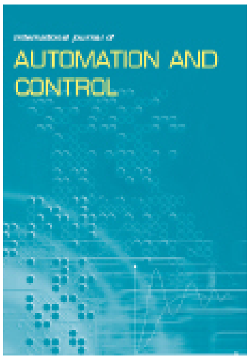



**International Journal of Automation and Control**

ISSN online: 1740-7524 - ISSN print: 1740-7516
https://www.inderscience.com/ijaac

# Observer controller-based structure for a modified flower pollination algorithm for wind power generation

Amoh Mensah Akwasi, Xie Wei, Otuo-Acheampong Duku

**DOI:** 10.1504/IJAAC.2024.10054838

**Article History:**

| | |
|---|---|
| Received: | 01 December 2022 |
| Last revised: | 27 January 2023 |
| Accepted: | 30 January 2023 |
| Published online: | 30 November 2023 |

# Observer controller-based structure for a modified flower pollination algorithm for wind power generation

Amoh Mensah Akwasi and Xie Wei*

School of Automation Science and Engineering,
South China University of Technology,
Wushan, Tianhe District, Guangzhou, 510641, China
Email: amom74650@gmail.com
Email: weixie@scut.edu.cn
*Corresponding author

Otuo-Acheampong Duku

School of Electrical Engineering,
Wuhan University,
Luojia Hill, Wuchang District,
Wuhan, 430072, China
Email: otuoduku@whu.edu.cn

**Abstract:** A proposed two-stage Luenberger observer controller-based structure is used by the pollinators in the flower pollination algorithm (FPA) to search for global and local pollination. The FPA moves in only one direction in search of the best solution and has a slow speed which is unable to properly find the best solution at longer distances. The modified FPA with the Luenberger method observes the best and optimal proliferation of specific flower species based on the combination of global and local pollination, which is used in the search for the best solution by the pollinators in both forward and backward movement, and with fast speed in the search for the local and global best solutions at both shorter and longer distances, which measure the input and output power generated by the wind turbine. The output MATLAB/Simulink simulation results show the effectiveness and high performance of the proposed method.

**Keywords:** wind power generation; Luenberger observer controller; modified flower pollination algorithm; global and local pollination; forward and backward pollination.

**Reference** to this paper should be made as follows: Akwasi, A.M., Wei, X. and Duku, O-A. (2024) 'Observer controller-based structure for a modified flower pollination algorithm for wind power generation', *Int. J. Automation and Control*, Vol. 18, No. 1, pp.53–86.

**Biographical notes:** Amoh Mensah Akwasi received his Bachelor's degree in Electrical Engineering and Automation from China Three Gorges University, China in 2021. He is currently pursuing his Master's degree in Electrical and Computer Engineering at South China University of Technology, China. He

worked as an Assistant Electrical Engineer at Guangdong Jinlong Company Limited in Dongguan, China. His research interest includes power system and its automation.

Xie Wei received his BEng and MEng degrees with the Automation Department, Wuhan University of Science and Technology, China, and the PhD degree from the Kitami Institute of Technology, Japan. He worked as a Postdoctoral Researcher with the Satellite Venture Business Laboratory from 2003 to 2006. In 2006, he joined the School of Automation Science and Engineering, South China University of Technology, as an Associate Professor, where he was promoted to Full Professor, in 2010. His research interests include intelligent robot, control theory and application. He has published more than 80 articles in international journals and conferences.

Otuo-Acheampong Duku received his Bachelor's degree in Electrical Engineering and Automation from China Three Gorges University, and now studying at Wuhan University, China for his Master's degree. He worked as an engineer at Centre for Plant Medicine Research. His fields of interest include voltage stability and FACTS device.

## 1 Introduction

Due to the high increase in power demand, the use of renewable energy to generate power has been adopted over the years. There are several renewable energies, but solar and wind energy are used mostly. These two renewable energies are used mostly because they are environmentally friendly and cost-effective since it's easy to be implemented (Faccio et al., 2018; Ma et al., 2014). Wind energy plays a vital role in power generation due to its availability both during the day and at night. Wind energy conversion involves the change from kinetic energy to electrical energy. The wind turbines installation has multiplied and the penetration level of wind generation on the grid has increased rapidly. A wind turbine generates stabilised output power during normal operations. Power fluctuations occur mainly as a result caused by the effects such as wind shear, wind turbine tower shadow, turbulence, and the power system control.

For the wind-generated being disturbed by the distribution network to be reduced, different induction generators such as permanent magnet synchronous generator (PMSG), doubly-fed induction generator, squirrel-cage induction generator, and wound-rotor induction generator (Eduard et al., 2016; Mittal et al., 2009; Syahputra et al., 2017) has been implemented. These generators can be connected directly to the grid. A PMSG is the most commonly used induction generator since its cost-effective, has low maintenance, has higher efficiency, and also can be used without the need for a gearbox (Alejandro et al., 2009). Different authors have proposed the use of wind turbines connected with different induction generators. For the wind turbine's speed to be controlled, a wind turbine-driven PMSG connected to the grid power system is proposed (Mohd et al., 2015). Basic circuit equations were used to model and analyse a standalone wind power system. The wind turbine's speed is controlled using PMSG based on the variable speed. AC/DC and DC/DC converters were used where the output AC voltage was increased using the AC/DC converter and the maximum power output of the wind system was controlled using the DC/DC converter (Mohammed and Abu, (2014). The proposed work by the researchers' output results was achieved but had some shortcomings which include low tracking speed, low power generation, power instability and fluctuations, and inconsistent tracking due to changes in the weather conditions.

The output of the wind turbine is nonlinear which decreases the output power generated and is also affected by reducing the rotation speed of the wind turbine due to the effects of power fluctuations. Perturb and observe (P&O) algorithm, incremental conductance (INC) algorithm, genetic algorithm (GA), particle swarm optimisation (PSO) algorithm, artificial neural network (ANN) algorithm, artificial bee colony (ABC) algorithm (Amoh and Xie, 2022; Amoh et al., 2022; Chen et al., 2012; Chiranjit et al., 2020; Dinesh et al., 2007a, 2007b; Duku et al., 2022; Ellaia et al., 2012; Gaurav et al., 2023; Hadjer et al., 2019, Hazem et al., 2022; Jing and Shi, 2022; Jinqiang et al., 2020; Ju and Li, 2019; Naidu et al., 2017; Raj and Kumar, 2018; Song et al., 2018; Yu and Lingjuan, 2022) has been proposed to increase the rotating speed of the wind turbine to extract more power and also decreases power fluctuation by stabilising the output power of the wind turbine. The flower pollination algorithm (FPA) performs better than other optimisation algorithms (Yang, 2012; Zaid et al., 2018; Mohammed et al., 2019; Nayak et al., 2018). Ahmed et al. (2018) carried out a research using three optimisation algorithms which are the flower pollination algorithm (FPA), GA, and PSO on the advantages of the gain and time optimisation of controller-based double-feed induction generators based on the wind power system. The results of the research carried out show that the FPA performed better than the other two optimisation algorithms but there was a delay in the global search time of the global best. Qu et al. (2017) proposed a method where optimal weight coefficients were searched using two-hybrid optimisation algorithms which are the flower pollination algorithm and a bat algorithm to investigate the potential application of a multi-objective optimisation algorithm for wind speed forecasting optimisation. The results show that the flower pollination algorithm's time for searching was faster than the bat algorithm but the tracking speed in such for the global best was slow. The parameters used were few which were only suitable in search for the local pollination.

The FPA is a swarm-based evolutionary algorithm inspired by the biological evolution of the pollination of flowers (Mohamed and Ibrahim, 2015a; Yang, 2012). The pollination is based on global and local pollination to search for the best solution for each generation. The FPA has been modified in this paper.

The flower pollination algorithm proposed by Yang (2012) only enlightens on the next and current position in search of the local and global best solution. However, the speed at which the pollinators moved was not enlightened. The FPA proposed by Yang (2012) moves in only one direction in search of the best solution and has a slow speed which is unable to properly find the best solution at longer distances. The speed is added in this paper and also the movement of pollinators in search for the local and global best is improved by forward and backward based by using the Luenberger observer controller-based method. When the pollinators search for the local and global best solution, they later go back to the previous position to search for the next local and global best position by transfer of pollen which increases the rate at which more local and global best solutions can be found. The time interval for the next position is detected and increased. For the global best solution to be searched faster, two-step sizes were added with the number of iterations, and levy flight is used to imitate the random movement of pollinators. These two-step sizes find the directions of the global best from the current position and the global best solution is attained at a longer distance. For the local best solution to be searched faster, two randomly selected variables are used and the iteration numbers are increased for the local best solution to be attained at a shorter distance.

The Luenberger observer controller-based method (Luenberger, 1971) is added whereby the forward and backward movement of the pollinators search for the local and global best solution. The Luenberger observer controller is a two-stage designed structure that attains the forward movement of the pollinators' search for the local and global best solution and also attains the backward movement of the pollinators' search for the local and global best solution. The interval estimation error for the position of the vectors in the forward and backward movement for both global and local pollination is achieved.

This modified FPA movement enhances and improves the wind turbine's rotation speed and also stabilises the output of the wind turbine, thereby causing more energy to be received by the wind turbine and more power to be extracted from the wind turbine to be supplied to the grid for use. The modified FPA also reduces power fluctuations. A control system was designed and implemented with the modified FPA. This control design is integrated from the energy-receiving stage to the power-generation stage. The wind turbine is modelled to receive energy from the wind. An induction generator is connected to the wind turbine. AC-DC and DC-AC converters are used to enhance the output voltage and current generated. The AC-DC converter controls the torque of the generator to set the generator's operation point strategy and the DC-AC converter maximises the output voltage generated. The induction generator enhances the rotational speed of the wind turbine. The design was drawn and also implemented in MATLAB/Simulink software. The PSMG was used as the induction generator during the simulation process. The control system was designed with and without the modified FPA which serves as the maximum power point tracker (MPPT). The block diagram for the wind turbine system, control diagram for the generator, and mechanical torque was drawn. The output results of the control system were compared with the INC algorithm, GA, PSO algorithm, ABC algorithm, FPA, and the modified FPA. The wind power generation system tests for these algorithms was carried out with wind speed of 2 m/s, 4 m/s, 8 m/s, 10 m/s and 12 m/s with tracking time of 10 s, and generator speed of 1 pu at 0 deg pitch angle respectively.

The modified FPA dataset used for the evaluation was set. The minimum and maximum iteration number used was 50 and 100, respectively of which 1 was added for the next position to be achieved. The fitness was attained with the randomly selected values of 0.5 and 0.9. Two randomly selected solutions from the current pollinators were set with the values of 5 and 10 as the minimum and maximum values, respectively. The initial position is set as 0.4 for the local pollinators and 0.8 for the global pollinators. The zeros of the local pollinators are 0 and 1 and the zeros for the global pollinators are 1 and 5.

The sections of this paper are as follows: Section 2 discusses the modelling of a wind turbine. Section 3 describes wind power conversion. Section 4 introduces the flower pollination algorithm and its modification. Section 5 presents the observer controller-based flower pollination algorithm. The simulation results and discussion is presented in Section 6 and Section 7 is the conclusion.

## 2 Wind turbine model

A wind turbine converts the kinetic energy of the wind received to mechanical energy connected to the low shaft speed to rotate the rotor blades. The mechanical energy changes the low shaft speed to a high shaft speed by the use of the gearbox to generate electrical energy in the induction generator (Saberi et al., 2020). The wind turbine consists of the power generated from the wind $P_w$ and the mechanical torque $T_m$ to obtain a modelled wind turbine equation. The power generated by the wind $P_w$ can be expressed mathematically as:

$$P_w = \frac{1}{2}\rho\pi R^2 C_P(\lambda, \beta)V_w^3 \tag{1}$$

where $\rho$ is the density of air, $R$ is the radius of the wind turbine, $\lambda$ is the ratio of the tip speed of the wind turbine, $\beta$ is the pitch angle of the blade of the wind turbine, $C_p$ is the coefficient power, $V_w$ is the speed of the wind.

The angular speed of the wind turbine is given as:

$$\omega_w = \frac{V_w \lambda}{R} \tag{2}$$

The ratio of the tip speed of the wind turbine is given as:

$$\lambda = \frac{\omega_w R}{V_w} \tag{3}$$

where $\omega_w$ is the angular speed of the wind turbine. The mechanical torque $T_m$ can be expressed mathematically as:

$$T_m = \frac{P_w}{\omega_w} \tag{4}$$

Put equations (1) and (2) into equation (4). The mechanical torque can be expressed as:

$$T_m = \frac{1}{2}\rho\pi R^3 C_P(\lambda, \beta)V_w^2 \tag{5}$$

## 3 Observer control-based wind energy conversion system

### *3.1 Optimisation objective*

The output of the wind turbine is nonlinear which decreases the output power generated and is also affected by reducing the rotation speed of the wind turbine due to the effects of power fluctuations. The wind turbine in this paper is connected to the modified FPA which serves as the MPPT. This optimisation method tracks maximum power under different weather conditions. This modified FPA movement enhances and improves the wind turbine's rotation speed and also stabilises the output of the wind turbine, thereby causing more energy to be received by the wind turbine and more power to be extracted from the wind turbine to be supplied to the grid for use. The modified FPA also reduces power fluctuations.

### 3.2 Wind energy conversion system with the observer controller

An induction generator, an AC-DC converter, and a DC-AC converter circuit were added to the control system. The AC-DC converter controls the torque of the generator to set the generator's operation point strategy and the DC-AC converter maximises the output voltage generated. The induction generator enhances the rotational speed of the wind turbine. MPPT based on the observer controller flower pollination algorithm is proposed to extract more power by increasing the rotation speed of the wind turbine. These components form the wind power generation system. Figures 1 and 2 show the wind power generation system without and with the modified FPA-MPPT. Figures 3(a) and 3(b) show the control diagram for the generator torque and mechanical torque respectively. Figure 4 shows the block diagram of the proposed wind turbine system which was drawn based on the observer controller method.

**Figure 1** Wind power generation system diagram without modified FPA-MPPT

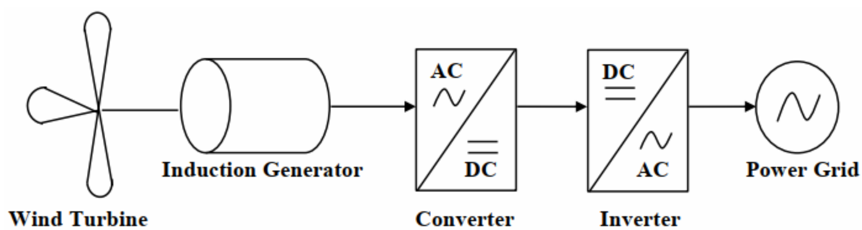
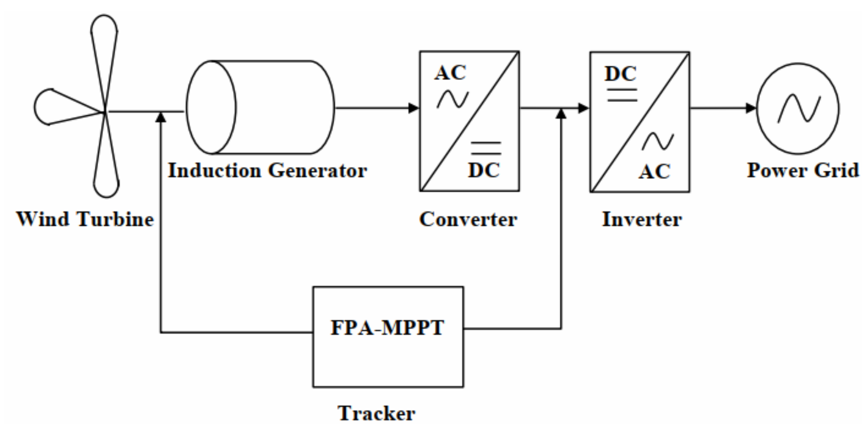



**Figure 2** Wind power generation system diagram with modified FPA-MPPT




The dynamic of the speed of the wind turbine is obtained as:

$$J_w \omega_w = T_w - T_m \tag{6}$$

where $J_w$ is the moment of inertia of the wind turbine, $\omega_w$ is the angular shaft speed of the wind turbine, $T_m$ is the mechanical torque, and $T_w$ is the wind torque.

The dynamic of the speed of the generator is obtained as:

$$J_g \omega_g = T_g - T_m \tag{7}$$

where $J_g$ is the moment of inertia of the generation, $\omega_g$ is the angular shaft speed of the generator, $T_m$ is the mechanical torque, and $T_g$ is the generator torque.

The wind torque can be deduced as:

$$T_w = G_s\left(\omega_{w1} - \omega_{g1}\right) + K_s\left(\omega_{w2} - \omega_{g2}\right) \tag{8}$$

where $G_s$ and $K_s$ are the fixed and friction coefficient of the torque respectively. $\omega_{w1}$ and $\omega_{w2}$ are the angular shaft speed of the fixed and friction coefficient of the torque of the wind turbine and $\omega_{g1}$ and $\omega_{g2}$ are the angular shaft speed of the fixed and friction coefficient of the torque of the generator.

The generator torque can be deduced as:

$$T_g = T_{g1}\omega_g - T_{g2}\omega_w \tag{9}$$

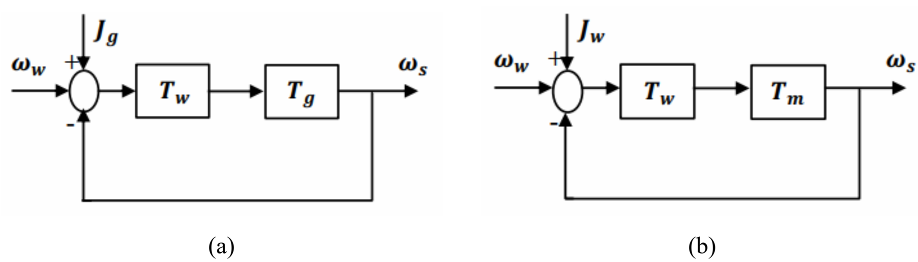
where $T_{g1}$ and $T_{g2}$ are the fixed and friction coefficient of the torque.

Wind speed error is calculated as:

$$e = \omega_s - \omega_g \tag{10}$$

where $\omega_s$ is the angular wind speed and $\omega_g$ is the angular generator speed

**Figure 3** (a) Control diagram for generator torque (b) Control diagram for mechanical torque




The equation given in (6)–(8) can be deduced using the Luenberger observer controller method:

$$\dot{q} = Aq + Bq + Cy \tag{11}$$

where $\dot{q}$ is the Luenberger observer, $q$ and $y$ are the input and output vectors, $A$, $B$ and $C$ are the stability matrix.

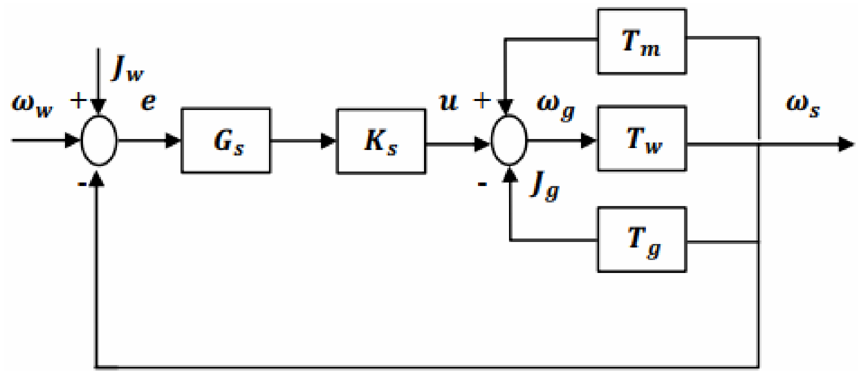
Equations (12), (13), and (14) deduce the stability matrix of $A$, $B$ and $C$.

$$A = \begin{bmatrix} 0 & 0 & -\dfrac{1}{J_w} \\ 0 & -\dfrac{T_{g1}}{J_g} & \dfrac{1}{J_g} \\ G_s & \left(\dfrac{K_s T_{g1}}{J_g} - G_s\right) & -K_s\left(\dfrac{1}{J_w} + \dfrac{1}{J_g}\right) \end{bmatrix} \tag{12}$$

$$B^T = \begin{bmatrix} 0 & -\dfrac{T_{g2}}{J_g} & \dfrac{K_s T_{g2}}{J_g} \end{bmatrix} \tag{13}$$

$$C^T = \begin{bmatrix} \frac{T_g}{J_w} & 0 & \frac{K_s T_g}{J_w} \end{bmatrix} \tag{14}$$

**Figure 4** Block diagram of a wind turbine system based on the observer controller method




## 4 Flower pollination algorithm

The flower pollination algorithm was proposed by Yang (2012). Since most plants are blooming species, their proliferation is transpired through pollination. Pollination is the transfer of pollen from one flower to a similar or different flower (Mohammed et al., 2019; Zaid et al., 2018). Pollinators such as insects, birds, humans, rodents, and all types of animals transfer the pollen (Mohamed and Ibrahim, 2015b). The transfer of pollen takes place from the anther of a flower to the stigma of a flower. Self-pollination and cross-pollination are the two types of pollination. Self-pollination is the transfer of pollen from the anther to the stigma of the same flower. Self-pollinating flowers can fertilise themselves. Cross-pollination is the process by which pollen is transferred from the anther to the stigma of another flower. Cross-pollinating flowers use vectors such as pollinators or wind to transfer pollen from one flower to another of the same species. Flower pollination is observed with the fitness and optimal proliferation of a specific plant species. Most insects like bees consist of Levy flight characteristics with flight distance steps that act by the Levy distribution. The flower pollination process mostly involves insects and birds whereby they fly or move to particular species of flowers only, which is also called flower constancy.

These insects and birds receive food from the flowers. Flower constancy maximises proliferation by increasing the number of particular kinds of flowers in the pollination process. The pollination process involves the flower constancy and pollinator behaviour of the flower pollination algorithm. These pollination processes were based on the following:

1 Biotic pollination and cross-pollination are observed with global pollination processes which are used in search of the global best solutions by which pollinators carry pollen through levy flights.

2 Abiotic pollination and self-pollination are considered as the local pollination which is used in search of current best solutions.

3 Flower constancy is described as the proliferation relationship that is proportional to the rate at which two flowers are similar.

4 Local pollination and global pollination are controlled-switch probability *P* between the interval value of 0 and 1 because of the physical proximity and other factors like wind. Local pollination has a significant fraction of *P* in the overall pollination processes.

The pollinators in global pollination transport the pollen of a flower. The pollinators can move over long distances to search for the global best solution. Global pollination can be expressed mathematically as:

$$x_i^{t+1} = x_i^t + L\left(x_i^t - x_g^t\right) \tag{15}$$

where $x_i^{t+1}$ is the next position, $x_i^t$ is the current position of the $i^{\text{th}}$ iteration, $x_g^t$ is the global best solution, and *L* is the levy flight step size which shows the firmness of the pollinators.

The Levy flight is used to imitate the random movement of pollinators. Levy flight can be expressed mathematically as:

$$L \sim \frac{\lambda\Gamma(\lambda)\sin\left(\frac{\pi\lambda}{2}\right)}{\pi}\frac{1}{S^1 + \lambda} \qquad (S \gg S_0 > 0) \tag{16}$$

where $\Gamma(\lambda)$ is the standard gamma function when the levy flight has valid long steps and big step $S > 0$ is the right value to obtain this distribution.

Local pollination can be expressed mathematically as:

$$x_i^{t+1} = x_i^t + \varepsilon\left(x_k^t - x_j^t\right) \tag{17}$$

where $\varepsilon$ is the uniform distribution interval value between 0 and 1 of a variable including a randomly generated value. $x_k^t$ and $x_j^t$ are two randomly selected solutions from the current pollinators or vectors.

## *4.1 Modified flower pollination algorithm*

Flower pollination has been modified in this section based on global pollination and local pollination.

### *4.1.1 Global pollination modification*

Global pollination is designed in such a way that pollinators can transfer the pollen of a flower over long distances. The speed of the pollinator's position and movement is increased for pollinators to be able to transfer pollen over longer distances. The pollinators move in the direction of one flower and transport the pollen to another flower. The proposed flower pollination algorithm by Yang (2012) only enlightens on the next and current position in search of the global best solution. However, the speed at which the pollinators moved was not enlightened. The speed is added in this paper and also the movement of pollinators in search for the global best is improved by forward and backward-based. When the pollinators search for the global best solution, they later go back to the previous position to search for the next global best position by transfer of pollen which increases the rate at which the more global best solution can be found. The time interval at which the next position is detected and increased.

The global pollination for the next position is obtained as:

$$x_i(t+1) = V_i(t) + x_i(t) + L\left(x_i(t) - x_g(t)\right) \tag{18}$$

The next speed is obtained as:

$$V_i(t+1) = V_i(t) + x_i(t) \tag{19}$$

where $V_i(t + 1)$ and $V_i(t)$ are the next speed and the pollinators' current speed, respectively. $x_i(t + 1)$ is the next position, $x_i(t)$ is the current position of the $i^{th}$ iteration, $x_g^t$ is the global best solution, and $L$ is the levy flight step size which shows the firmness of the pollinators.

The Levy flight step size which shows the firmness of the pollinators $L$ can be deduced as:

$$L = \left(\frac{t}{t_{\max}} - \frac{t}{t_{\min}}\right)d \tag{20}$$

where $t_{\max}$ and $t_{\min}$ are the maximum and the minimum number of iterations respectively and $d$ is the distance between the current and global best position detected.

The global best solution can be searched faster with two-step sizes $r_1$ and $r_2$. These two-step sizes will find the directions of the global best from the current position.

$$x_i(t+1) = V_i(t) + x_i(t) + L\left(2r\left(x_{r1}(t) - x_{r2}(t)\right) - x_g(t)\right) \tag{21}$$

where $x_{r1}(t)$ and $x_{r2}(t)$ are the random step sizes at iteration $t$ and $r$ is any number between the interval of 0 and 1.

The global pollination for the forward solution can be given as:

$$x_i(t+1) = V_i(t) + x_i(t) + \left(\frac{t}{t_{\max}} - \frac{t}{t_{\min}}\right) \cdot \left(2rx_i(t) - x_g(t)\right) \tag{22}$$

Equation (22) can be simplified as:

$$x_i(t+1) = V_i(t) + x_i(t) + 2rx_i(t)\left(\frac{t}{t_{\max}} - \frac{t}{t_{\min}}\right) - x_g(t)\left(\frac{t}{t_{\max}} - \frac{t}{t_{\min}}\right) \tag{23}$$

For forward global pollination, the pollinators move forward to the next position. So the next position is given as $x_i(t + 1)$.

The global pollination for the backward solution can be given as

$$x_i(t-1) = V_i(t) + x_i(t) + \left(\frac{t}{t_{\max}} - \frac{t}{t_{\min}}\right) \cdot \left(2rx_i(t) + x_g(t)\right) \tag{24}$$

Equation (24) can be simplified as:

$$x_i(t-1) = V_i(t) + x_i(t) + 2rx_i(t)\left(\frac{t}{t_{\max}} - \frac{t}{t_{\min}}\right) + x_g(t)\left(\frac{t}{t_{\max}} - \frac{t}{t_{\min}}\right) \tag{25}$$

For backward global pollination, the pollinators move backward to the previous position. So the previous position is given as $x_i(t-1)$.

### *4.1.2 Local pollination modification*

Local pollination transfer pollen from the same flowers. The speed is fast because the distance is shorter since it's the same flower whereby the transfer is taking place. The current position is where the pollinator takes the pollen from and the next position is where the pollinator takes the pollen since the transfer is between the same flowers. Local pollination performs well when two or more random pollinators are selected with maximum iteration.

The local pollination for the next position is obtained as:

$$x_i(t+1) = V_i(t) + x_i(t) - x_{best}(t) \tag{26}$$

Equation (26) is deduced by substituting the minimum and maximum iteration numbers. The next speed is obtained as:

$$x_i(t+1) = V_i(t) + \frac{t}{t_{\max} - t_{\min}} x_i(t) - \frac{t}{t_{\max} - t_{\min}} x_{best}(t) \tag{27}$$

Adding two randomly selected solutions to equation (27):

$$x_i(t+1) = V_i(t) + \frac{t}{t_{\max} - t_{\min}} x_i(t) - \frac{t}{t_{\max} - t_{\min}} x_{best}(t)\varepsilon\left(x_k(t) - x_j(t)\right) \tag{28}$$

When another two randomly selected solutions are added to equation (27), the equation can be derived for forward local pollination can be obtained as:

$$\begin{aligned} x_i(t+1) = {} & V_i(t) + \frac{t}{t_{\max} - t_{\min}} x_i(t) - \frac{t}{t_{\max} - t_{\min}} x_{best}(t) + \varepsilon\left(x_k(t) - x_j(t)\right) \\ & + \varepsilon_1\left(x_n(t) - x_m(t)\right) \end{aligned} \tag{29}$$

When another two randomly selected solutions are added to equation (27), the equation can be derived for backward local pollination can be obtained as:

$$\begin{aligned} x_i(t+1) = {} & V_i(t) + \frac{t}{t_{\max} - t_{\min}} x_i(t) - \frac{t}{t_{\max} - t_{\min}} x_{best}(t) + \varepsilon\left(x_k(t) - x_j(t)\right) \\ & + \varepsilon_1\left(x_n(t) - x_m(t)\right) \end{aligned} \tag{30}$$

where $x_{best}$ is the current best position, $x_n$ and $x_m$ are two randomly selected solutions.

**Algorithm 1** Modified flower pollination algorithm pseudocode

1: Define the objective function min, max $f(x)$, $x = (x(1), x(2), x(3), \ldots, x(d))$

2: Initialise a population of the $n$ flower pollination with a random solution

3: Add the speed $V_i$ to the initial population

4: Find the best global solution $x_g$ in the initial population

5: Find the best local solution $x_{best}$ in the initial population

6: Define a switch probability $p \in [0, 1]$

**While** $t < t_{\max}$ and $t > t_{\min}$ **do**
  **for** $i = 1, \ldots, N$ **do**
    **if** $rand < p$ **then**
      Draw a (d-dimensional) step vector $L$ which obeys a Levy distribution
      Evaluate global pollination via $x_i(t+1) = V_i(t) + x_i(t) + L(x_i(t) - x_g(t))$
      Increase the population speed via $V_i(t+1) = V_i(t) + x_i(t)$
    **else**
      Select two random step sizes $r_1$ and $r_2$
      Double the step vector L to increase the population speed
      Find the $t_{\max}$ and $t_{\min}$ via $L = \left(\frac{t}{t_{\max}} - \frac{t}{t_{\min}}\right)d$
      Update global pollination via $x_i(t+1) = V_i(t) + x_i(t) + L(2r(x_{r1}(t) - x_{r2}(t)) - x_g(t))$
      Find forward global pollination via
      $$x_i(t+1) = V_i(t) + x_i(t) + \left(\frac{t}{t_{\max}} - \frac{t}{t_{\min}}\right) \cdot (2rx_i(t) - x_g(t))$$
      Find backward global pollination via
      $$x_i(t-1) = V_i(t) + x_i(t) + \left(\frac{t}{t_{\max}} - \frac{t}{t_{\min}}\right) \cdot \left(2rx_i(t) + x_g(t)\right)$$
    **else**
      Draw $\in$ from a uniform distribution in [0, 1]
      Select two random solutions $k$ and $j$
      Evaluate local pollination via
      $$x_i(t+1) = V_i(t) + \left(\frac{t}{t_{\max}} - \frac{t}{t_{\min}}\right)(x_i(t) - x_{best}(t)) + \varepsilon(x_k(t) - x_j(t))$$
      Increase the local pollination by selecting two random solutions $n$ and $m$
      Evaluate forward local pollination via
      $$x_i(t+1) = V_i(t) + \left(\frac{t}{t_{\max}} - \frac{t}{t_{\min}}\right)(x_i(t) - x_{best}(t)) + \varepsilon(x_k(t) - x_j(t)) + \varepsilon_1(x_n(t) - x_m(t))$$
      Evaluate backward local pollination via
      $$x_i(t+1) = V_i(t) + \left(\frac{t}{t_{\max}} - \frac{t}{t_{\min}}\right)(x_i(t) - x_{best}(t)) + \varepsilon(x_k(t) - x_j(t)) + \varepsilon_1(x_n(t) - x_m(t))$$
    **end if**
      Global and local new solution is better
      Forward global and local pollination $f(x_i(t+1)) > f(x_i(\mathrm{t}))$
      Speed of the forward global and local pollination $f(V_i(t+1)) > f(V_i(t))$
    **else**
      Backward global and local pollination $f(x_i(t-1)) < f(x_i(t))$
      Speed of the backward global and local pollination $f(V_i(t-1)) < f(V_i(t))$
      New solution is better and updated in the population
  **end for**

Global best pollination $f(x_g(t) > f(x_i(t))$
Local best pollination $f(x_{best}(t)) > f(x_i(t))$
**end while**

## 5 Observed controller-based flower pollination algorithm

### *5.1 Observer controller based on the forward global and local pollination*

From the Luenberger observer control method, considering the observer controller of the forward flower pollination with global and local pollination.

The forward global pollination of the Luenberger observer control method is obtained as:

$$\begin{cases} x_i(t+1) = Ax_i(t) + Lx_i(t) - Lx_g(t) \\ y_i(t) = Bx_i(t) + V_i(t) \end{cases} \tag{31}$$

The forward local pollination of the Luenberger observer control method is obtained as:

$$\begin{cases} x_i(t+1) = Ax_i(t) + x_b(t) + \varepsilon x_k(t) - \varepsilon x_j(t) \\ y_i(t) = Cx_i(t) + V_i(t) \end{cases} \tag{32}$$

where $x_i(t + 1)$ is the next position vector, $x_i(t) \in R^n$, $y_i(t) \in R^m$, $Lx_i(t)$: $Z_+ \to R^n$, $Lx_g(t)$: $Z_+ \to R^n$, $x_b(t)$: $Z_+ \to R^n$ represent the current position vector, position vector, current position vector of levy flight, global best position vector, and current best position vector respectively. $\varepsilon x_k(t)$: $Z_+ \to R^n$ and $\varepsilon x_j(t)$: $Z_+ \to R^n$ represent two randomly selected vectors. $V_i(t)$: $Z_+ \to R^m$ represent the speed of the vector, $A$: $Z_+ \to R^{n\times n}$, $B$: $Z_+ \to R^{m\times n}$, $C$: $Z_+ \to R^{m\times n}$ are the matrix functions of the appropriate positions vector.

The parameter's value of the position vector can be represented by

$$A(\alpha, t) = A(t) + \alpha_1 A_1(t) + \alpha_2 A_2(t) + \alpha_3 A_3(t) + \ldots + \alpha_n A_n(t) \tag{33}$$

where $\alpha_1$, $\alpha_2$, $\alpha_3$, …, $\alpha_n$ and $A(t)$, $A_1(t)$, $A_2(t)$, $A_3(t)$, …, $A_n(t)$ are parameters and the appropriate positions of the vector respectively.

The parameter's value of the position vector is simplified as:

$$A(\alpha, t) = A(t) + \sum_{i=1}^{n} \alpha_i A_i(t) \tag{34}$$

Substituting equation (34) into equations (31) and (32).

The forward global pollination vector position is obtained as:

$$\begin{cases} x_i(t+1) = Ax_i(t) + \sum_{i=1}^{n} \alpha_i A_i x_i(t) + Lx_i(t) - Lx_g(t) \\ y_i(t) = Bx_i(t) + V_i(t) \end{cases} \tag{35}$$

The forward local pollination vector position is obtained as:

$$\begin{cases} x_i(t+1) = Ax_i(t) + \sum_{i=1}^{n} \alpha_i A_i x_i(t) + x_b(t) + \varepsilon x_k(t) - \varepsilon x_j(t) \\ y_i(t) = Cx_i(t) + V_i(t) \end{cases} \tag{36}$$

Substituting equations (35) and (36) into the Luenberger observer theorem,

The forward global pollination Luenberger observer theorem is obtained as:

$$\begin{cases} x_i(t+1) = Dx_i(t) + \sum_{i=1}^{t} \beta_i x_i(t) + Lx_i(t) - Lx_g(t) + L_T\left(y_i(t) - V_i(t)\right) \\ y_i(t) = Bx_i(t) + V_i(t) \end{cases} \tag{37}$$

The forward local pollination Luenberger observer theorem is obtained as:

$$\begin{cases} x_i(t+1) = Dx_i(t) + \sum_{i=1}^{t} \beta_i x_i(t) + x_b(t) + \varepsilon x_k(t) - \varepsilon x_j(t) + LT\left(y_i(t) - V_i(t)\right) \\ y_i(t) = Cx_i(t) + V_i(t) \end{cases} \tag{38}$$

where $Dx_i(t) = Ax_i(t) - L_TBx_i(t)$, $Dx_i(t) = Ax_i(t) - L_TCx_i(t)$ and $\beta_i x_i(t) = \alpha_i A_i x_i(t)$ and $L_T$ represents the Luenberger theorem.

The forward position vector for the global pollination can be given using the Luenberger theorem:

$$\begin{cases} x_i(t+1) = Dx_i(t) + \sum_{i=1}^{t} \beta_i x_i(t) + Lx_i(t) - Lx_g(t) + L_T y_i(t) - L_T V_i(t) \\ y_i(t) = Bx_i(t) + V_i(t) \end{cases} \tag{39}$$

The forward position vector for the local pollination can be given using the Luenberger theorem:

$$\begin{cases} x_i(t+1) = Dx_i(t) + \sum_{i=1}^{t} \beta_i x_i(t) + x_b(t) + \varepsilon x_k(t) - \varepsilon x_j(t) + L_T y_i(t) - L_T V_i(t) \\ y_i(t) = Cx_i(t) + V_i(t) \end{cases} \tag{40}$$

Assuming there is an interval estimation error for the position of the vectors in the forward global pollination, the error is calculated as:

$$e_g(t) = x_i(t) - x_g(t) \tag{41}$$

Assuming there is an interval estimation error for the position of the vectors in the forward local pollination, the error is calculated as:

$$e_L(t) = x_i(t) - x_b(t) \tag{42}$$

From $Dx_i(t) = Ax_i(t) - L_TBx_i(t)$ and $Dx_i(t) = Ax_i(t) - L_TCx_i(\mathrm{t})$ in equations (39) and (40), the error for the next position vector for global $e_g$ and local $e_L$ pollination is obtained as:

$$\begin{cases} e_g(t+1) = Dx_i(t)e_g(t) + \partial(t) + \tau(t) \\ e_L(t+1) = Dx_i(t)e_L(t) + \varnothing(t) + \theta(t) \end{cases} \tag{43}$$

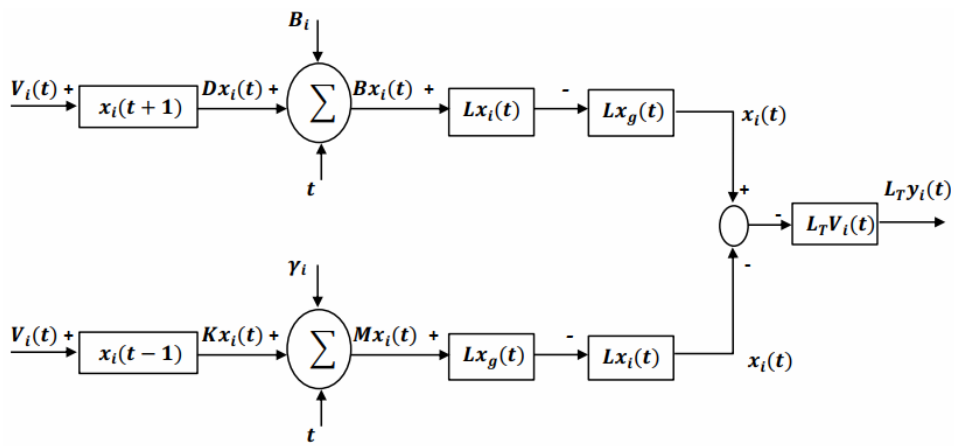
where the parameters in equation (43) are deduced as:

$$\begin{cases} \partial(t) = \sum_{i=1}^{t} \beta_i x_i(t) + Lx_i(t) \\ \tau(t) = \sum_{i=1}^{t} \beta_i x_i(t) + x_b(t) \end{cases} \tag{44}$$

The parameters in equation (43) are deduced as:

$$\begin{cases} \varnothing(t) = Lx_g(t) + L_T y_i(t) + L_T V_i(t) \\ \theta(t) = \varepsilon x_k(t) - \varepsilon x_j(t) + L_T y_i(t) + L_T V_i(t) \end{cases} \tag{45}$$

**Figure 5** Global pollination forward and backward observer controller diagram




## *5.2 Observer controller based on the backward global and local pollination*

From the Luenberger observer control method, considering the observer controller of the backward flower pollination with global and local pollination.

The backward global pollination of the Luenberger observer control method is obtained as:

$$\begin{cases} x_i(t-1) = Fx_i(t) + Lx_g(t) - Lx_i(t) \\ y_i(t) = Mx_i(t) + V_i(t) \end{cases} \tag{46}$$

The backward local pollination of the Luenberger observer control method is obtained as:

$$\begin{cases} x_i(t-1) = Fx_i(t) + \varepsilon x_k(t) - \varepsilon x_j(t) - x_b(t) \\ y_i(t) = Nx_i(t) + V_i(t) \end{cases} \tag{47}$$

where $x_i(t-1)$ is the previous position vector, $x_i(t) \in R^n$, $y_i(t) \in R^m$, $Lx_g(t)$: $Z_+ \rightarrow R^n$, $Lx_i(t)$: $Z_+ \rightarrow R^n$, $x_b(t)$: $Z_+ \rightarrow R^n$ represent the current position vector, position vector, global best position vector, current position vector of levy flight, global best position vector, and current best position vector respectively. $\varepsilon x_k(t)$: $Z_+ \rightarrow R^n$ and $\varepsilon x_j(t)$: $Z_+ \rightarrow R^n$ represent two randomly selected vectors. $V_i(t)$: $Z_+ \rightarrow R^m$ represent the speed of the vector, $F$: $Z_+ \rightarrow R^{n\times n}$, $M$: $Z_+ \rightarrow R^{m\times n}$, $N$: $Z_+ \rightarrow R^{m\times n}$are the matrix functions of the appropriate positions vector.

The parameter's value of the position vector can be represented by

$$F(\partial, t) = F(t) + \partial_1 F_1(t) + \partial_2 F_2(t) + \partial_3 F_3(t) + \ldots + \partial_k F_k(t) \tag{48}$$

where $\partial_1$, $\partial_2$, $\partial_3$, …, $\partial_k$ and $F(t)$, $F_1(t)$, $F_2(t)$, $F_3(t)$, …, $F_k(t)$ are parameters, and the appropriate positions of the vector respectively.

$$F(\partial, t) = F(t) + \sum_{i=1}^{k} \partial_i F_i(t) \tag{49}$$

Substituting equation (49) into equations (46) and (47).

The backward global pollination vector position is obtained as:

$$\begin{cases} x_i(t-1) = Fx_i(t) + \sum_{i=1}^{k} \partial_i F_i x_i(t) + Lx_g(t) - Lx_i(t) \\ y_i(t) = Mx_i(t) + V_i(t) \end{cases} \tag{50}$$

The backward local pollination vector position is obtained as:

$$\begin{cases} x_i(t-1) = Fx_i(t) + \sum_{i=1}^{k} \partial_i F_i x_i(t) + \varepsilon x_k(t) - \varepsilon x_j(t) - x_b(t) \\ y_i(t) = Nx_i(t) + V_i(t) \end{cases} \tag{51}$$

Substituting equations (35) and (36) into the Luenberger observer theorem.

The backward global pollination Luenberger observer theorem is obtained as:

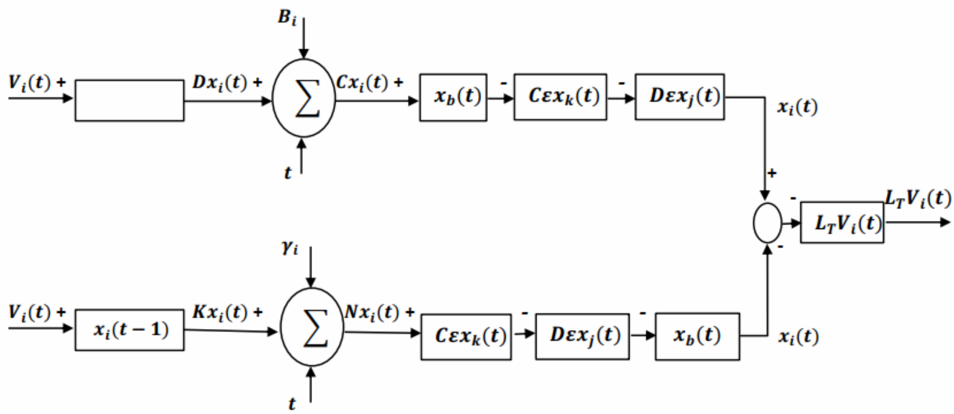
$$\begin{cases} x_i(t-1) = Kx_i(t) + \sum_{i=1}^{t} \gamma_i x_i(t) + Lx_g(t) - Lx_i(t) + L_T\left(y_i(t) - V_i(t)\right) \\ y_i(t) = Mx_i(t) + V_i(t) \end{cases} \tag{52}$$

The backward local pollination Luenberger observer theorem is obtained as:

$$\begin{cases} x_i(t-1) = Kx_i(t) + \sum_{i=1}^{t} \gamma_i x_i(t) + \varepsilon x_k(t) - \varepsilon x_j(t) - x_b(t) + L_T\left(y_i(t) - V_i(t)\right) \\ y_i(t) = Nx_i(t) + V_i(t) \end{cases} \tag{53}$$

where $Kx_i(t) = Fx_i(t) - L_T M x_i(t)$, $Kx_i(t) = Fx_i(t) - L_T N x_i(t)$ and $\gamma_i x_i(t) = \partial_i F_i x_i(t)$ and $L_T$ represents the Luenberger theorem.

**Figure 6** Local pollination forward and backward observer controller diagram




The backward position vector for the global pollination can be given using the Luenberger theorem:

$$\begin{cases} x_i(t-1) = Kx_i(t) + \sum_{i=1}^{t} \gamma_i x_i(t) + Lx_g(t) - Lx_i(t) + L_T y_i(t) - L_T V_i(t) \\ y_i(t) = Mx_i(t) + V_i(t) \end{cases} \tag{54}$$

The backward position vector for the local pollination can be given using the Luenberger theorem:

$$\begin{cases} x_i(t-1) = Kx_i(t) + \sum_{i=1}^{t} \gamma_i x_i(t) + \varepsilon x_k(t) - \varepsilon x_j(t) - x_b(t) + L_T y_i(t) - L_T V_i(t) \\ y_i(t) = Nx_i(t) + V_i(t) \end{cases} \tag{55}$$

Assuming there is an interval estimation error for the position of the vectors in the backward global pollination, the error is calculated as:

$$e_g(t) = x_g(t) - x_i(t) \tag{56}$$

Assuming there is an interval estimation error for the position of the vectors in the backward local pollination, the error is calculated as:

$$e_L(t) = x_b(t) - x_i(t) \tag{57}$$

From $Kx_i(t) = Fx_i(t) - L_T Mx_i(t)$ and $Kx_i(t) = Fx_i(t) - L_T Nx_i(t)$ in equations (54) and (55), the error for the next position vector for global $e_g$ and local $e_L$ pollination is obtained as:

$$\begin{cases} e_g(t-1) = Kx_i(t) e_g(t) + \delta(t) + \sigma(t) \\ e_L(t-1) = Kx_i(t) e_L(t) + \aleph(t) + \varphi(t) \end{cases} \tag{58}$$

where the parameters in equation (58) are deduced as:

$$\begin{cases} (t) = \sum_{i=1}^{t} \gamma_i x_i(t) + Lx_g(t) \\ \sigma(t) = \sum_{i=1}^{t} \gamma_i x_i(t) + \varepsilon x_k(t) - \varepsilon x_j(t) \end{cases} \tag{59}$$

where the parameters in equation (58) are deduced as:

$$\begin{cases} \aleph(t) = Lx_i(t) + L_T y_i(t) - L_T V_i(t) \\ \varphi(t) = x_b(t) + L_T y_i(t) - L_T V_i(t) \end{cases} \tag{60}$$

**Algorithm 2** Luenberger observer control-based flower pollination algorithm pseudocode

Define the objective function $f = (x, y)$

Initialise a population of the constant matrix vectors $A$, $B$, $C$, $D$, $E$ and $F$ with a random solution

Regulate the speed with the input $y_i$ and output $x_i$ vectors

Find the best solution for global and local pollination via Luenberger method

Represent the parameter values of the position vector with constant vectors

**While**

  $x_i(t + 1) = y_i(t)$

  **for** $i = 1$ **do**

    **if** $rand < t$ **then**

      Draw an input and output Luenberger observer global pollination

      Add the input $y_i(t) = x_i(t) + V_i(t)$

      Increase the forward global pollination via

$$x_i(t+1) = Ax_i(t) + \sum_{i=1}^{n} \alpha_i A_i x_i(t) + Lx_i(t) - Lx_g(t)$$

      Increase the backward global pollination via

$$x_i(t-1) = Fx_i(t) + \sum_{i=1}^{k} \partial_i F_i x_i(t) + Lx_g(t) - Lx_i(t)$$

      Evaluate the global pollination with Luenberger theorem

      Update the forward global pollination via

$$x_i(t+1) = Dx_i(t) + \sum_{i=1}^{t} \beta_i x_i(t) + Lx_i(t) - Lx_g(t) + L_T y_i(t) - L_T V_i(t)$$

      Update the backward global pollination via

$$x_i(t-1) = Kx_i(t) + \sum_{i=1}^{t} \gamma_i x_i(t) + Lx_g(t) - Lx_i(t) + L_T y_i(t) - L_T V_i(t)$$

    **else**

      Draw Luenberger observer local pollination for the input and output

      Add the input $y_i(t) = x_i(t) + V_i(t)$

      Evaluate forward local pollination via

$$x_i(t+1) = Ax_i(t) + \sum_{i=1}^{n} \alpha_i A_i x_i(t) + x_b(t) + \varepsilon x_k(t) - \varepsilon x_j(t)$$

      Evaluate backward local pollination via

$$x_i(t-1) = Fx_i(t) + \sum_{i=1}^{k} \partial_i F_i x_i(t) + \varepsilon x_k(t) - \varepsilon x_j(t) - x_b(t)$$

Evaluate local pollination with Luenberger theorem

Update the forward local pollination via

$$x_i(t+1) = Dx_i(t) + \sum_{i=1}^{t} \beta_i x_i(t) + x_b(t) + \varepsilon x_k(t) - \varepsilon x_j(t) + L_T y_i(t) - L_T V_i(t)$$

Update the backward local pollination via

$$x_i(t-1) = Kx_i(t) + \sum_{i=1}^{t} \gamma_i x_i(t) + \varepsilon x_k(t) - \varepsilon x_j(t) - x_b(t) + L_T y_i(t) - L_T V_i(t)$$

**end if**

Evaluate the errors of global and local

$e_g > (t + 1)$ and $e_L > x_i(t + 1)$ then

$e_g < x_i(t - 1)$ and $e_L < x_i(t - 1)$

Evaluate global and local new solution

$f(x_i(t + 1)) > f(L_T y_i(t))$

**else**

$f(x_i(t - 1)) < f(L_T y_i(t))$

New solution is better and updated in the population

**end for**

Global best pollination $f(x_g(t)) > f(x_i(t))$

Local best pollination $f(x_{best}(t)) > f(x_i(t))$

**end while**

## 6 Simulation results and discussion

Figures 7 and 8 show the wind power generation diagram designed and modelled in MATLAB Simulink software without and with the modified FPA-MPPT algorithm. The PSMG was used as the induction generator during the simulation process. The parameters used for the simulation have been given in Tables 1, 2, and 3.

A wind speed of 2.5 m/s with a time of 1 s was tested in Figure 9 and compared with and without the modified FPA-MPPT method to check for the effectiveness of the method. From the simulation results of Figure 9, the wind speed increased initially at 0.04 s where a speed of 1.9 m/s was achieved but it later increased and remained at 2.2 m/s at 0.1 s without MPPT causing a decrease in the initial wind speed with low tracking speed since the initial speed used was 2.5 m/s. When tested with the modified FPA-MPPT, the wind speed increased initially at 0.03 s where a speed of 3 m/s was obtained and increases faster to 3.6 m/s at 0.1 s which indicates that the wind speed increased and was higher than the initial speed. The wind speed error measured is shown in Figure 10. The wind speed error occurs as a result of the wind direction and also the inconsistent tracking speed due to changes in weather conditions. The measurement of the wind speed error helps to know the changes that occur in the weather conditions during tracking. From the simulation result of Figure 10, the error occurred mostly at the time interval of 0.13 s to 0.27 s.

**Figure 7** Wind power generation system without modified FPA-MPPT in MATLAB Simulink (see online version for colours)

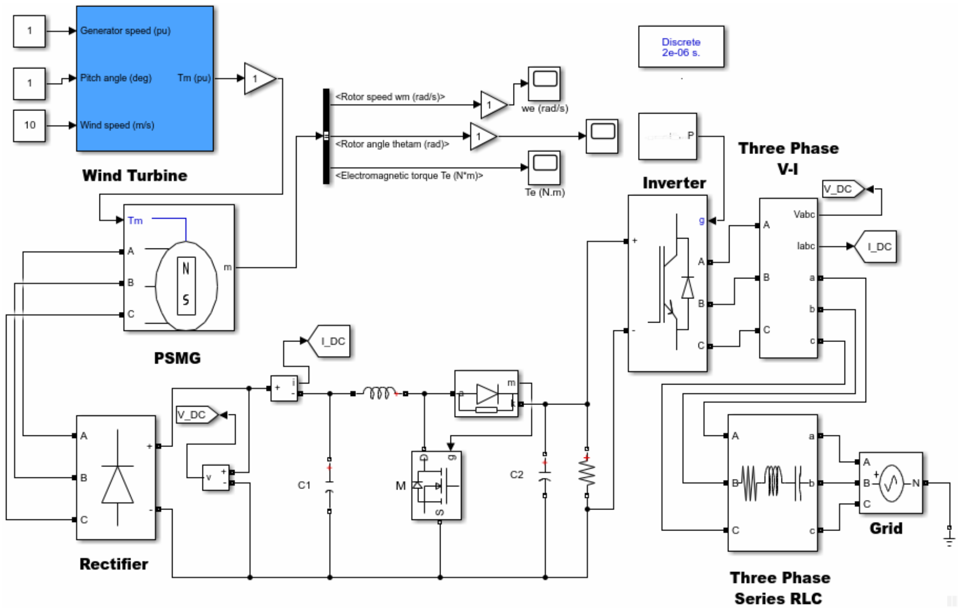



**Figure 8** Wind power generation system with modified FPA-MPPT in MATLAB Simulink (see online version for colours)

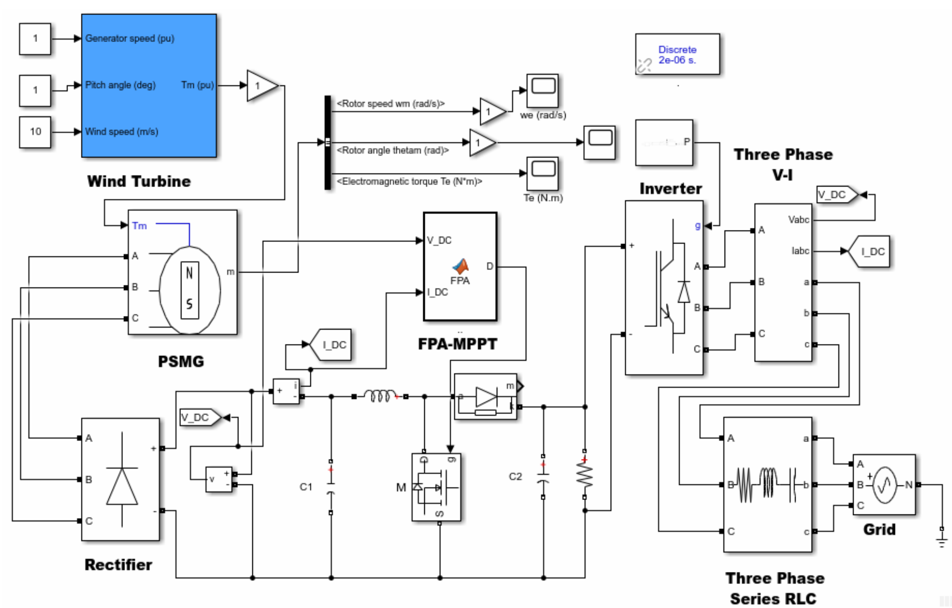


**Table 1** Parameters of wind turbine used

| *Parameters* | *Quantities and units* |
|---|---|
| Nominal mechanical output power | 5,000 W |
| Base power of the electrical generator | 5,000/0.9 VA |
| Base wind speed | 2 m/s |
| Maximum power at a base wind speed of nominal mechanical power | 1 p.u |
| Base rotational speed of base generator speed | 1 p.u |
| Pitch angle beta to display wind turbine power characteristics | 0 |

**Table 2** Parameters of PMSG

| Parameters | Quantities and units |
|---|---|
| Back electromotive force (EMF) waveform | Sinusoidal |
| Rotor type | Salient pole |
| Mechanical input | Torque (Tm) |
| Stator phase resistance | 0.00867 W |
| Inductances (*Ld*) | 0.00286 H |
| Inductances (*Lq*) | 0.00344 H |
| Flux linkage | 0.175 V.s. |
| Voltage constant | 126.966 W |
| Torque constant | 1.05 Nm |

**Table 3** Dataset of modified FPA

| *Parameters* | *Values* |
|---|---|
| Minimum number of iterations | 50 |
| Maximum number of iterations | 100 |
| Global initial position | 0.8 |
| Local initial position | 0.4 |
| Minimum random solution | 5 |
| Maximum random solution | 10 |
| First fitness | 0.5 |
| Second fitness | 0.9 |
| Zeros of the local pollinators | 0 and 1 |
| Zeros of the global pollinators | 1 and 5 |

Figure 11 shows the simulation results of the global pollination. The pollinator's speed used to test for the methods was 45 m/s with a tracking time of 2 s. The tests were carried out with constant, initial, forward, and backward pollination, and the output results were achieved. At constant pollination, the speed attained was 0 m/s since the pollinators were at a fixed location and there was no movement. At the initial pollination stage before the pollinators moved, the initial speed remained constant from the beginning of the initialisation until the time of 0.12 s then it decreased gradually from 45 m/s to 30 m/s between 0.12 s to 0.2 s. The speed continued to decrease at 0.2 s from 30 m/s to 15 m/s at

0.4 s. At the forward pollination stage, the speed increased to 50 m/s at the initial time and was stable throughout. At the backward pollination stage, the speed moved in a reverse direction. The speed from the initial time 0 s to 0.13 s was –20 m/s but decreased to –45 m/s at 0.2 s. At 0.4 s, the speed attained was –50 m/s, which shows the direct reverse speed of the forward pollination to the backward pollination. The forward and backward pollination moved at the same speed as shown in the simulation results. The proposed algorithm worked perfectly as shown in the simulation result. As the initial pollination decreases, the global and local pollination are searched quickly with the forward pollination maintaining the power stability and the backward pollination maintaining the power when reversed.

**Figure 9** Wind speed with and without modified FPA (see online version for colours)

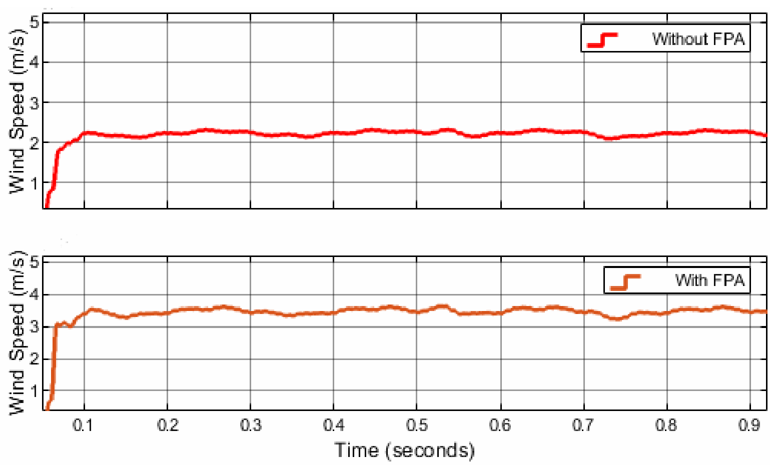



**Figure 10** Wind speed measured error with time (see online version for colours)

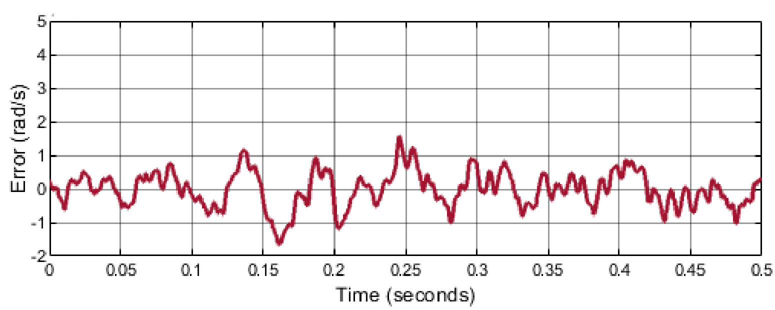
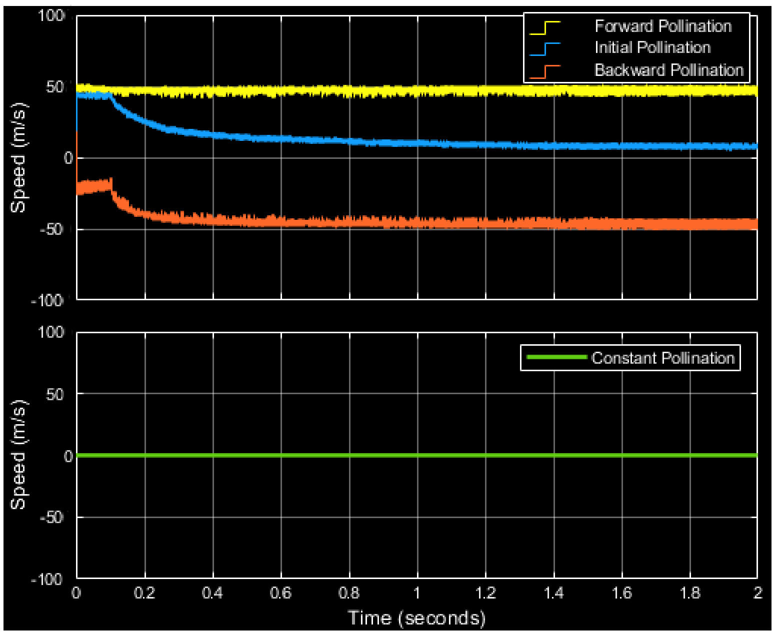



Figure 12 shows the simulation results of the local pollination. The pollinator's speed used to test for the methods was 10 m/s with a tracking time of 2 s. The tests were carried out with constant, initial, forward, and backward pollination, and the output results were achieved. At constant pollination, the speed attained was 0 m/s since the pollinators were at a fixed location and there was no movement. At the initial pollination stage before the pollinators moved, the initial dropped to 0 m/s due to the sudden tracking change. The speed then increased to 4 m/s at 0.1 s. The speed dropped gradually from 4 m/s at 0.1 s to 0 m/s at 0.2 s. At 2 s, the speed dropped to –5 m/s. At the forward pollination stage, the speed increased from 10 m/s to 17 m/s at 0.1 s and was stable but dropped a bit to 14 m/s at 1.4 s. At the backward pollination stage, the speed moved in a reverse direction. The speed from the initial time 0 s to 0.1 s was –17 m/s and remained stable but at 1 s, the speed increased, and at 2 s a speed of –12 m/s was achieved. The proposed algorithm produced a great result. For forward and backward pollination from the results obtained, the tracking speed increases more at the beginning however it slightly decreased at the end which shows that the tracking wasn't stable but the performance was good as more power is tracked. For the initial pollination, the algorithm didn't perform well in the beginning as it increases which delays the pollinator's movement but after a few seconds, it decreases which increases the speed.

**Figure 11** Global pollination speed with time (see online version for colours)




In Figures 13 and 14, the wind power generation system test was carried out with input power of 2,500 W and 5,000 W with a wind speed of 2 m/s and tracking time of 1 s, and generator speed of 1 pu at 0 deg pitch angle respectively. In Figures 13 and 14, when the system is connected without the modified FPA, the output power generated was 2,100 W and 2,400 W at 1 s respectively since there are fluctuations in the wind speed, the output power of the wind turbine is decreased due to variations in the wind speed. When the system is connected with the modified FPA, the output generated power was 3,600 W and 9,900 W at 1 s respectively which shows that the output power generated was increased since the modified FPA enhances the output power by increasing the rotation speed of the wind turbine and extracting more power from the wind turbine.

**Figure 12** Local pollination speed with time (see online version for colours)

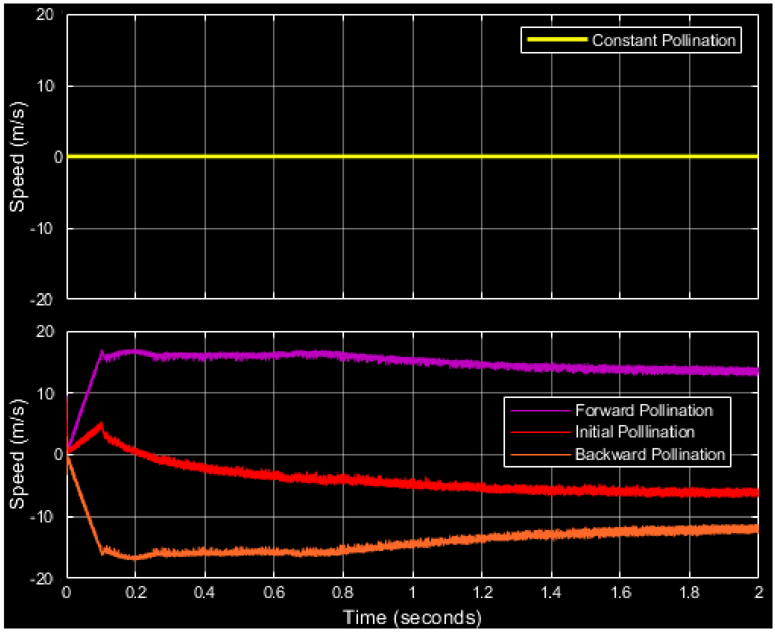



**Figure 13** Output power generated at 2,500 W with and without modified FPA (see online version for colours)

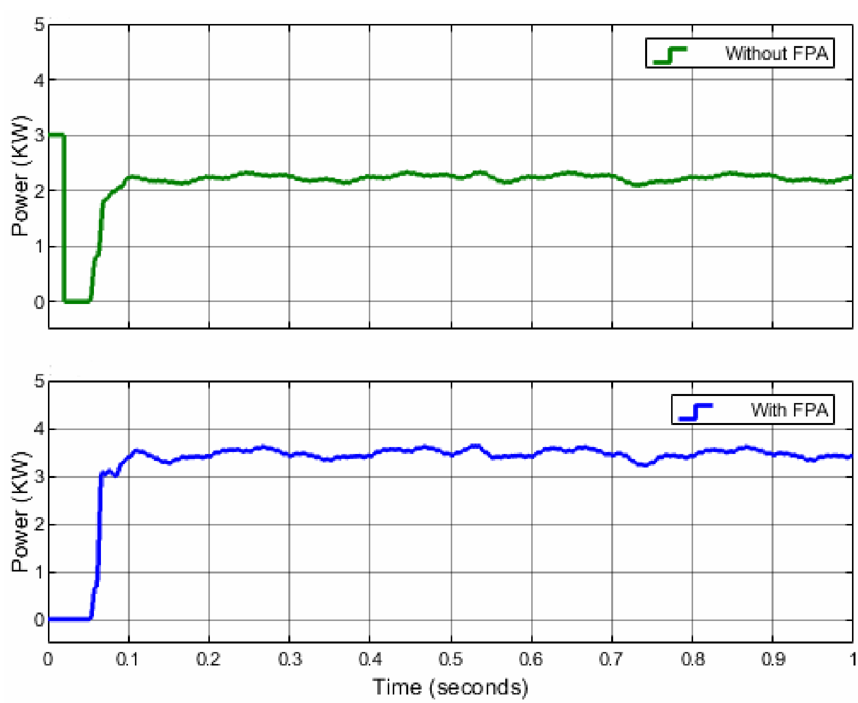
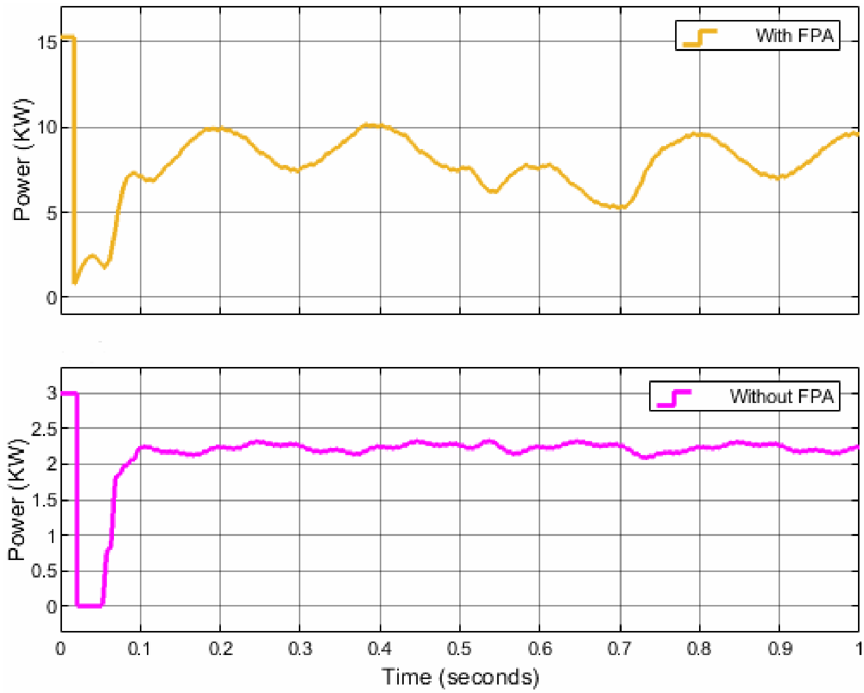


**Figure 14** Output power generated at 5,000 W with and without modified FPA (see online version for colours)




Figures 15 and 16 show the output generated active and reactive power without and with the modified FPA. The test was carried out with input power of 1,000 W with a wind speed of 2 m/s and tracking time of 1 s, and generator speed of 1 pu at 0 deg pitch angle respectively.

There is power instability when the active power is without the modified FPA which decreases the output active power in Figure 15. At the initial stage, the power decreases at 0.02 s to –950 W and increases to 100 W at 0.05 s. It became stable between 0.07 s to 0.6 s with an active power of –300 W. At 0.61 s, the generated power decreased to –1,200 W and increased at 0.64 s to 1,200 W. At 0.65 s, it decreased again to –1,100 W and at 0.66 s it increases to 100 W. There was high power instability between 0.61 s to 0.66 s as it decreases and increases simultaneously. It became stable again between 0.66 s to 1 s. at the initial stage and at 0.64 s, it increased to 100 W. For active power with the modified FPA in Figure 16, the output power was stable with the active power of 100 W generated.

The reactive power generated without the modified FPA dropped to –1,400 W at 0.04 s and was stable from 0.04 s to 0.62 s then it gradually increased to 2,000 W at 0.63 s. At 0.64 s, it decreases to 1,800 W and was stable at 1 s as shown in Figure 15. For reactive power generated with the modified FPA, at the initial stage, the power generated was -1,200 W at 0.03 s. Between 0.03 s to 0.6 s, there was power stability and at 0.61 s, the power gradually increases obtaining an output power of 2,100 W. At 0.63 s, it slightly decreases to 1,900 W and was stable at 1 s as shown in Figure 16.

**Figure 15** Output active and reactive power generated without FPA (see online version for colours)

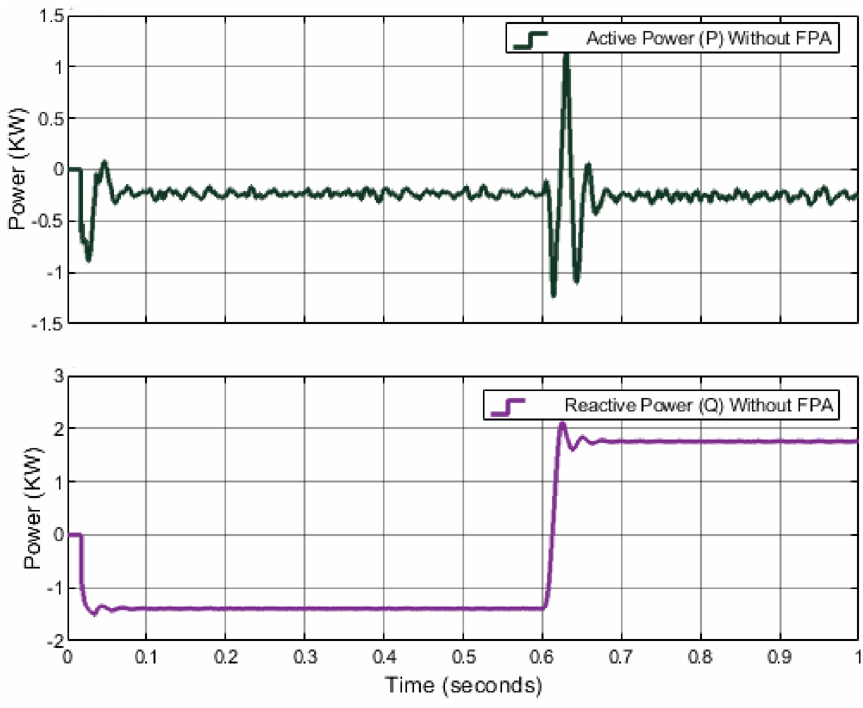



**Figure 16** Output active and reactive power generated with modified FPA (see online version for colours)

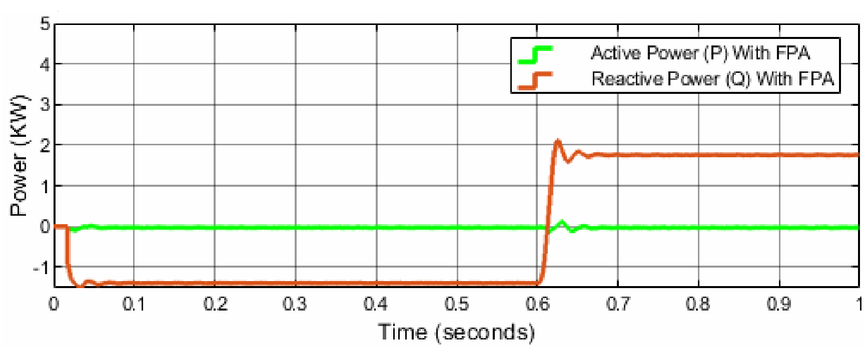
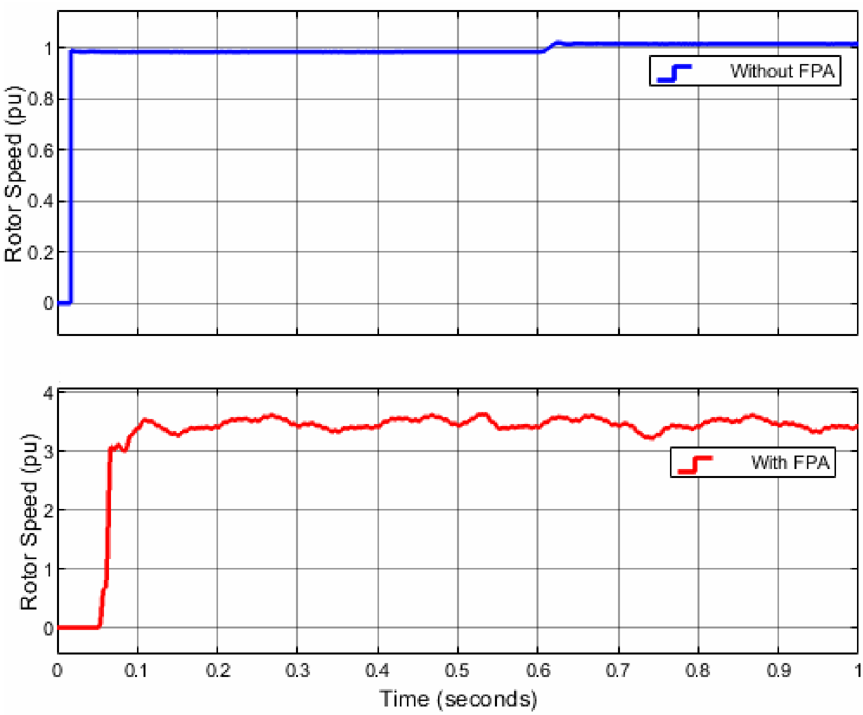



Figure 17 shows the rotor speed of the wind system with and without the modified FPA. The initial rotor speed used to carry out the test is 1 pu. When the system is connected without the modified FPA, the rotor speed remains at 1 pu at 0.01 s and remains constant with a slight increase at 0.62 s with 1.1 pu obtained. The rotor speed is low without the modified FPA which affects the output generated power since less wind is extracted by the wind turbine. When the system is connected to the modified FPA, the rotor speed increases from 3.5 pu to 0.07 s which enhances the output generated power by increasing the rotor speed which extracts more energy and produces more power. Between 0.07 s to 1 s, there are slight changes due to the change in the tracking speed.

**Figure 17** Rotor speed with and without modified FPA (see online version for colours)

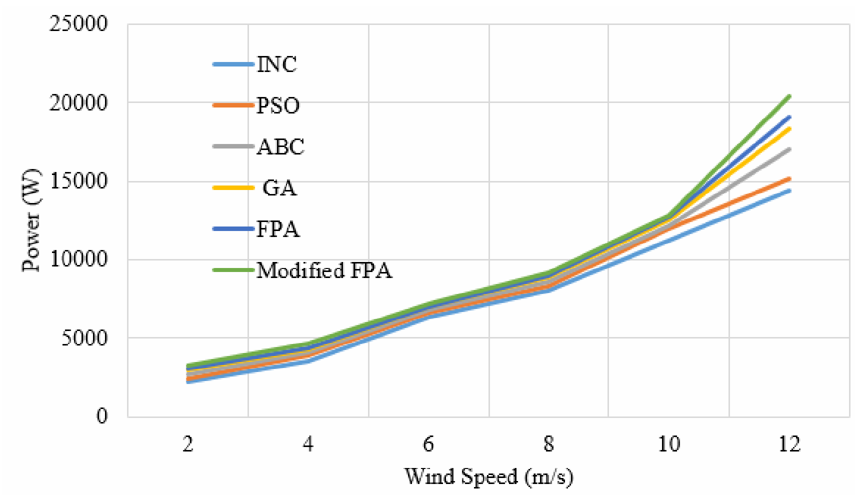



The proposed algorithm was compared with different algorithms. The wind power generation system test was carried out with wind speeds of 2 m/s, 4 m/s, 8 m/s, 10 m/s, and 12 m/s with tracking time of 10 s, and generator speed of 1 pu at 0 deg pitch angle respectively. Table 4 shows the output generated power compared with different methods proposed in the literature with different wind speeds when connected to the wind power system designed. For the simulation of the different methods, the minimum and maximum iteration number used was 50 and 100 respectively of which 1 was added for the next position to be achieved. The fitness was attained with the randomly selected values of 0.5 and 0.9. Two randomly selected solutions from the current pollinators were set with the values of 5 and 10 as the minimum and maximum values respectively. The initial position is set as 0.4 for the local pollinators and 0.8 for the global pollinators. The zeros of the local pollinators are 0 and 1 and the zeros for the global pollinators are 1 and 5. The output power is generated by multiplying the output voltage and current generated. The power is achieved when the best solution is equal to the current solution. The speed is added to the current position. Figure 18 shows the graph of the proposed algorithm compared with different algorithms with different wind speeds. From the results, the proposed modified FPA performs better than INC, PSO, ABC, GA, and the FPA algorithms. The modified FPA generates more power than all these MPPT algorithms which shows the effectiveness of the proposed method as shown and highlighted in Table 4.

**Table 4** Results of the output generated power of the wind generation system

| *Wind speed (m/s)* | *Power (W)* | | | | | |
|---|---|---|---|---|---|---|
| | *INC* | *PSO* | *ABC* | *GA* | *FPA* | *Modified FPA* |
| 2 | 2,175.63 | 2,387.59 | 2,693.53 | 2,997.11 | 3,028.34 | 3,208.89 |
| 4 | 3,569.47 | 3,877.41 | 4,081.38 | 4,287.09 | 4,346.85 | 4,611.21 |
| 6 | 6,338.14 | 6,647.69 | 6,853.47 | 6,968.15 | 7,009.56 | 7,184.62 |
| 8 | 8,059.15 | 8,275.51 | 8,595.16 | 8,907.43 | 8,985.78 | 9,122.38 |
| 10 | 11,256.59 | 11,971.51 | 12,188.42 | 12,501.75 | 12,675.65 | 12,826.74 |
| 12 | 14,432.72 | 15,166.22 | 17,041.09 | 18,348.64 | 19,063.04 | 20,364.58 |

**Figure 18** Output generated power compared with different algorithms (see online version for colours)




## 7 Conclusions

In this paper, a wind power generation system was designed based on the Luenberger observer controller which was used to modify the FPA which served as a MPPT was proposed. The Luenberger observer controller-based method used is designed as a two-stage structure added whereby the forward and backward movement of the pollinators search for the local and global best solution. The interval estimation error for the position of the vectors in the forward and backward movement for both global and local pollination is achieved. The movement of pollinators moves along with the speed in two directions in search of the global and local best since the FPA proposed in the literature only moves in one direction with no speed added. The speed added increases the fast movement of the pollinators in search of the best solution in both shorter and longer distances whereby the time interval for the next position is detected. Two-step sizes were added with the number of iterations and Levy flight is used to imitate the random movement of pollinators which is used to search for the global best solution at longer distances and two randomly selected variables with an increase in the iteration numbers are used to search for the local best solution at shorter distances.

This modified FPA movement enhances and improves the wind turbine's rotation speed and also stabilises the output of the wind turbine when the best solution is found since the output of the wind turbine is nonlinear, thereby causing more energy to be received by the wind turbine and more power to be extracted from the wind turbine to be supplied to the grid for use. The modified FPA also reduces power fluctuations. A wind power generation system was designed and implemented with the modified FPA. This control design is integrated from the energy-receiving stage by the wind turbine to the power generation stage. The PSMG was used as the induction generator. The induction generator enhances the rotational speed of the wind turbine. AC-DC and DC-AC converters are used to enhance the output voltage and current generated. The AC-DC converter controls the torque of the generator to set the generator's operation point strategy and the DC-AC converter maximises the output voltage generated. The wind power generation system was modelled and tests were carried out in MATLAB/Simulink by the proposed FPA.

A wind speed of 2.5 m/s with a time of 1 s was tested and compared with and without the modified FPA-MPPT method to check for the effectiveness of the method. From the simulation results, with the modified FPA, the wind speed increased faster from 2.5 m/s to 3.6 m/s at 0.1 s and without the modified FPA, the wind speed decreased from 2.5 m/s to 2.2 m/s at 0.1 s which shows the proposed modified FPA has a great performance.

Tests were carried out and simulation results for global pollination at the initial, constant, forward, and backward stages with a wind speed of 45 m/s. At constant pollination, the speed attained was 0 m/s. At the initial pollination stage, the speed decreased from 45 m/s to 30 m/s between 0.12 s to 0.2 s and decreased again from 30 m/s to 15 m/s at 0.4 s. At the forward pollination stage, the speed increased from 45 m/s to 50 m/s. At the backward pollination stage, the speed moves in the reverse direction from –45 m/s to –50 m/s. The proposed algorithm worked perfectly as shown in the simulation result. As the initial pollination decreases, the global and local pollination are searched quickly with the forward pollination maintaining the power stability and the backward pollination maintaining the power when reversed.

Tests were carried out and simulation results for local pollination at the initial, constant, forward, and backward stages with a wind speed of 10 m/s. At constant pollination, the speed attained was 0 m/s. At the initial pollination stage, the speed increased to 4 m/s at 0.1 s and decreased again to –5 m/s at 2 s. At the forward pollination stage, the speed increased from 10 m/s to 17 m/s at 0.1 s and was stable but dropped a bit to 14 m/s at 1.4 s. At the backward pollination stage, the speed moves in the reverse direction from –17 m/s to –12 m/s. The constant, forward, and backward for both the local and global pollination performed well as the speed increases as it moves. The initial for both the local and global pollination performance wasn't good as it decreases which affects the pollinators' movement in the beginning. The proposed algorithm produced a great result. For forward and backward pollination from the results obtained, the tracking speed increases more at the beginning however it slightly decreased at the end which shows that the tracking was not stable but the performance was good as more power is tracked. For the initial pollination, the algorithm did not perform well in the beginning as it increases which delays the pollinator's movement but after a few seconds, it decreases which increases the speed.

The test was carried out with input power of 2,500 W and 5,000 W with a wind speed of 2 m/s and tracking time of 1 s, and generator speed of 1 pu at 0 deg pitch angle respectively with and without the modified FPA. Without the modified FPA, the power generated decreases from the initial input power of 2,500 W to 2,100 W and also decreases from the initial input power of 5,000 W to 2,400 W. The decrease in power is caused due to variations in wind speed. With the modified FPA, the power generated increases from the initial input power of 2,500 W to 3,600 W and also increases from the initial input power of 5,000 W to 9,900 W. The increase in power is because the modified FPA enhances the output power by increasing the rotation speed of the wind turbine and extracting more power from the wind turbine.

The test was carried out with input power of 1,000 W with a wind speed of 2 m/s and tracking time of 1 s, and generator speed of 1 pu at 0 deg pitch angle respectively for output generated active and reactive power without and with the modified FPA. There is power instability when the active power is without the modified FPA which decreases and increases the output active power simultaneously from 0.05 s to 0.66 s. At 0.65 s the active power decreases again to –1,100 W and at 0.66 s it increases to 100 W. It became stable again between 0.66 s to 1 s. For active power with the modified FPA, the output power was stable with the active power of 100 W generated. There was stability for the active power with the modified FPA but the output generated power was reduced. The reactive power generated without the modified FPA dropped to –1,400 W at 0.04 s and increased to 2,000 W at 0.63 s and then decreased to 1,800 W at 0.64 s which shows the power instability. For reactive power generated with the modified FPA, the output reactive power increases by 2,100 W at 0.61 s and it slightly decreases to 1,900 W at 0.63 s and was stable at 1 s which shows that although there was an increase in the reactive power, there is still power instability.

The initial rotor speed used to carry out the test is 1 pu. When the system is connected without the modified FPA, the rotor speed remains at 1 pu. The rotor speed is low without the modified FPA which affects the output generated power since less wind is extracted by the wind turbine. When the system is connected with the modified FPA, the rotor speed increases to 3.5 pu 0.07 s which enhances the output generated power by increasing the rotor speed which extracts more energy and produces more power.

The proposed algorithm was compared with four other different algorithms which were INC, PSO, ABC, GA, and FPA. The wind power generation system test was carried out with wind speeds of 2 m/s, 4 m/s, 8 m/s, 10 m/s, and 12 m/s with tracking time of 10 s, and generator speed of 1 pu at 0 deg pitch angle respectively. The minimum and maximum iteration number used was 50 and 100, respectively. The fitness was attained with the randomly selected values of 0 and 1. From the results, the proposed modified FPA performs better than INC, PSO, ABC, GA, and FPA algorithms. The modified FPA generates more power than all these MPPT algorithms which shows the effectiveness of the proposed method

The proposed method performed better by generating more output power for the system with high tracking speed. It has a high convergence rate and high efficiency. The global and local best solutions were searched quickly. However, there was still power instability and fluctuation most especially for the active and reactive power. The initial pollination wasn't stable in the beginning which affects the initial tracking speed. The algorithm needs to be further improved in the future and then test with more parameters and also the iteration number should be increased to maintain power stability. The proposed method is complicated but effective and can be used to solve complex computational problems. The modified FPA can be improved and implemented with other optimisation methods such as evolutionary algorithm (EA), simulated annealing (SA), cuckoo search (CS), ant colony optimisation (ACO), firefly algorithm (FA), grey wolf optimisation (GWO), glowworm swarm optimisation (GSO), bacteria foraging (BF), whale optimisation algorithm (WOA), shuffled frog leaping algorithm (SFLA) to solve complex computational optimisation problems in the future.

## References


Ahmed, A., Galib, M.M.H., Zaman, S.M.K. et al. (2018) 'An optimization methodology of susceptance variation sing lead-lag controller for grid-connected FSIG based wind generator system', *Journal of the Franklin Institute*, Vol. 355, No. 1, pp.197–217.

Alejandro, R., Alvaro, L., Gerardo, V. et al. (2009) 'Modeling of a variable speed wind turbine with a permanent magnet synchronous generator', *IEEE International Symposium on Industrial Electronics (ISlE 2009)*, Seoul Olympic Parktel, Seoul, Korea, pp.5–8.

Amoh, M.A. and Xie, W. (2022) 'Design of control system for solar power generation based on an improved bat algorithm for an island operation', *Energy Harvesting and System*, https://doi.org/10.1515/ehs-2022-0045.

Amoh, M.A., Chen, H., Duku, O.A. et al. (2022) 'Solar power generation system based on signal search artificial bee colony optimization algorithm for maximum power point tracking', *2022 the 12th International Conference on Power and Energy Systems (ICPES 2022)*, IEEE, Guangzhou, China, pp.914–918.

Chen, S.M., Sarosh, A. and Dong, Y.F. (2012) 'Simulated annealing based artificial bee colony algorithm or global numerical optimization', *Applied Mathematics and Computation*, Vol. 219, No. 8, pp.3575–3589.

Chiranjit, S., Atanu, B., Pabitra, K.B. et al. (2020) 'Performance optimisation for closed loop control strategies towards simplified model of a PMSM drive by comparing with different classical and fuzzy intelligent controllers', *International Journal of Automation and Control*, Vol. 14, No. 4, pp.469–493.

Dinesh, K.S., Loveleen, G. and Daniel, O. (2007a) 'Image compression and feature extraction with neural network', *Allied Academies International Conference, Proceedings of the Academy of Information and Management Sciences*, Vol. 11, No. 1, pp.33–37.

Dinesh, K.S., Loveleen, G. and Daniel, O. (2007b) 'Image compression and feature extraction using Kohonen's self-organizing map neural network', *Journal of Strategic E-Commerce*, Vol. 5, No. 1, pp.25–38.

Duku, O.A., Ghamgeen, I.R., Hussain, H. et al. (2022) 'Using artificial bee colony algorithm for STATCOM sizing and placement to prevent voltage instability on Pakistan power system', *2022 7th Asia Conference on Power and Electrical Engineering (ACPEE 2022)*, IEEE, Hangzhou, China, pp.2129–2134.

Eduard, M., Zhang, Y.C., Vahan, G. et al. (2016) 'Understanding dynamic model validation of wind turbine generator and a wind power plant', *IEEE Energy Conversion Congress and Exposition (ECCE 2016)*, Milwaukee, Wisconsin, USA, pp.1–5.

Ellaia, R., Hachimi, H. and Elhami, A. (2012) 'A new hybrid genetic algorithm and particle swarm optimization', *Key Engineering Materials*, Vol. 498, pp.115–125.

Faccio, M., Gamberi, M., Bortolini, M. et al. (2018) 'State-of-art review of the optimization methods to design the configuration of hybrid renewable energy systems (HRESs)', *Front Energy*, Vol. 12, No. 4, pp.591–622.

Gaurav, K., Roshan, K., Ashok, K. et al. (2023) 'Development of modified LQG controller for mitigation of seismic vibrations using swarm intelligence', *International Journal of Automation and Control*, Vol. 17, No.1, pp.19–42.

Hadjer, A., Ameur, A. and Harmas, M.N. (2019) 'Modified PSO-based nonlinear controllers applied to a DC-DC converter', *International Journal of Automation and Control*, Vol. 13, No. 1, pp.1–16.

Hazem, H.A., Ghada, S.E. and Nashwa, A.K. (2022) 'Control of the rated production power of DFIG-wind turbine using adaptive PSO and PI conventional controllers', *International Journal of Automation and Control*, Vol. 16, No. 2, pp.219–249.

Jing, L. and Shi, C. (2022) 'Parameter settings in particle swarm optimisation algorithms: a survey', *International Journal of Automation and Control*, Vol. 16, No. 2, pp.164–182.

Jinqiang, H., Husheng, W., Bin, Z. et al. (2020) 'Swarm intelligence-based optimisation algorithms: an overview and future research issues', *International Journal of Automation and Control*, Vol. 14, Nos. 5–6, pp.656–693.

Ju, X. and Li, F. (2019) 'Wind farm layout optimization using self-informed genetic algorithm with information guided exploitation', *Applied Energy*, August, Vol. 248, pp.429–445.

Luenberger, D. (1971) 'An introduction to observers', *IEEE Transaction on Automatic Control*, Vol. 16, No. 60, pp.596–602.

Ma, T., Yang, H. and Lu, L. (2014) 'A feasibility study of a stand-alone hybrid solar-wind-battery system for a remote island', *Applied Energy*, Vol. 121, No. C, pp.149–158.

Mittal, R., Sandhu, K.S. and Jain, D.K. (2009) 'Low voltage ride-through (LVRT) of grid interfaced wind-driven PMSG', *Journal of Engineering and Applied Sciences*, Vol. 4, No. 5, pp.73-83.

Mohamed, A.B. and Ibrahim, M.H. (2015a) 'An effective hybrid flower pollination and genetic algorithm for constrained optimization problems', *Advanced Engineering Technology and Application*, Vol. 4, No. 2, pp.27–34.

Mohamed, A.B. and Ibrahim, M.H. (2015b) 'An improved flower pollination algorithm for ratios optimization problems', *Applied Mathematics & Information Sciences*, Vol. 3, No. 2, pp.83–91.

Mohammed, A.A., Awadallah, A.M., Iyad, A.D. et al. (2019) 'Island flower pollination algorithm for global optimization', *The Journal of Supercomputing*, Vol. 75, pp.5280–5323, https://doi.org/10.1007/s11227-019-02776-y.

Mohammed, A.H. and Abu, T. (2014) 'Modeling and study of a standalone PMSG wind generation system using MATLAB/Simulink', *Universal Journal of Electrical and Electronic Engineering,* Vol. 2, No. 7, pp.270–277.

Mohd, I., Rana, Y. and Khan, J. S. (2015) 'Analysis of the grid connected wind energy a system using MATLAB/Simulink', *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, Vol. 4, No. 7, pp.1–7.

Naidu, K., Mokhlis, H., Bakar, A.H.A. et al. (2017) 'Performance investigation of ABC algorithm in multi-area power system with multiple interconnected generators', *Applied Soft Computing*, Vol. 57, No. 2017, pp.436–451.

Nayak, P.C., Prusty, U.C. and Sahu, P.C. (2018) 'Design and tuning of flower pollination algorithm based 2DOF controller in AGC of multi area thermal wind system', *International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)*, Vol. 7, No. 2, pp.118–126.

Qu, Z., Zhang, K., Mao, W. et al. (2017) 'Research and application of ensemble forecasting based on a novel multi-objective optimization algorithm for wind-speed forecasting', *Energy Conversion and Management*, Vol. 154, pp.440–454.

Raj, T.G. and Kumar, B.R. (2018) 'Comparative analysis of incremental conductance and perturb & observe MPPT methods for single-switch DC/DC converter', *Proceedings of the 2018 National Power Engineering Conference (NPEC)*, Madurai, India, pp.1–5.

Saberi, Z., Fudholi, A. and Sopian, K. (2020) 'Fitting of weibull distribution method to analysis wind energy potential at Kuala Terengganu, Malaysia', *Journal of Advanced Research in Fluid Mechanics and Thermal Sciences*, Vol. 69, pp.13–22.

Song, M., Chen, K. and Wang, J. (2018) 'Three-dimensional wind turbine positioning using gaussian particle swarm optimization with differential evolution', *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 172, pp.317–324.

Syahputra, R., Wiyagi, R.O. and Sudarisman (2017) 'Performance analysis of a wind turbine with permanent magnet synchronous generator', *Journal of Theoretical and Applied Information Technology*, Vol. 95, No. 9, pp.1950–1957.

Yang, X.S. (2012) 'Flower pollination algorithm for global optimization', *11th International Conference on Unconventional Computing and Natural Computation (UCNC 2012)*, Orléans, France, pp.240–249.

Yu, S. and Lingjuan, H. (2022) 'Swarm activity-based dynamic PSO for distribution decision', *International Journal of Automation and Control*, Vol. 16, Nos. 3–4, pp.503–517.

Zaid, A.A., Ahamad, T.A., Mohammed, A.A. et al. (2018) 'Variants of the flower pollination algorithm: a review', *Nature-Inspired Algorithms and Applied Optimization*, pp.91–118, DOI: 10.1007/978-3-319-67669-2_5.

## Appendix

**Table A1** Abbreviations

| *Abbreviation* | *Meaning* |
|---|---|
| MPPT | Maximum power point tracking |
| FPA | Flower pollination algorithm |
| PMSG | Permanent magnet synchronous generator |
| AC | Alternating current |
| DC | Direct current |
| INC | Incremental conductance |
| PSO | Particle swarm optimisation |
| ABC | Artificial bee colony |
| GA | Genetic algorithm |

**Table A2** Symbols

| *Symbols* | *Quantity* | *Units* |
|---|---|---|
| $C_P$ | The coefficient of electric power of the wind turbine system | - |
| $V_w$ | The wind speed | m/s |
| $T_M$ | The mechanical torque of the wind turbine | Nm |
| $P_w$ | Power generated by wind turbine | W |
| $\omega_w$ | The angular speed of the wind turbine | rad/s |
| $T_w$ | Wind torque | Nm |
| $T_g$ | Generator torque | Nm |
| $\lambda$ | The ratio of the tip speed of the wind turbine | - |
| $\rho$ | Air density | kg/m$^3$ |
| $R$ | The radius of the wind turbine | m |
| $\beta$ | The pitch angle of the blade of the wind turbine | deg |