Optimizing wind turbine maintenance using artificial intelligence: Particle Swarm Optimization (PSO).

Tsartsari..

June 26, 2023

Math department , University of Ioannina, ma11481@uoi.gr

Abstract

Abstract: In this paper, we present an application of artificial intelligence (AI) to optimize wind turbine maintenance. Wind energy is one of the most used and developed as renwable energy, since it is a cost-effective way to generate clean and sustainable energy. The system used Particle Swarm Optimization (PSO) a computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. Moreover, the significance of the operation and maintenance costs of wind farms becomes even greater if we consider the latest tendency to increase the efficiency of Particle Swarm Optimization for the optimal cost achieved. Finally, the influence of the model parameters is studied by performing a sensitivity study, that shows the importance of preventive maintenance and the reduction of corrective maintenance tasks.

Contents

1 Introduction

2 Literature Review

3 Description of the algorithm

4 Particle Swarm Optimization Algorithm

5 Conclusions

1. INTRODUCTION

The wind energy requirement is at its peak and increases exponentially over time. Conventional energy sources cannot fulfill the demand along with it contributing a major share in global warming and pollution. Renewable energy sources (RES) are replacing conventional sources to keep the world green and clean. Wind energy is a prominent source among RES due to its abundant nature. A hybrid energy system model which also includes wind source is presented by integrating various energy storage systems. Grid integration with wind turbines (WT) as well as wind's nonlinear nature makes the control more challenging. Maintaining power stability along with running the WT in a safe mode is quite a difficult task. The WT operation is mainly divided into three different regions according to the variable wind speed. The speed below 3m/s is said to be in the cut-in region. The wind speed from 3m/s to 25m/s belongs to the normal operating region, while the region in which the speed is above 25m/s is called the cut-out region. The wind turbines cannot operate in a cut-in region due to economic scenarios and in the cut-out region for safety purposes. At the above rated speed, the pitch angle adjust the blades to extract optimal power and keep the wind turbines safe. Multiple controlling techniques have been developed to control the pitch angle. In resent times, the particle swarm optimization (PSO) is widely used for the optimization of PID parameters. Gambier and Bahera used the multi-objective PSO for the integrated pitch control system by combining the individual and collective picth control. The power extracted from the hybrid energy sources such as wind and solar is optimized by employing PSO and the cost of energy is reduced. The amount of RES is optimized by the addition of energy storage and demand response in the system. For feedback control, the PSO methodology is applied to acquire the maximum power from wind energy. Normally, the PSO-based optimization for complex systems, such as the WT is not much accurate. Hence, there is a need to develop an algorithm that can rectify this problem and provide optimum parameters for the PID controller to regulate the pitch angle more effectively as compared to the conventional PSO. The proposed technique is deployed on a 5-MW offshore WT. The system is linearized at different wind speeds to investigate the working of the prescribed methodology.

2. LITERATURE REVIEW

a. "Particle Swarm Optimization of a hybrid wind/tibal/PV/battery energy system. Application to a remote area in Bretagne France", Energy Proced 2019, Mohammed OH, Amirat Y. and Benbouzid M.

-The ability of PSO was developed to reach the best results in double speeds at a time rate better than 80% of conventional technology time and less than 20 repetitions only. The (PSO) developed has several characteristics and advatages over other traditional techniques and algorithms. In fact it allows to achieve the optimal solution and to reduce the overall cost with high speed and accuracy. The PSO algorithm program was developed using MATLAB software.

b. "A novel approach to capture the maximum power from variable speed wind turbines using PI controller, RBF neural network and GSA evolutionary algorithm", Renew Sust Energy Rev. 2015, Assareh E and Biglari M.

- A 5MW wind turbine model software code developed at the US National Renewable Energy Laboratory is used to simulate and verify the results.In the proposed method, thw wind turbine generator torque is regulated using a proportional ans integral (PI) controller.The optimal dataset to train this neural network is provided by the Gravitational Search Algorithm. The results show that the proposed method has a good performance.

c. "Knowing about wind turbines", Energy Press, 19/08/22, Giannis Bourdoubas.

- The construction cost of offshore wind turbines amounts to approximately 1.3 million euros per MW of installed capacity. A 2-3 MW wind turbine costs around 2.6-4 million euros. The maintenance cost of offshore wind turbines amounts to approximately 0.01-0.02 euros per KWh produced. Offshore wind turbines cost more to built than their onshore counterparts , but an offshore wind turbine can produce 1.5 to 2 times more electricity than a onshore one.

3. DESCRIPTION OF THE ALGORITHM

Particle Swarm Optimization was proposed by Kennedy and Eberhart in 1995. Sociologists believe a school of fish or a flock of birds that moves in a group can profit from the experience of all other members.In other words, while a bird flying and searching randomly for food, for instance, all birds in the flock can share their discovery and help the entire flock get the best hunt. While we can simulate the movement of a flock of birds, we can also imagine each bird is to help us find the optimal solution in a high-dimensional solution space and the best solution found by the flock is the best solution in the space. This is a heuristic solution because we can never prove the real global optimal solution can be found and it is usually not. However, we often find that the solution found by PSO is quite close to the global optimal. In computational science , particle swarm optimization is a computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure. It solves a problem by having a population of candidate solutions, here dubbed particles and moving these particles around in the search-space according to simple mathematical formula over the particle's position and velocity. Each particle's movement is influenced by its local best known position, but is also guided toward the best known positions in the search-space, which are updated as better positions are found by other particles. This is expected to move the swarm toward the best solution.

PSO is a metaheuristis as it makes few or no assumptions about the problem being optimized and can search very large spaces of candidate solutions. Also, PSO, does not use the gradient of the problem being optimized, which means PSO does not require that the optimization problem be differentiable as is required by classic optimization methods. However, metaheuristics such as PSO do not guarantee an optimal solution is ever found.

4. PARTICLE SWARM OPTIMIZATION ALGORITHM

```
import random


class Particle:
def__int__(self, x0):
self.position_i = []
self.velocity_i = []
self.pos.best_i = []
self.err_best_i = []
self.err_i = -1


for i in range(0, num_dimensions):
self.velocity_i.append(random.uniform(-1,1))
self.position_i.append(x0[i])
def evaluate(self, costFunc):
self.err_i = costFunc(self.position_i)
if self.err_i < self.err_best_i or self.err_best_i == -1;
self.pos.best_i = self.position_i
self.err_best_i = self.err_i
def update_velocity(self, pos_best_g):
w = 0.5
c1 = 1
c2 = 2
for i in range(0,num_dimensions):
r1 = random.random()
r2=random.random()
vel_cognitive = c1* r1 * (self.pos_best_i[i] - self.position_i[i])
vel_social = c2 * r2 * (pos_best_g[i] - self.position_i[i])
self.velocity_i[i] = w * self.velocity_i[i] + vel_cognitive + vel_social
update_position(self, bounds):
for i in range(0, num_dimensions):
self.position_i[i] = self.position_i[i] + self.velocity_i[i]
if self.position_i[i] > bounds[i][1]:
self.position_i[i] = bounds[i][1]
if self.position_i[i] < bounds[i][0]:
self.position_i[i] = bounds[i][0]
class PSO():
def __init__(self, costFunc , x0 , bounds , num_particles, maxiter):
```

```
global num_dimensions
num_dimensions = len(x0)
err_best_g = -1
pos_best.g = []
swarm = []
for i in range (0 , num_particles):
swarm.append(Particle(x0))
i=1
while i < maxiter:
for j in range(0, num_particles):
swarm[j].evaluate(costFunc):
if swarm[j].err_i <err.best_g == -1;
pos_best_g = list(swarm[j].position_i)
err_best_g = float(swarm[j].err_i)
for j in range(0 , num_particles):
swarm[j].update_velocity(pos_best_g)
for j in range(0 , num_particles):
swarm[j].update_position(bounds)
i+=1
print('Best position found:  ',pos_best_g)
print('Best error found)
```

5. CONCLUTIONS

Other algorithms are not precise and efficient for system stability. The proposed PSO algorithm is used for this purpose in comparison with the conventional technique. The proposed technique is employed on a 5- MW WT. The transfer function of the designed WT is acquired using the generator demanded and measured generator speed as output. Parameters of the PID are optimized by employing the proposed technique at different wind speeds.The results acquired using the proposed methodology stabilizes the system more quickly and effectively. The system stability is verified with the Lyapunov function. The results demonstrate that the proposed PSO technique yields higher output power and limits the rotor speed along with the mechanical torque, thus providing efficient regulation of pitch angle. The stability of the system can be increased. This could be possible by further reducing the settling time. Multiple approaches may be utilized by combining the pattern search with the existing techniques.

6.REFERENCES

a. Soufi Y . Kahla S. and Bechouat M. Feedback linearization control based particle swarm optimization for maximum power point tracking of wind turbine equipped by PMSG connected to the grid. Int J Hydrogen Energy 2016.

b.Tahir MF , Haoyong C., Khan A. Optimizing size of variable renewable energy sources by incorporating energy storage and demand response. IEEE Access 2019.

c.Olondriz J., Jugo J., Elorza I. A feedback control loop optimization methodology for floating offshore wind turbines. Energies 2019.

d.Zammani M., Karimi-Ghartemani M., Sadati N. Design of a fractional order PID controller for an AVR using particle swarm optimization. Control Eng Pract 2009.

e.Jabbari Asi H and Yoon J. Power capture optimization of variable-speed wind turbines using an output feedback controller. Renew Energ 2016.