Journal of Theoretical and Applied Information Technology
30th June 2021. Vol.99. No 12

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

HYBRID SOLUTION FOR WIND TURBINES POWER CURVE MODELING FOUNDED ON CASE BASED REASONING, MULTI-AGENT SYSTEM AND THE K-NEAREST NEIGHBORS ALGORITHM

[1]MOHAMED KOUISSI, [2]EL MOKHTAR EN-NAIMI, [3]ABDELHAMID ZOUHAIR

[1,2,3]University of Abdelmalek Essaâdi (UAE), Faculty of Sciences and Technologies of Tangier, Computer Science Department, LIST laboratory, Morocco

E-mail: [1]mohamed.kouissi@gmail.com, [2]en-naimi@uae.ac.ma, [3]zouhair07@gmail.com

ABSTRACT

The aim of the wind turbines power curve is to represent the performance of a wind turbine, aids in wind power assessment and also helps in wind power forecasting. The wind turbines power curve captures the nonlinear relationship between wind speed and output power. In this paper, we present a hybrid approach of wind turbines power curve modeling based on Case Based Reasoning approach, multi agent system and a machine learning algorithm, which is the K-Nearest Neighbors method to propose a new adapted wind turbines power curve for our target case based on the wind turbines power curve of similar wind turbines. The K-Nearest Neighbors algorithm is used in the retrieve step of the case based reasoning cycle to search for similar wind turbines based on their characteristics. These wind turbines are then classified and sorted on the basis of features similarity measure. Then, a new wind turbines power curve of the target case is proposed based on the experiences of similar cases.

Keywords: *Case Based Reasoning (CBR), Multi Agents System (MAS), Wind Turbines Power Curve (WTPC), K-Nearest Neighbors algorithm (KNN), Modeling.*

1. INTRODUCTION

Renewable energies play a very important role in the development and economic growth of all countries. However, the use of these resources is rising more and more due to population growth and the improvement in standard living of citizens, which leads to its overexploitation. To ensure a secure and sustainable energy supply, most countries seek to improve their capacity to produce energy by using renewable resources to provide people with reliable and efficient energy services. Among all these resources, wind power can play a major role in restarting economies around the world. It is very promising because the technology is commercially mature, and it can deliver clean and affordable energy. The global wind power capacity is on-track to achieve to nearly 1,000 GW by the end of 2024, which is an increase of 54 per cent for total wind power installations compared to 2019 [1].

Before starting the installation of wind turbines, it should first to study their performance. a wind turbine power curve (WTPC) shown in Fig1is used in this case. namely the input-output relationship between the generated wind turbine power and the wind speed. It can be also used for power assessment and forecasting, wind turbine selection and capacity estimation [22]. The wind turbine power curve is essentially comprised of three regions [25, 26]. The first region when the wind speed is less than the cut-in speed (Uc) doesn't generate any wind power, so it has zero output. The second region, when the wind speed is located between Uc and Ur, the output power increases as the wind speed increases. The third region is when the wind speed is greater than the Ur and less than the cutout speed Us, in this case the output power remains constant. The Fourth region is when the wind speed is larger than the Us, the wind turbine will be shut down for security concerns.

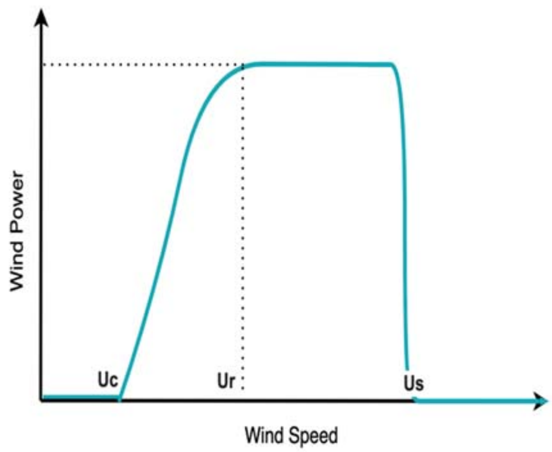

Figure 1. Wind Turbines Power Curve

Since wind speed is stochastic, WTPCs play an important role in modeling and forecasting the energy produced by wind turbines and also to monitor their state of health [2]. The estimation and simulation of WTPC must be done from the actual operational data of the wind turbines already installed.

Modeled power curves can serve as a solid basis of comparison between the performances of the different available turbines, and also helping customers to make the right choice according to their needs and requirements [3].

To achieve smart monitoring of wind power forecasting, we need to benefit from the past experiences of already installed wind turbines and reuse their WTPC to predict a new one according to new data input as wind speed data. In this paper, we propose a hybrid approach of wind turbines power curve modeling based on Case Based Reasoning approach and a machine learning algorithm, which is the K-Nearest Neighbors method (KNN) to propose a new adapted WTPC based on the WTPC of other wind turbines

The structure of this paper is arranged as follows: In section two, we present the literature review of which we introduce the main concepts related to our approach. Section three is devoted to describe and detail the different steps of our approach. In section four, we present the result of our approach using the CBR and the KNN algorithm. In the last section, we sum-up with a conclusion, limitation and future direction.

2. LITERATURE REVIEW

2.1 Case Based Reasoning

The Case based Reasoning (CBR) is a problem-solving paradigm that solves a problem based on similar past experiences [4]. It takes its origins from Schank's memory [6]. It is one of the evolving approaches to design and develop intelligent decision-making systems. CBR has been used to solve many problems in different fields like Learning System, Recipe generation, Music etc… In CBR, new solutions for a current situation are generated by retrieving the most similar cases from the case base and adapting those cases to current contexts [5]. To Solve a problem using the Case Based Reasoning approach, we follow a typical cycle with a set of five steps as shown in figure Fig.1: Elaboration, Retrieve, Reuse, Revise and Retain [7, 8, 27]. The first step of CBR consists in building and setup the specification of the problem to be solved. Information and knowledge are inferred from an initial request submitted to the system, this gives a detailed description of the target problem to better guide the search for a specific objective and task. In the second step, the elaborated target problem is then used to find a set of cases in the case base that are judged similar. During the Reuse step, the solutions of the selected cases are adapted and reused to obtain solutions to the target case. In the revision step, the proposed solutions will be analyzed, modified, refused or accepted by the user. Finally, the Retain step is the step during which the greatest number of knowledge is likely to emerge. It allows to incorporate whether the knowledge learned from the already solved new case is useful to be incorporated into the case base [5].

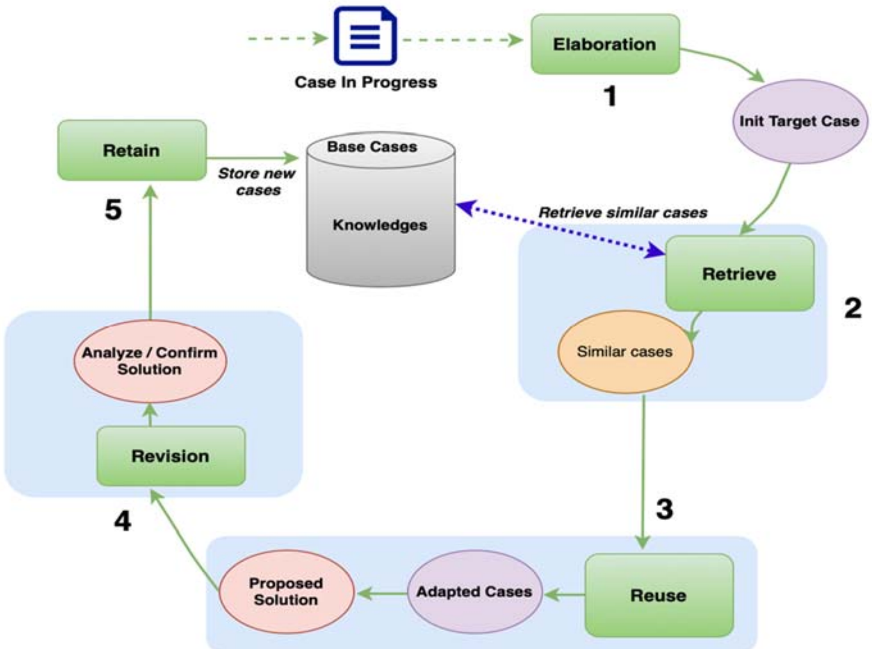

Figure 2. Classic CBR cycle inspired from [4]

2.2 Classification with k-nearest Neighbors

The K-Nearest Neighbors (KNN) algorithm is a super simple way to classify data. It's a non-parametric, supervised machine learning algorithm that can be used for both classification and regression problems. The aim of this algorithm is to classify new objects based on attribute and training data [9]. Each object is classified by looking at the k nearest neighbors. The K-Nearest Neighbors algorithm has the following steps:

- First, Specifying the number of nearest neighbors (K) to use.
- Specifying the training data set D(yj).
- Calculate the distance d(xi, yj) between the new observation Xi and the whole of training data (yj). In this case, there are many important methods to calculate this distance. By default, the KNN algorithm uses the Euclidean distance which is calculated by the following equation

$$D(x,y) = \sqrt{(\boldsymbol{x0} - \boldsymbol{y0})^2 + (\boldsymbol{x1} - \boldsymbol{y1})^2 + \cdots + (\boldsymbol{xn} - \boldsymbol{yn})^2} \quad (1)$$

X and Y are subjects with n characteristics to be compared [10]. There is also another method to calculate this distance such as Manhattan distance [11].

- Sort result of distances in ascending order.
- Get the first K entries of the result distances
- Classify the observation Xi according to the class of its neighbors that gets the most votes.

KNN has several advantages, we cite:

- It's very simple and easy to implement.
- KNN is versatile, we can use it for classification, regression and search.

Meanwhile, the KNN algorithm has some limitations, we cite:

- It gets significantly slower as the data set elements and characteristics increase.
- High cost computing of the calculation of the distance between the new observation Xi and in the whole data set training.

2.3 Multi Agent System

The multi agent system (MAS) is a system composed of a collection of intelligent agents [24], operating in an environment, interacting with each other and with the environment to solve a common problem and to achieve desired goals [15, 16]. They are able to learn and act in their environment as presented in [17]. Agents may behave towards each other as collaborators, competitors or strangers [18, 19]. A System multi agent is composed of [20]:

- The environment.
- Agents and the relationships that link them with each other in their environment.
- Behaviors and Operations of each agent.
- Objects that can be created, modified and perceived by agents.

3. OUR PROPOSED APPROACH AND ITS ARCHITECTURE

Our approach involves the use of the multi-Agents system and the case based reasoning approach for the resolution of the power curve modeling problem.

The aim of our approach is to create a multi-agents system for smart monitoring of wind turbines power forecasting and which is based on past experiences of already installed ones. This approach allows decision makers to simulate the WTPC of new installed wind turbine and analyze its performance in real time. A K-Nearest Neighbors machine learning algorithm is used to build our power curve model.

Our approach has many positive points as follow:

- Smart management system to ensure in performance monitoring of the turbine.
- Forecasting of power of newly installed wind turbines.
- Aids in wind power assessment.
- Benefit from past experiences.
- Build a model using machine learning algorithm.

Our system architecture contains five steps. Schematically, we can represent our proposed approach as indicated in Figure 3 below:

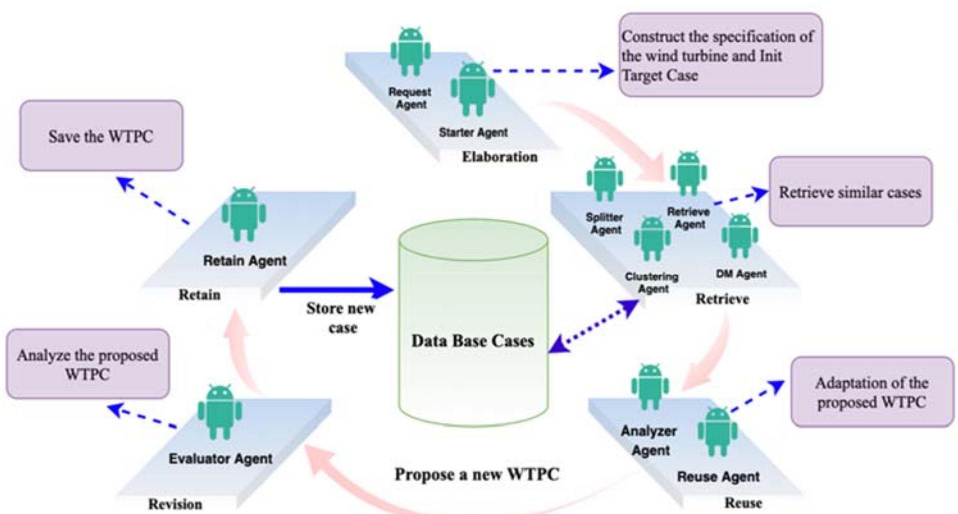

Figure 3. Our Proposed Architecture

- Elaboration step: The aim of this step is to construct the target case based on the specification of the wind turbine data. This step contains the following agents:

  - **Request Agent**: This agent is the interface between the system and the environment. It's responsible for cleaning and processing the collected wind turbine data. This serves to better guide the search for similar cases to the target case.

  - **Starter agent**: this agent is responsible for constructing and initializing the wind turbine target case. Each wind turbine is modeled as a case presented by a vector ***Casei*** which describes the characteristic of the wind turbine.

$$\boldsymbol{Case}_i = \begin{pmatrix} n\_power \\ r\_diameter \\ h\_height \\ p\_density \\ pcw\_speed \\ pc\_pow\_value \end{pmatrix} \quad (2)$$

  - nominal_power: Represents the nominal power of the wind turbine.
  - rotor_diameter: Represents the rotor diameter.
  - hub_height: The rotor's height above ground.
  - power_density: The power density parameter.
  - power_curve_wind_speeds: a list of wind speed values
  - power_curve_values: a set of power wind values according to wind speed.

- Retrieve step: This step contains a set of agents that compare and retrieve the most appropriate cases of the target case. The Retrieve step contains the following agents:

  - **The Retrieve agent**: The goal of this agent is to collect the elaborated target case from the starter agent of the Elaboration step. This agent is also responsible for evaluating and finding a set of cases in the case base that are judged similar to the target case using similarity measure. To search for similar cases, we used the K-Nearest Neighbors (KNN) algorithm, the following steps will be executed:
    a. Split our dataset into inputs (X) and target (Y). Our input contains three parameters: the nominal power, rotor diameter and the hub height of the wind turbine. The power density of the wind turbine will be the target that we will be attempting to predict to find similar cases.
    b. Split our dataset into training and testing data. The training dataset is used to train our model, and the testing dataset is used to see the result of our model on unseen data.
    c. Find the good value K of the number of nearest neighbors to use.
    d. Calculate Euclidean distance between the target case and all the cases in the database.
    e. Sort the obtained result on the basis of distance.
    f. Extract the top K neighbors.

  - **Splitter Agent**: The role of this agent is to execute the steps a and b above of the KNN algorithm.
  - **Clustering Agent**: This agent executes the step c above of the KNN algorithm.
  - **Distance Measure Agent**: This agent is responsible to execute the steps d and e above.

- Reuse step: The agents of this step aims to adapt and reuse all cases deemed similar to our target case. This step contains the following agents:
  - **Reuse Agent**: The reuse agent aims to adapt the solutions of the K nearest neighbors cases proposed by the retrieve agent and reuse them to obtain a solution for the target case. In our case, the solution is a function that captures the relationship between wind speed and wind power [23].
    - **Analyzer agent**: This agent check in a dynamic way if there is a change in the target case, then the analyzer agent asks the starter agent of the Elaboration step to update the target case. This process is continuous and will be triggered whenever any updates or modifications are made to the target case.

- Revision step: The goal of the agent of this step is to analyze and evaluate the proposed solution by the reuse agent, make a modification if necessary, then refuse or accept it.
- Retain step: In this step, the retain agent retains the validated solution as a new learnt case and store it into the case bases for future reuse. The proposed WTPC for the target case will be used

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

then to solve future power curve modeling problem.

4. RESULTS AND DISCUSSIONS

In this section, we present the result of our approach for the resolution of the power curve modeling problem using the KNN algorithm in the retrieve step of the CBR cycle. The KNN algorithm has been applied to find the nearest wind turbines of our wind turbine target case. To determine an optimal value k of the number of neighbors, we used the Elbow method [14] which is one of the most popular methods to find this value of k.

4.1 Data Processing and Presentation

The modeling of the WTPC has been done using a set of 67 wind turbine after the clean of abnormal or missing values from an initial data of 137 element. Each wind turbine is represented by a vector Casei of characteristics (nominal power, rotor diameter, hub height, power density, list of power curve wind speed, list of power curve values). For a better visualization, we will present the data in two tables. The table 1 contains the first four parameters. Otherwise, the table 2 contains a set of vector values that captures the relationship between wind speed and wind power of power curve.

$$\boldsymbol{Case_i} = \begin{pmatrix} n\_power \\ r\_diameter \\ h\_height \\ p\_density \\ pcw\_speed \\ pc\_pow\_value \end{pmatrix} \quad (3)$$

	nominal_power	rotor_diameter	hub_height	power_density
0	4200	141	129.0	269.00
1	4200	127	135.0	331.00
2	3500	101	99.0	436.80
3	3200	116	122.0	304.40
4	3050	101	124.0	380.70
..	...	...	...	...
62	1800	100	110.0	225.00
63	3600	117	99.0	307.00
64	8000	168	104.0	302.00
65	7500	127	125.0	1.69
66	7580	127	127.0	345.00

Table 1. The first four Parameters of the vector ***Casei***

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

```
                                           power_curve_wind_speeds
0   [1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0, 11.0, 12.0, 13.0, 14.0, ...
1   [1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0, 11.0, 12.0, 13.0, 14.0, ...
2   [1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0, 11.0, 12.0, 13.0, 14.0, ...
3   [1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0, 11.0, 12.0, 13.0, 14.0, ...
4   [0.0, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0, 6.5, 7.0, ...
..                                                                            ...
62  [3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0, 6.5, 7.0, 7.5, 8.0, 8.5, 9.0, 9.5, 10.0,...
63  [3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0, 6.5, 7.0, 7.5, 8.0, 8.5, 9.0, 9.5, 10.0,...
64  [1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0, 11.0, 12.0, 13.0, 14.0, ...
65  [0.0, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0, 6.5, 7.0, ...
66  [0.0, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0, 6.5, 7.0, ...

                                                 power_curve_values
0   [0.0, 22.0, 104.0, 260.0, 523.0, 920.0, 1471.0, 2151.0, 2867.0, 3481.0, 3903...
1   [0.0, 0.0, 58.0, 185.0, 400.0, 745.0, 1200.0, 1790.0, 2450.0, 3120.0, 3660.0...
2   [0.0, 3.0, 37.0, 116.0, 253.0, 469.0, 775.0, 1175.0, 1680.0, 2280.0, 2810.0,...
3   [0.0, 3.0, 49.0, 155.0, 339.0, 628.0, 1036.0, 1522.0, 2215.0, 2677.0, 3030.0...
4   [0.0, 0.0, 0.0, 0.0, 3.0, 22.0, 49.0, 92.0, 155.0, 240.0, 339.0, 480.0, 628....
..                                                                            ...
62  [12.0, 40.0, 93.0, 161.0, 241.0, 336.0, 446.0, 575.0, 727.0, 900.0, 1094.0, ...
63  [22.0, 78.0, 150.0, 237.0, 340.0, 466.0, 617.0, 796.0, 1006.0, 1247.0, 1522....
64  [0.0, 0.0, 0.0, 100.0, 500.0, 1000.0, 2000.0, 3000.0, 4000.0, 5000.0, 6000.0...
65  [0.0, 0.0, 0.0, 0.0, 0.0, 25.0, 55.0, 110.0, 175.0, 290.0, 410.0, 580.0, 760...
66  [0.0, 0.0, 0.0, 0.0, 0.0, 25.0, 55.0, 110.0, 175.0, 293.0, 410.0, 585.0, 760...
```

Table 2. Data that captures the relationship between wind speed and wind power

.

The Figure 4 generated by the Matplotlib library [12] in python programming language [13] shows the distribution of our wind turbine data according to the first four parameters.

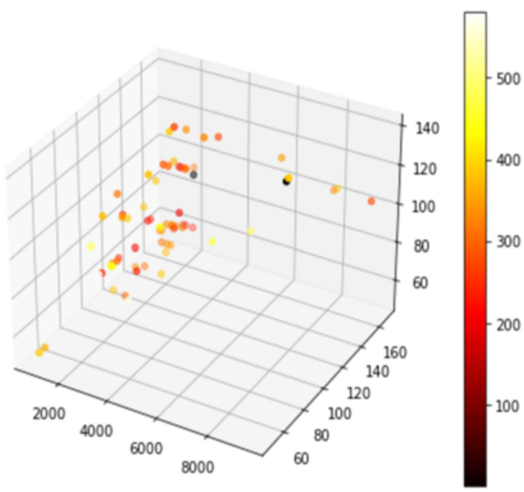

Figure 4. Distribution of our wind turbine data

4.2 The Optimal Value k of KNN

One of the most important step in KNN is to determine the optimal value of k to achieve the maximum accuracy of the model. The value k represents the number of clusters of the data or the number of individual values. Since we are facing a regression problem, in our case, the value of k is the number of nearest neighbors of the wind turbine target case. The Elbow method is very popular to determine this value of k.

We train our model with different values of k using both the training and testing data set. For each value of k, we keep track of the error rate. Algorithm 1 represents the Elbow classifier method to calculate the test mean squared error [28].

Algorithm1 Elbow.

```
function Elbow(k):
    #Initiating empy list
    test_mse : Array
    #Training model for every value of k
    for each value index in k:
        #Creating an instance of KNN
        model = KNN(n_neighbors = index)
        #Fitting our model
        #train_x and train_y are
        #the training and testing data set
        model.fit(train_x, train_y)
        #Predict over the train_set
        #and calculate the MSE
        tmp = model.predict(test_x)
        tmp = mse(tmp, test_y)
        test_mse.append(tmp)
    end for
    return test_mse
end function
```

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The figure 5 represents the graph of error rate vs K value.

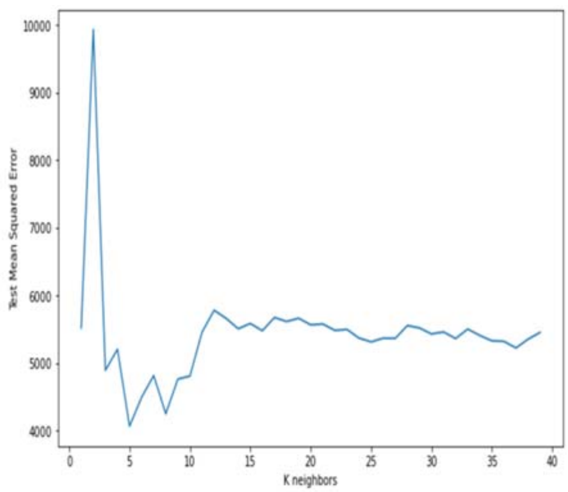

Figure 5. Error rate vs. K value

As we remark in the figure 5, the graph of the error rate increases after the value 5 of k neighbors. So, the optimal value for our model is 5.

4.3 Classification Using KNN

We retrain our model using the K nearest neighbors algorithm with the optimal k = 5 value. Algorithm 2 is used in the retrieval step of our CBR cycle, to calculate the distance between the target case and each case in the case base.

Algorithm2 EuclideanDistance.

```
function euclideanDistance(targetCase, dataCases, length):
  distance ← 0
   for each value x in range(length):
        distance ← square(targetCase[x] - dataCases[x]) + square(targetCase[x] - dataCases[x])
   end for
return sqrt(distance)
end function
```

The cases are then sorted on the basis of distance to extract the top five similar neighbors Algorithm3.

Algorithm3 KNN_Training.

```
function knn_train(trainingSet, dataCases, k):
distances : Dictionary
sorted_distance : Array
neighbors : Array
# get the size of the data cases
length ← dataCases.count
# Calculating euclidean distance between each row of training data and test data
for each value x in range(len(trainingSet)):
    dist ← euclideanDistance(dataCases, trainingSet.iloc[x], length)
    distances[x] ← dist[0]
end for
# Sorting cases on the basis of distance
sorted_distance = sort(distances)
# Extracting top k neighbors
for each value x in range(k):
    neighbors.append(sorted_distance[x])
end for
return neighbors
end function
```

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

We get the following result for the 5 nearest neighbors as shown in Figure 6. Each plotted curve is a function that captures the relationship between the wind speed and the wind power for all of the five nearest neighbors and which represent the most similar cases for our target case. The proposed hybrid approach provides a good power curve forecasting of a new installed wind turbine at different wind speed value.

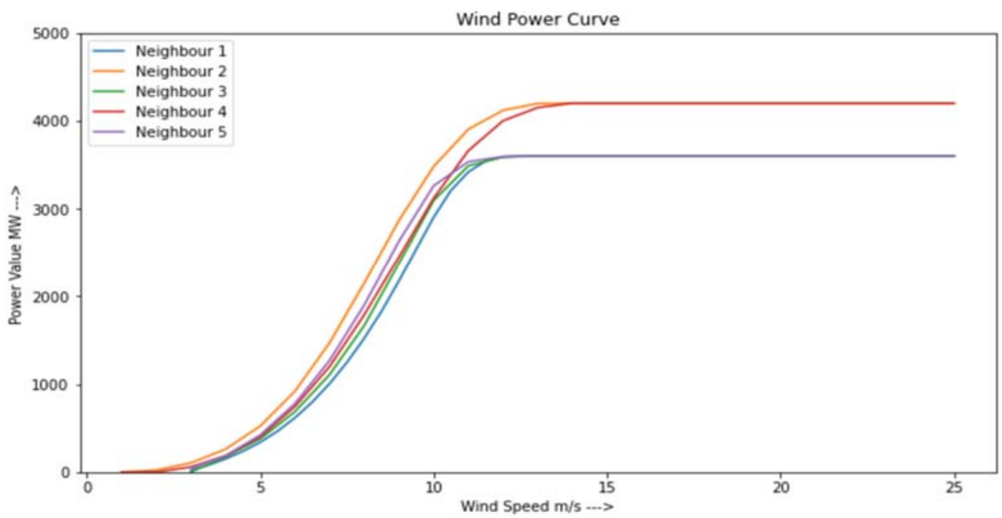

Figure 6. Wind Power Curve for the 5 nearest neighbors

From the results obtained using the KNN algorithm, we can present the power curve of our target case as a function that captures the relationship between the wind speed and the average of the power values of the five nearest neighbors as shown in Figure 7. The average of the power value and the power value of each corresponding neighbor can be correlated as:

$$Average_{Power}{}^{i} = 1/k \sum_{j=0}^{k} P_i{}^{j} \qquad (4)$$

where k is the number of nearest neighbors, Pi^j is the power value for a given nearest wind turbine (j) at a given wind speed (i).

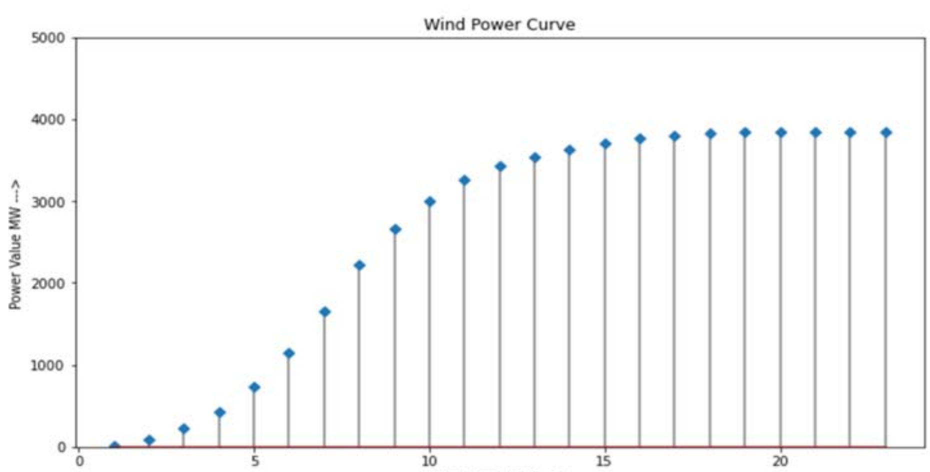

Figure 7. Proposed Wind Power Curve for the target case.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

5. CONCLUSION AND PERSPECTIVES

The wind turbine performance can be described by its WTPC which is a function that captures the relationship between wind speed and wind power. The wind turbine power curve plays an important role in wind power forecasting and therefore, the best selection of the wind farm installation. In this paper, a hybrid approach of wind turbines power curve modeling based on Case Based Reasoning approach and a machine learning algorithm has been proposed to obtain a new adapted and more accurate WTPC based on the WTPC of similar wind turbines. To enhance the search process, the KNN machine learning algorithm has been used in the retrieve step of the CBR cycle to search for wind turbines with similar characteristics, these wind turbines are then classified and sorted on the basis of distance of the target case. In the retain step, a new WTPC of the target case has been proposed based on the experiences of the similar cases. During this step, also, new experience and knowledge have been retained and stored into the case bases. The proposed WTPC for the target case will be used to solve future power curve modeling problem.

The main contribution of our work is the combination of the multi agent system, the case based reasoning approach and the KNN algorithm which allows to simulate the WTPC of new installed wind turbine and analyze its performance in real time based on past experiences of already installed ones. In this research, a real data base of 67 wind turbines has been used and divided into training and testing dataset to examine the performance of our hybrid approach. Our proposed model has succeeded in increasing the accuracy of the proposed WTPC and provided a good power curve forecasting of a new installed wind turbine at different wind speed value.

Our future work will be focused on testing our model on a large number of wind turbines using several datasets of different wind farms that will enrich our database cases, also to increase the number of features or wind turbines characteristics in order to get a better accuracy of our model. Finally, a multi agent system web based tool will be implemented based on our proposed approach for WTPC.

REFERENCES

[1] https://gwec.net/gwec-wind-power-industry-to-install-71-3-gw-in-2020-showing-resilience-during-covid-19-crisis/.

[2] Brown, C., 2012. Fast Verication of Wind Turbine Power Curves: Summary of Project Results. Master's thesis, Technical 487 University of Denmark, Denmark.

[3] LYDIA, M., SELVAKUMAR, A. Immanuel, KUMAR, S. Suresh, et al. Advanced algorithms for wind turbine power curve modeling. IEEE Transactions on sustainable energy, 2013, vol. 4, no 3, p. 827-835.

[4] AAMODT A. & PLAZA E. Case-Based Reasoning: Foundational Issues,Methodological Variations, and System Approaches. AI Communications. IOS Press, Vol. 7: 1, pp. 39-59, 1994.

[5] Bokolo ANTHONY JNR.: 'Hybrid Multi-Agents and Case Based Reasoning for Aiding Green Practice in Institutions of Higher Learning, Tehnički vjesnik' 26, 1(2019), 13-21.

[6] Schank R., Dynamic Memory (1982): 'A Theory of Reminding and Learning in Computersand People'. Cambridge University Press, New York.

[7] A. Zouhair, "Raisonnement à Partir de Cas Dynamique Multi-Agents: application à un système de tuteur intelligent»," PhD in computer science, in Cotutelle between the Faculty of Sciences and Technologies of Tangier (Morocco) and the University of Le Havre (France), 2014.

[8] E. M. E. Naimi and A. Zouhair, "Intelligent dynamic case-based reasoning using multiagents system in adaptive e-service, e-commerce and e-learning systems," Int. J. of Knowledge and Learning (IJKL), vol. 11, pp. 42-57, 2016.

[9] Y. A. Gerhana, et al., "The implementation of K-nearest neighbor algorithm in case-based reasoning model for forming automatic answer identity and searching answer similarity of algorithm case," Cyber and IT Service Management (CITSM), 2017 5th International Conference on, pp. 1-5, 2017.

[10] Z. Zhang, "Introduction to machine learning: k-nearest neighbors," Annals of translational medicine, vol. 4, 2016.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[11] Cost, Scott, and Steven Salzberg. "A weighted nearest neighbor algorithm for learning with symbolic features." Machine learning 10.1 (1993): 57-78.

[12] https://matplotlib.org/.

[13] https://www.python.org/

[14] YUAN, Chunhui et YANG, Haitao. Research on K-value selection method of K-means clustering algorithm. J—Multidisciplinary Scientific Journal, 2019, vol. 2, no 2, p. 226-235.

[15] REHMAN, Shafiq Ur, NADEEM, Aamer, and SINDHU, "Towards Aut mated Testing of Multi-Agent Systems Using Prometheus Design Models.", The International Arab Journal of Information Technology. 2016, vol. 16, no. 1.

[16] MOHAMED, Kouissi, EL MOKHTAR, En-Naimi, ABDELHAMID, Zouhair, et al. Computing platform architecture for management of renewable resources using multi-agents system based on the case based reasoning approach. Journal of Theoretical and Applied Information Technology, 2021, vol. 99, No 2.

[17] Vafadar, Shiva, and Ahmad Barfourosh, "Towards Intelligence Engineering in Agent-Based Systems." International Arab Journal of Information Technology (IAJIT) 12.1 (2015).

[18] Chaib-Draa, Brahim, Imed Jarras, and Bernard Moulin, "Systèmes multi-agents: principes généraux et applications.", Edition Hermès 242 (2001): 1030-1044.

[19] WANG, Meijia, LI, Qingshan, LIN, Yishuai, et al, "A Personalized Metasearch Engine Based on Multi-agent System (P)", The International Arab Journal of Information Technology, vol. 16, no. 6, In: SEKE. 2018. p. 707-706.

[20] Weyns, Danny, et al. "Environments for multi agent systems state of the art and research challenges.", International Workshop on Environments for Multi-Agent Systems. Springer, Berlin, Heidelberg, 20040.

[21] NIHAD, El Ghouch, EL MOKHTAR, En-Naimi, ABDELHAMID, Zouhair, et al. Hybrid approach of the fuzzy C-means and the K-nearest neighbors methods during the retrieve phase of dynamic case based reasoning for personalized follow-up of learners in real time. International Journal of Electrical and Computer Engineering, 2019, vol. 9, no 6, p. 4939.

[22] V. Sohoni, S. Gupta, R. Nema, A critical review on wind turbine power curve modelling techniques and their applications in wind based energy systems,J. Energy (2016) 1e18, 2016.

[23] VIRGOLINO, Gustavo CM, MATTOS, César LC, MAGALHÃES, José Augusto F., et al. Gaussian processes with logistic mean function for modeling wind turbine power curves. Renewable Energy, 2020, vol. 162, p. 458-465.

[24] JUBAIR, Mohammed Ahmed, MOSTAFA, Salama A., MUSTAPHA, Aida, et al. A survey of multi-agent systems and case-based reasoning integration. In : 2018 International Symposium on Agent, Multi-Agent Systems and Robotics (ISAMSR). IEEE, 2018. p. 1-6.

[25] OUYANG, Tinghui, KUSIAK, Andrew, et HE, Yusen. Modeling wind-turbine power curve: A data partitioning and mining approach. Renewable Energy, 2017, vol. 102, p. 1-8.

[26] PEI, Shenglei et LI, Yifen. Wind turbine power curve modeling with a hybrid machine learning technique. Applied Sciences, 2019, vol. 9, no 22, p. 4930.

[27] HOMEM, Thiago Pedro Donadon, SANTOS, Paulo Eduardo, COSTA, Anna Helena Reali, et al. Qualitative case-based reasoning and learning. Artificial Intelligence, 2020, vol. 283, p. 103258.

[28] AHUJA, Rishabh, SOLANKI, Arun, et NAYYAR, Anand. Movie recommender system using K-Means clustering and K-Nearest Neighbor. In : 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence). IEEE, 2019. p. 263-268.