. Xu, X. Y. Ying, and Z. Zhang, "Three degree of freedom coupled numerical technique for extracting 18 aerodynamic derivatives of bridge decks," *Journal of Structural Engineering*, vol. 140, no. 11, article 04014085, 2014.

[26] N. K. Poulsen, A. Damsgaard, and T. A. Reinhold, "Determination of flutter derivatives for the great belt bridge," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 41, no. 1–3, pp. 153–164, 1992.

[27] A. Larsen and J. H. Walther, "Discrete vortex simulation of flow around five generic bridge deck sections," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 77-78, pp. 591–602, 1998.

[28] Z. Zhu and Z. Chen, "Numerical simulations for aerodynamic derivaives and critical flutter velocity of bridge deck," *China Journal of Highway and Transport*, vol. 17, no. 3, pp. 41–45, 2004.

[29] Y. Cui and G. Chen, "Simulations for identification of flutter derivatives of bridge section using the coupled-forced-vibration method," *Journal of Vibration Engineering*, vol. 20, no. 1, pp. 35–39, 2007.