

The Application of Particle Swarm Optimization Control in Pitch Wind Turbine

O.Belghazi¹ *, My R.Douiri¹, M.Cherkaoui¹, T. Ainane² and M.Sobh³

¹Department of Electrical Engineering Mohammedia Engineering School, University Mohammed V Agdal,

²University Hassan II, Casablanca, Morocco

³Laboratory of Biotechnology, Environment and Quality, Faculty of Science, Ibn Tofail University, Kenitra, Morocco
BP 765 Agdal Rabat Morocco, Morocco

Received 4 July 2014; Accepted 3 August 2014

Abstract

Recent advances in high-throughput technologies and an increased knowledge of biological systems have enabled the used in systems engineering and its applicability to command and control. In this work, we examine the applicability and performance of Particle Swarm Optimization -PSO-, a biologically inspired algorithm, for command and control. Particle swarm optimization -PSO- has been recently used for this reverse engineering task. In this paper, we introduce three scoring metrics for assessing the optimality of the solution found by PSO. We address the problem of finding blade pitch profiles for maximizing power production while simultaneously minimizing fatigue loads. While there are a number of publications where the speed control of a pitch regulated turbine, In this paper, we show how this problem can be solved using PSO optimization. The conception of controller strategies will address the following main principles: Optimization of power production with simultaneous load reduction of the major wind turbine components, and Adaptive control with respect to actual operational conditions.

The simulation shows that the PSO controller can achieve better control performances than conventional pitch angle control strategies.

Keywords: Particle Swarm Optimization, Controller, Wind.

1. Introduction

Bio-engineering or engineering Biology is an emerging interdisciplinary field that is based on the increasing symbiosis between the biological sciences and engineering disciplines. Evolution in scientific knowledge and engineering technology resulted in new specializations in the work force. Training in engineering biology provides the necessary foundations in engineering and the life sciences.

Nonlinearity and uncertainty are difficult problems in controller design for the wind turbine because there is no systematic manner to find a stability condition necessary and sufficient to ensure robustness and performance.

Although there are a number of publications where the speed control of a pitch-regulated turbine is presented, for instance [1-5], no publications comparing the obtained results with measurements have been found by the authors. Particle Swarm Optimization (PSO) also is an evolutionary computational model which is based on swarm intelligence.

PSO is developed by Kennedy and Elberhart [6] who have been inspired by the research of the artificial living, PSO is also an optimizer based on population. The system is initialized firstly in a set of randomly generated potential solutions, and then performs the search for the optimum one iteratively. PSO finds the optimum solution by swarms following the best particle. Compared to GAs, the PSO has much more profound intelligent background and could be performed more easily. Based on its advantages, the PSO is

not only suitable for science research, but also engineering applications, in the fields of evolutionary computing, optimization and many others [7-10].

PSO does not use the gradient of the problem being optimized, which means PSO does not require for the optimization problem to be differentiable as is required by classic optimization methods such as quasi Newton methods and gradient descent. PSO can therefore also be used on optimization problems that are partially irregular, change over time, noisy, etc.

2. Particle Swarm Optimization

Particle swarm optimization (PSO) is a population based stochastic optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995, inspired by social behavior of fish schooling or bird flocking.

PSO shares many similarities with evolutionary computation techniques such as Genetic Algorithms. The system is initialized with a population of random solutions and searches for optima by updating generations. However, unlike Genetic Algorithms, PSO has no evolution operators such as crossover and mutation. In PSO, the potential solutions, called particles, fly through the problem space by following the current optimum particles.

Each particle keeps track of its coordinates in the problem space which are associated with the best solution (fitness) it has achieved so far. (The fitness value is also stored.) This value is called pbest. Another "best" value that

* E-mail address: ouissam.belghazi@yahoo.fr

is tracked by the particle swarm optimizer is the best value, obtained so far by any particle in the neighbors of the particle. This location is called l_{best} . When a particle takes all the population as its topological neighbors, the best value is a global best and is called g_{best} .

The particle swarm optimization concept consists of, at each time step, changing the velocity of (accelerating) each particle toward its p_{best} and l_{best} locations (local version of PSO). Acceleration is weighted by a random term, with separate random numbers being generated for acceleration toward p_{best} and l_{best} locations.

In past several years, PSO has been successfully applied in many research and application areas. It is demonstrated that PSO gets better results in a faster, cheaper way compared with other methods. Another reason that PSO is attractive is that there are few parameters to adjust. One version, with slight variations, works well in a wide variety of applications. Particle swarm optimization has been used for approaches that can be used across a wide range of applications, as well as for specific applications focused on a specific requirement. A more detailed description of the general PSO framework is shown in Algorithm 1

Algorithm 1 Particle Swarm Optimization (PSO)

1) **Input:**

A swarm of m particles $\{p_1, p_2, \dots, p_{m-1}, p_m\}$, each represented by an N dimensional initial position vector p and initial velocity vector V_p .

2) **Repeat:**

a) For each particle p , update the velocity vector V_p for time $t + 1$ according to

$$V_p^{t+1} = \omega \cdot V_p^t + c_1 \cdot r_1 (l_p - r_p^t) + c_2 \cdot r_2 (G - r_p^t)$$

Where for particle p

ω	: is the inertia of the particle
V_p^t	: is the velocity at time t of this particle.
r_p^t	: is the position at time t of this particle.
l_p	: is the best solution found by this particle thus far (local memory)
c_1, c_2	: are parameters representing a particle's - trust- into itself and the swarm, respectively.
r_1, r_2	: are random numbers in the range (0; 1).

b) For each particle p , update its position at time $t + 1$ according to

c)

$$r_p^{t+1} = r_p^t + V_p^{t+1}$$

c) For each particle p , score the position vector r_p^t in the context of the problem being addressed and update:

l_p	: if the current position produces the best local score thus far.
G	: if the current position of produces the best global score thus far.

3) **Output:**

The global best position G

3. Wind Power Turbine

The wind turbine is a device for conversion of kinetic energy in the wind into electricity. Although there are many different configurations of wind turbines systems they all work in the same way. The available power originates from the mass flow of the moving air, referred to as the wind speed. The transformation to mechanical torque is done by aerodynamical forces acting on the rotors blades, the actuator disc. The wind turbine shaft then transports the power to the generator which is connected to the electrical grid.

Usually there is a gearbox between the slowly rotating turbine shaft and the more rapidly rotating generator shaft. This is described by the model shown in Figure 1.

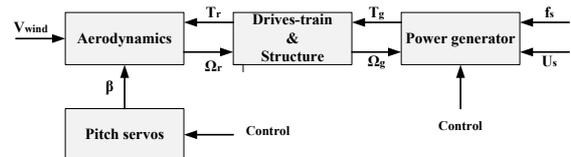


Fig. 1. Subsystem-level block diagram of a variable-speed variable-pitch Wind Energy Conversion Systems

As the incoming wind contains the energy input to the system, the power output is dependent on the wind speed. A lowest and highest wind speed, WS_{cut-in} respectively $WS_{cut-out}$, determines the range in which wind turbine can operate as shown in Figure 2.

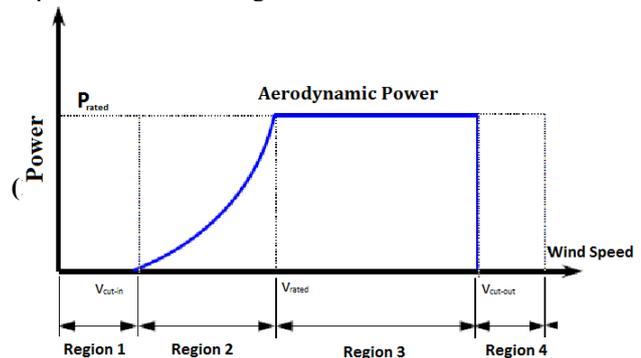


Fig. 2. Wind power, turbine power, and operating regions

The wind speeds that are considered as the boundaries of this division are cut-in wind speed v_{cut-in} , rated wind speed v_{rated} and cut-out wind speed $v_{cut-out}$.

For this paper, the values of v_{cut-in} , $v_{cut-out}$ and v_{rated} respectively are 3 m/s, 11.4 m/s and 25 m/s

According to Figure 2, the operations of the wind turbine can be divided into four regions.

In region I, the wind speed is lower than v_{cut-in} and due to low wind speed; the wind turbine is at stop state. Thus, the generator torque is zero and wind turbine cannot generate power.

In region II that is called partial load region, the wind speed is higher than v_{cut-in} but lower than v_{rated} . The main control objective in this region is to maximize power generated by the wind turbine. In order to capture the maximum power, power coefficient C_p should be set to maximum power coefficient C_p . This is achieved by keeping blade pitch approximately constant and using generator torque to vary the rotor speed. With small pitch changes about the optimal

angle, a controller can also reduce dynamic loads in the structure

In region III that is called full load region, the wind speed is upper than v_{rated} but lower than the $v_{cut-out}$. The main control purpose in this region is to keep the generator power P_g around the rated generator power P_g ,

In region IV that the wind speed is upper than $v_{cut-out}$, in order to protect wind turbine against the stresses and fatigue damages, the wind turbine must be shut down. In this case, power generation is stopped.

1) In the low-wind speed course, it operates with the variable speed. By adjusting the reverse torque of generator, the rotate speed can vary with the wind speed until it has got to the upper limit. During the period, the tip speed ratio λ keeps invariant. So, it makes the wind power utilizing coefficient C_p be a constant.

2) Since the rotor speed has gone to critical, if wind speed increases, the wind turbine should operate at a constant speed until the largest output is realized. In this period, C_p may not be the largest.

3) When the wind speed is larger than the rated, the energy captured by the turbine should be confined through adjusting the paddle pitch. Then the output of wind turbine will keep the maximum level.

Individual pitch control strategies have been examined for some time, e.g. [11-21]. Some of the available power in the wind is converted by the rotor blades to mechanical power acting on the rotor shaft of the wind turbine. For steady-state calculations of the mechanical power from a wind turbine, the so called $C_p(\lambda, \beta)$ -Curve can be used. The mechanical power, P_a , can be determined by [22]

The output power of the turbine is given by the following equation:

$$P_a = \frac{1}{2} \rho \pi R^2 C_p(\lambda, \beta) v^3 \quad (3)$$

$$T_t = \frac{P_a}{\Omega_t} \quad (4)$$

C_p value is calculated using a generic equation proposed in [22] given by

$$C_p(\lambda, \beta) = c_1 \left(\frac{c_2}{\lambda_i} - c_3 \cdot \beta - c_4 \right) e^{-\frac{c_5}{\lambda_i}} + c_6 \cdot \lambda \quad (5)$$

Where

$$\frac{1}{\lambda_i} = \frac{1}{\lambda + 0.08 \cdot \beta} - \frac{0.035}{1 + \beta^3} \quad (6)$$

where C_p is the power coefficient, β is the pitch angle, λ is the tip speed ratio, w is the wind speed, ΩT is the rotor speed of the turbine (on the low-speed side of the gearbox), r is the rotor-plane radius, ρ is the air density and A is the area swept by the rotor. The produced torque T_T , of the turbine can be found from

The $C_p(\lambda, \beta)$ -curve for the wind turbine investigated here can be seen in Figure 3.

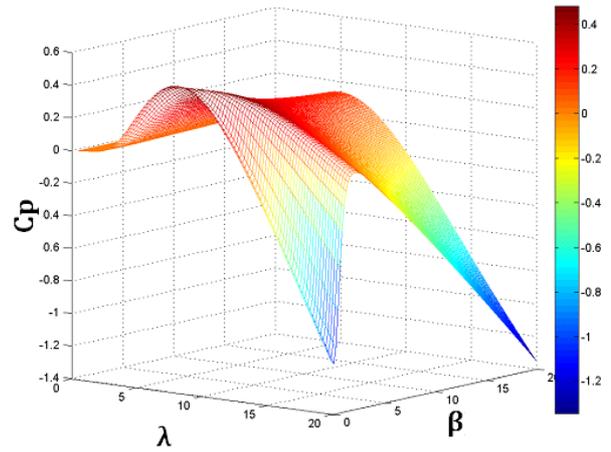


Fig. 3. Power coefficient as a function of the tip-speed ratio and pitch angle

4. Results and Discussion

The following gives the design step for the PSO algorithm:

Step 1. Initialize a population of particles with random positions and velocities in N dimensions of the problem space and fly them.

Step 2. Evaluate the fitness of each particle in the swarm.

Step 3. For every iteration, compare each particle's fitness with its previous best fitness P_{best} obtained. If P_{best}^{t+1} value is better than P_{best}^t , then set $P_{best} = P_{best}^{t+1}$ and the P_{best} location equal to the current location in the N -dimensional space.

Step 4. Compare P_{best} of particles with each other and update the swarm global best location with the greatest fitness (G_{best}).

Step 5. Change the velocity and position of the particle

According to equations (1) and (2) respectively

$$v_i(t+1) = \omega \cdot v_i(t) \cdot r_1 \cdot c_1(p_i - X_i(t)) + r_2 \cdot c_2(g - X_i(t)) \quad (7)$$

$$X_i(t+1) = X_i(t) + v_i(t+1) \quad (8)$$

Where: $v_i(t)$ and $X_i(t)$ represent the velocity and position of the i_{th} particle with N dimensions, respectively.

ω is the inertia weight, which is chosen beforehand. And r_1 and r_2 are randomly generated values in the range of $[0-1]$, and ω is the inertia weight, which is chosen beforehand.

Step 6. Repeat steps (1) to (5) until convergence is reached based on some desired single or multiple criteria

We simulated a pitch-controlled variable-speed wind turbine operating in turbulent winds. We described the philosophy behind the control strategy. Slow and fast pitch rates were investigated, and their impacts were shown on the operating characteristics of the wind turbine.

Figure 4 shows the evolution of C_p with PSO Controller

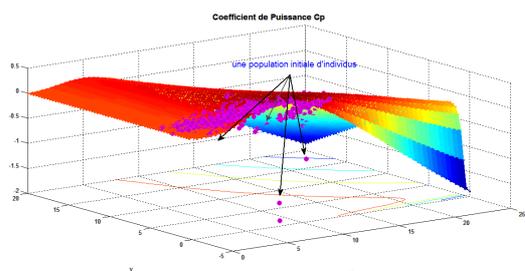


Fig. 4-a. Cp with PSO Controller $t=0$

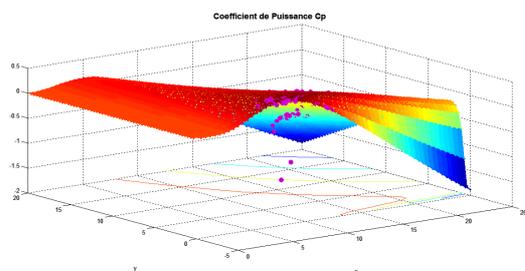


Fig. 4-b .Cp with PSO Controller $t=ti$

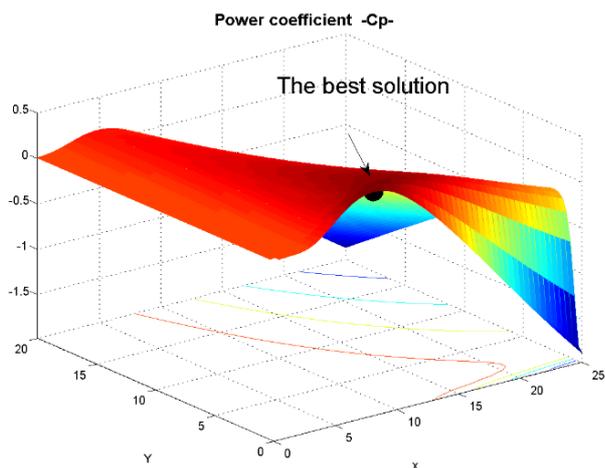


Fig. 4-c Cp with PSO Controller $t=t_{final}$

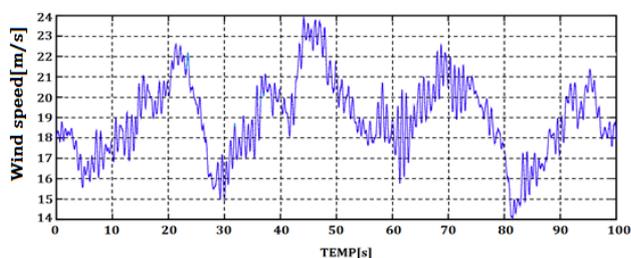


Fig. 5.a Wind speed

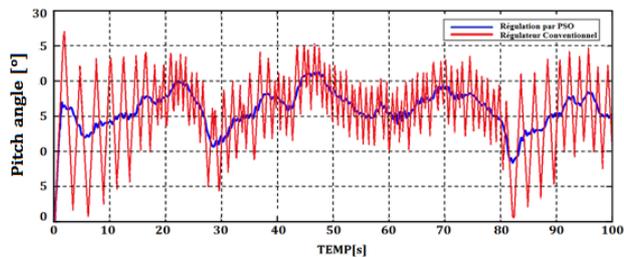


Fig. 5.b Pitch angle β

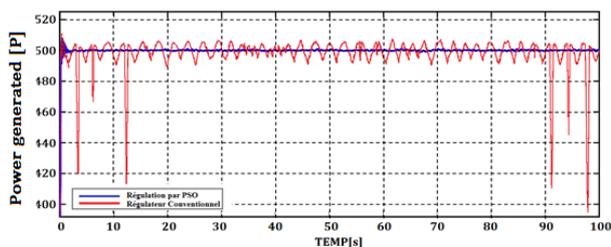


Fig. 5.c Generated power

Simulations have been used to demonstrate PSO controller capabilities of wind power Figure 5.

In PSO-based control, output power and rotor speed reach the rated value quickly

The performance coefficient, tip speed ratio and pitch angle were changed according to the trained PSO controller outputs. With PSO regulator, wind turbines used for power generation is able to produce maximum power by making the most of available energy in the wind.

5. Conclusion

In this work, the speed control of a modern pitch-regulated wind turbine is investigated. A controller structure is derived and the significance of various parameters in the controller structure is investigated.

Considering our results, we draw the following conclusions.

- In the lower wind-speed region, maximum operating mode is adapted. The pitch angle is kept at its optimum operation. The average is shown to be maximum as the wind speed varies.
- In the higher wind-speed region, the rotor speed must be controlled to avoid rotor speed increases beyond a controllable limit.

PSO algorithm is another example of successful artificial-engineering swarm intelligence system.

References

1. A. Larsson, P. Sørensen, and F. Santjer, "Grid impact of variable speed wind turbines," in Proc. of European Wind Energy Conference and Exhibition (EWEC '99), Nice, France, Mar., 1–5, 1999.
2. Measurement and assessment of power quality characteristics of grid connected wind turbines (11/2000), International electrotechnical Commission Std. IEC 61 000-21, 2000.
3. E. Muljadi and C. P. Butterfield, "Annualized wind energy improvement using variable speeds," IEEE Trans. Ind. Applicat., vol. 37, no. 1, pp. 240–246, Jan./Feb. 2001.
4. G. L. Johnsson, Wind Energy Systems. Englewood Cliffs, N.J., USA.: Prentice-Hall, 1985.
5. N. Horiuchi and T. Kawahito, "Torque and power limitations of variable speed wind turbines using pitch control and generator power control," in IEEE Power Engineering Society Summer Meeting, vol. 1, Vancouver, USA, July, 15–19, 2001, pp. 638–

- 643.
7. J. Kennedy, R.C. Eberhart, Particle swarm optimization, in: Proc. IEEE Internat. Conf. on Neural Networks, Perth, Australia,
 8. vol. IV, IEEE Service Center, Piscataway, NJ, 1995, pp. 1942–1948.
 9. I.C. Trelea, The particle swarm optimization algorithm: convergence analysis and parameter selection, Inform. Process. Lett. 85 (6) (2003) 317–325
 10. J. Kennedy, The Particle Swarm: social adaptation of knowledge, in: Proc. IEEE Internat. Conf. on Evolutionary Computation, Indianapolis, IN, IEEE Service Center, Piscataway, NJ, 1997, pp. 303–308.
 11. M. Clerc, J. Kennedy, The particle swarm: explosion stability and convergence in a multi-dimensional complex space, IEEE Trans. Evolutionary Comput. 6 (1) (2002) 58–73.
 12. R.C. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: Proc. 6th Internat. Symp. on Micro Machine and Human Science, Nagoya, Japan, IEEE Service Center, Piscataway, NJ, 1995, pp. 39–43.
 13. V. Engelen T G 2006 Proceedings of the European Wind Energy Conference and Exhibition (EWEC)
 14. Wright A D and Stol K A 2010 48th AIAA Aerospace Sciences Meeting
 15. K A Stol and Balas M J 2003 Journal of Solar Energy Engineering 125 379
 16. M. Geyler and Caselitz P 2007 Proceedings of EWEC
 17. Bossanyi E 2003 Wind Energy 6 119–128
 18. Selvam K, Kanev S, van Wingerden J W, van Engelen T and Verhaegen M 2009 International Journal of Robust and Nonlinear control 00 1–22
 19. Wright A D, Fleming P and van Wingerden J W 2011 49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition
 20. Larsen T J, Madsen H A and Thomsen K 2005 Wind Energy 8 67–80
 21. Kanev S and van Engelen T 2009 European Wind Energy Conference 2009 New Horizons Forum and Aerospace Exposition
 22. Bossanyi E A 2005 Wind Energy 8 481–485
 23. Caselitz P, Kleinkauf W, Krüger T, Petschenka J, Reichard M and Störzel K 1997 European Wind Energy Conference 1997, Proceedings
 24. El Aimani, “ Modélisation de différentes technologies d'éoliennes intégrées dans un réseau de moyenne tension”, Ecole Centrale de Lille, Université des Sciences et Technologies de Lille, 2004