The Early View of

Global Journal of Researches in Engineering
Electrical and Electronics Engineering

In case of any minor updation/modification/correction, kindly inform within

3 working days after you have received this.

Kindly note, the Research papers may be removed, added, or altered according to their final status.
Contents of the Issue

i. Copyright Notice
ii. Editorial Board Members
iii. Chief Author and Dean
iv. Contents of the Issue

1. Analysis of Weibull and Poisson Distribution use in Medium Voltage Circuit Breakers RUL Assessment. 1-6
2. Failure Prediction of Induction Motors: A Case Study Using CSLGH900/6-214, 5.8 MW, 11kV/3ph/50 Hz Sag Mill Motor at Goldfields, Damang Mine. 7-28
3. MONSDA: - A Novel Multi-Objective Non-Dominated Sorting Dragonfly Algorithm. 29-52
4. Research on Smooth Control of Wind Power Fluctuation with Hybrid Energy Storage. 53-58

v. Fellows
vi. Auxiliary Memberships
vii. Preferred Author Guidelines
viii. Index
Analysis of Weibull and Poisson Distribution use in Medium Voltage Circuit Breakers RUL Assessment

Dragan Stevanovic

Abstract- In this paper, Weibull and Poisson distribution calculation are carried out with new data to conclude a conclusion are they suitable for circuit breakers remaining useful life assessment (RUL). Old data are covering a 10 years period consisting of measured voltage drop on CB contacts and number of tripped short circuit faults. In this paper, new data, from the last 3 years, would be used to make a comparison with old data and make conclusions have been probability distributions correctly chosen.

Keywords: circuit breaker, weibull, poisson, remaining useful life, risk.

I. INTRODUCTION

Electrical companies nowadays are facing a lot of pressure considering equipment maintenance or replacement on the one hand and reducing operating expenses on another hand. Maintaining old equipment can be an expensive task, and that’s why power network operators should create a strategy of a most cost-effective method of equipment maintenance or replacement.

The same situation about equipment maintenance is happening at the Power Industry of Serbia. Among other equipment circuit breakers (CB) are a matter of concern, because most of CB’s that are currently in operation are installed during 70es and 80es (minimum oil CB’s), which means that they are at the end of their life, which is period characterized by increased number of faults and consequently increased maintenance.

Findings in this paper represent continuing of CB’s RUL assessment [1]. After gathering recent data, it is useful to investigate results from previous research and try to make new conclusions.

In previous research [1], using Weibull distribution we determine CB’s probability of failure by analyzing voltage drop values on its contacts, and using Poisson distribution the probability of failure if the number of short circuit trips exceeds limit value.

Both distributions were already used in literature and research for similar problems.

In [2], Weibull distribution was used for statistical analysis of age or wear out related CB faults.

In[3].Poisson distribution was used for modeling component faults in the power system with statistical data from maintenance and repairs.

In [4] presents analysis of fault types and their consequences, with cost structure and maintenance strategies. During wear out period fault intensity of high voltage (HV) CB’s follows Weibull distribution.

In [5], analysis of SF6 and minimum oil, CB’s faults were performed. Research includes totally 1546 CB’s from the Swedish and Finland power systems. Weibull distribution is assessing the RUL of CB’s components, which were the source of the fault.

In [6] they use the same distribution for reliability, RUL, and fault intensity assessment of HV SF6 CB’s.

In [7], few modified models of Weibull distribution were purposed for equipment reliability assessment in the power system. Least Squares method estimates parameters of Weibull distribution.

In [8] they use the same method for parameter estimation, where researchers are creating transformer lifetime model with Weibull distribution based on condition monitoring data.

II. WEIBULL DISTRIBUTION ASSESSMENT

Basic recommendations when choosing distribution are following [9]:

- Choose distribution, which researchers most frequently use in the same field of work.
- Choose distribution, which gives the most conservative results.
- Choose a simpler type of distribution. For example, if two-parameter distribution gives similar results like three-parameter distribution, then two-parameter distribution should be used.

Researchers deploy Weibull distribution very often when equipment aging and reliability has to be analyzed [10]. Weibull distribution can describe three types of equipment states (infant mortality, a period of normal work, wear out state) through the bathtub curve [11].

Weibull cumulative distribution function represents the probability of failure in a given period(1). In this case, two-parameter distribution was used, which consists of slop parameter (η) and shape parameter (β).
\[ F(t) = 1 - e^{-\left(\frac{t}{\eta}\right)^\beta} \] (1)

Slop parameter shows the time at which 63.2% of analyzed units fail. The shape parameter represents failure rate behavior and tells whether failures are decreasing or increasing. Shape parameter value has the following meaning:

- \( \beta < 1 \) indicates infant mortality,
- \( \beta = 1 \) period of normal work
- \( \beta > 1 \) wear-out failures. The higher value of beta indicates a greater rate of failure.

**a) Data analysis with Weibull distribution**

One power company (which is part of the Electrical Industry of Serbia) owns all CB’s which are part of analysis, and the same specialized work force is maintaining and monitoring their work years back. Totally 427 CB’s were part of the analysis, and their monitoring process started in 2007.

This research consists of two separate periods. Data for the first research are including period of 10 years (2007-2017), after that, next research includes new data from the last three years (2017-2020).

One of the main goals is to conclude whether new data follow Weibull distribution and is it justifiable to use it for this type of RUL assessment.

The calculation covers CB’s in 5 different categories (considering feeder type and rated voltage) and with two subcategories (1. Normal voltage drop value, 2. Permissible voltage drop value is by 25% larger [12]), making that way ten different categories in total. By this categorization, we can observe RUL more clearly, and come to the conclusion what makes the greatest influence on CB’s aging process.

Minitab 17 software and least square method [13] calculates Weibull distribution function with right-censored data (case when some devices didn’t fail during the period of analysis) for all CB’s categories.

For old and new data following values were calculated and compared: Weibull parameters and correlation coefficient. Table 1 shows all calculated values.

**Table 1: Correlation coefficient values with old and new data**

<table>
<thead>
<tr>
<th>Feeder type</th>
<th>Correlation coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>until 2017 yr.</td>
</tr>
<tr>
<td>Overhead +25%</td>
<td>0.985</td>
</tr>
<tr>
<td>Overhead</td>
<td>0.993</td>
</tr>
<tr>
<td>Underground +25%</td>
<td>0.976</td>
</tr>
<tr>
<td>Underground</td>
<td>0.965</td>
</tr>
<tr>
<td>10 kV feeders +25%</td>
<td>0.988</td>
</tr>
<tr>
<td>10 kV feeders</td>
<td>0.989</td>
</tr>
<tr>
<td>35 kV feeders +25%</td>
<td>0.972</td>
</tr>
<tr>
<td>35 kV feeders</td>
<td>0.984</td>
</tr>
<tr>
<td>All feeders +25%</td>
<td>0.989</td>
</tr>
<tr>
<td>All feeders</td>
<td>0.990</td>
</tr>
</tbody>
</table>

By observing the results of calculated correlation coefficient it is obvious that with an increased number of the data correlation coefficient is becoming greater, which means that data are becoming closer to Weibull distribution.

Next, Weibull parameters (scale parameter and shape parameter) were calculated with new data and compared with old ones (Table 2). By observing Weibull parameters from table 2, two conclusions could be made (taking into account the results from a previous paper [1]): underground feeders (both criteria of voltage drop value limit) have the highest \( \beta \) while overhead feeder has the lowest value. Considering \( \eta \) parameter, 10 kV feeders (+25% limit voltage drop level) have a longer time to failure, while 35kV feeders have the lowest \( \eta \) value (they will fail sooner than 10kV CB’s).

**Table 2: Weibull parameters with old and new data**

<table>
<thead>
<tr>
<th>Feeder type</th>
<th>( \eta )</th>
<th>( \beta )</th>
<th>Failed suspensions</th>
<th>( \eta )</th>
<th>( \beta )</th>
<th>Failed suspensions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overhead +25%</td>
<td>39.09</td>
<td>5.147</td>
<td>100/87</td>
<td>39.42</td>
<td>5.069</td>
<td>111/78</td>
</tr>
<tr>
<td>Overhead</td>
<td>37.08</td>
<td>4.797</td>
<td>131/56</td>
<td>37.42</td>
<td>4.935</td>
<td>141/48</td>
</tr>
<tr>
<td>Underground +25%</td>
<td>41.54</td>
<td>6.055</td>
<td>63/169</td>
<td>44.52</td>
<td>5.268</td>
<td>66/167</td>
</tr>
<tr>
<td>Underground</td>
<td>38.09</td>
<td>6.070</td>
<td>97/135</td>
<td>40.23</td>
<td>5.490</td>
<td>101/134</td>
</tr>
<tr>
<td>10 kV feeders +25%</td>
<td>43.44</td>
<td>5.627</td>
<td>87/224</td>
<td>45.50</td>
<td>5.100</td>
<td>97/215</td>
</tr>
<tr>
<td>10 kV feeders</td>
<td>40.39</td>
<td>5.071</td>
<td>135/176</td>
<td>42.00</td>
<td>4.918</td>
<td>142/172</td>
</tr>
<tr>
<td>35 kV feeders +25%</td>
<td>35.24</td>
<td>5.593</td>
<td>79/31</td>
<td>35.76</td>
<td>5.419</td>
<td>80/30</td>
</tr>
<tr>
<td>35 kV feeders</td>
<td>33.83</td>
<td>5.615</td>
<td>96/14</td>
<td>34.14</td>
<td>5.662</td>
<td>99/11</td>
</tr>
<tr>
<td>All feeders +25%</td>
<td>40.37</td>
<td>5.582</td>
<td>166/255</td>
<td>41.77</td>
<td>5.206</td>
<td>177/245</td>
</tr>
<tr>
<td>All feeders</td>
<td>37.98</td>
<td>5.281</td>
<td>231/190</td>
<td>39.16</td>
<td>5.134</td>
<td>242/182</td>
</tr>
</tbody>
</table>

In Table 2, values are showing expected aging phenomena. The number of failed CB’s is increasing, but on the other hand, with a greater number of data, a new insight could be perceived. Scale parameter (\( \eta \)) is, in most cases, slightly increased, which suggests that RUL is not as we were expecting with old data and that CB’s survival time is slightly greater compared with previous research.
III. Poisson Distribution Assessment

Poisson distribution [14, 15] is the discrete distribution used for modeling a number of events which are appearing in a specific period. Poisson distribution for calculation probability for a known number of past events ($k$) in the time interval ($t$) is [16]:

$$P(k) = \frac{(\lambda t)^k e^{-\lambda t}}{k!}$$  \hspace{1cm} (2)

Where is:
- $k$ – the number of faults in period ($t$)
- $\lambda$ – fault intensity
- $t$ – a time interval
- $P(r)$ – the probability of appearing $r$ number of faults in period $t$

Cases of Poisson distribution use[17]:
1. Researcher can present an event with the whole number
2. The occurrence of an event doesn’t depend on any other event
3. Mean value of event occurrence in a specific period is known
4. Number of events is countable

In the power system, Poisson distribution can predict faults such as short circuit faults. The number of those faults depends on feeder type (underground or overhead) and also by the area configuration where power network is situated (residential area, forest). Another influencing factor is weather condition and power network quality.

Procedure for Chi-Square Goodness of Fit Test

One of the methods for determining if data follow Poisson distribution is the Chi-squared test ($\chi^2$ test). This method represents a test of statistical hypothesis and is used to determine a significant difference between expected and observed intensity [18]. The Chi-squared test can test hypothesis do analyze data follow a certain distribution. It can also test Poisson distribution[19]. The calculation is carried out in the following way [20, 21]:

$$\chi^2 = \sum \frac{(O-E)^2}{E}$$  \hspace{1cm} (3)

Where is:
- $\chi^2$–Chi-squared value
- $O$ – observed value
- $E$ – expected value

1. Hypothesis formulation:
   a. Null hypothesis: $H_0$: data $X$ is following Poisson distribution $X \sim \text{Poisson}$
   b. Alternative hypothesis: $H_1$: data $X$ doesn’t follow a Poisson distribution

2. Calculation of Chi-squared test:
   Table 3 presents a number of short circuit trips on one 10kV feeder.

Table 3: Number of observed short-circuit trips on 10kV feeder

<table>
<thead>
<tr>
<th>Year</th>
<th>Number of trips</th>
</tr>
</thead>
<tbody>
<tr>
<td>2013</td>
<td>0</td>
</tr>
<tr>
<td>2014</td>
<td>0</td>
</tr>
<tr>
<td>2015</td>
<td>0</td>
</tr>
<tr>
<td>2016</td>
<td>0</td>
</tr>
<tr>
<td>2017</td>
<td>0</td>
</tr>
<tr>
<td>2018</td>
<td>1</td>
</tr>
<tr>
<td>2019</td>
<td>1</td>
</tr>
</tbody>
</table>

Using values from table 3, we calculate each fault intensity (Table 4).
Table 4: Faults intensity in a period of 7 years

<table>
<thead>
<tr>
<th>Number of faults (k)</th>
<th>Intensity (n)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>&gt;10</td>
<td>0</td>
</tr>
</tbody>
</table>

K represents a number of faults during n=7 years.

The mean of the Poisson distribution is:

$$\lambda = \frac{(0 \cdot 5 + 1 \cdot 2 + 2 \cdot 0 + 3 \cdot 0 + 4 \cdot 0 + 5 \cdot 0 + 6 \cdot 0 + 7 \cdot 0 + 8 \cdot 0 + 9 \cdot 0 + 10 \cdot 0)}{7}$$

$$\lambda = 0.2857$$  

(4)

Example (5) and 6) are presenting expected fault intensity calculation, and the table 5 presents values of that calculation.

$$p_0 = P(K = 0) = e^{-0.2857}(0.2857)^0 = 0.7515$$  

(5)

$$E_0 = 0.7515 \cdot 7 = 5.26$$  

(6)

Table 5: Expected number of faults

<table>
<thead>
<tr>
<th>Number of faults (k)</th>
<th>Poisson</th>
<th>Observed</th>
<th>Expected (E₀)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.7515</td>
<td>5</td>
<td>5.2603</td>
</tr>
<tr>
<td>1</td>
<td>0.2147</td>
<td>2</td>
<td>1.5030</td>
</tr>
<tr>
<td>2</td>
<td>0.0307</td>
<td>0</td>
<td>0.2147</td>
</tr>
<tr>
<td>3</td>
<td>0.0029</td>
<td>0</td>
<td>0.0204</td>
</tr>
<tr>
<td>4</td>
<td>0.0002</td>
<td>0</td>
<td>0.0015</td>
</tr>
<tr>
<td>5</td>
<td>0.0000</td>
<td>0</td>
<td>0.0001</td>
</tr>
<tr>
<td>6</td>
<td>0.0000</td>
<td>0</td>
<td>0.0000</td>
</tr>
<tr>
<td>7</td>
<td>0.0000</td>
<td>0</td>
<td>0.0000</td>
</tr>
<tr>
<td>8</td>
<td>0.0000</td>
<td>0</td>
<td>0.0000</td>
</tr>
<tr>
<td>9</td>
<td>0.0000</td>
<td>0</td>
<td>0.0000</td>
</tr>
<tr>
<td>&gt;10</td>
<td>0.0000</td>
<td>0</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

Calculation of Chi-squared value:

$$\chi^2 = \sum \frac{(O - E)^2}{E} = \frac{(5 - 5.2603)^2}{5.2603} + \frac{(2 - 1.5030)^2}{1.5030} + \frac{(0 - 0.2147)^2}{0.2147} + \frac{(0 - 0.0015)^2}{0.0015} ...$$

$$\chi^2 = 0.4139$$  

(7)

Degrees of freedom are k−g−1. In this case number of classes is k = 11 (number of faults intensity), and from data we estimate one parameter g = 1 (in this case one parameter, λ). In the end, degrees of freedom are equal to 11−1−1=9.

Value of significance level is selected to be 0.05. That value means there is a 5% probability that the observed relationship between variables exists by coincidence [22]; in other words, data doesn’t follow assumed distribution [23].
Table 6: Table of critical values

<table>
<thead>
<tr>
<th>d.f.</th>
<th>.995</th>
<th>.99</th>
<th>.975</th>
<th>.95</th>
<th>.9</th>
<th>.05</th>
<th>.025</th>
<th>.01</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.02</td>
<td>2.71</td>
<td>3.84</td>
<td>5.02</td>
</tr>
<tr>
<td>2</td>
<td>0.01</td>
<td>0.02</td>
<td>0.05</td>
<td>0.10</td>
<td>0.21</td>
<td>4.61</td>
<td>5.99</td>
<td>7.38</td>
</tr>
<tr>
<td>3</td>
<td>0.07</td>
<td>0.11</td>
<td>0.22</td>
<td>0.35</td>
<td>0.58</td>
<td>6.25</td>
<td>7.81</td>
<td>9.35</td>
</tr>
<tr>
<td>4</td>
<td>0.21</td>
<td>0.30</td>
<td>0.48</td>
<td>0.71</td>
<td>1.06</td>
<td>7.78</td>
<td>9.49</td>
<td>11.14</td>
</tr>
<tr>
<td>5</td>
<td>0.41</td>
<td>0.55</td>
<td>0.83</td>
<td>1.15</td>
<td>1.61</td>
<td>9.24</td>
<td>11.07</td>
<td>12.83</td>
</tr>
<tr>
<td>6</td>
<td>0.68</td>
<td>0.87</td>
<td>1.24</td>
<td>1.64</td>
<td>2.20</td>
<td>10.64</td>
<td>12.59</td>
<td>14.45</td>
</tr>
<tr>
<td>7</td>
<td>0.99</td>
<td>1.24</td>
<td>1.69</td>
<td>2.17</td>
<td>2.83</td>
<td>12.02</td>
<td>14.07</td>
<td>16.01</td>
</tr>
<tr>
<td>8</td>
<td>1.34</td>
<td>1.65</td>
<td>2.18</td>
<td>2.73</td>
<td>3.49</td>
<td>13.36</td>
<td>15.51</td>
<td>17.53</td>
</tr>
<tr>
<td>9</td>
<td>1.73</td>
<td>2.09</td>
<td>2.70</td>
<td>3.33</td>
<td>4.17</td>
<td>14.68</td>
<td>16.92</td>
<td>19.02</td>
</tr>
<tr>
<td>10</td>
<td>2.16</td>
<td>2.56</td>
<td>3.25</td>
<td>3.94</td>
<td>4.87</td>
<td>15.99</td>
<td>18.31</td>
<td>20.48</td>
</tr>
</tbody>
</table>

Table 6: Table of critical values

Chi-square Distribution Table

<table>
<thead>
<tr>
<th>d.f.</th>
<th>.995</th>
<th>.99</th>
<th>.975</th>
<th>.95</th>
<th>.9</th>
<th>.05</th>
<th>.025</th>
<th>.01</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.02</td>
<td>2.71</td>
<td>3.84</td>
<td>5.02</td>
</tr>
<tr>
<td>2</td>
<td>0.01</td>
<td>0.02</td>
<td>0.05</td>
<td>0.10</td>
<td>0.21</td>
<td>4.61</td>
<td>5.99</td>
<td>7.38</td>
</tr>
<tr>
<td>3</td>
<td>0.07</td>
<td>0.11</td>
<td>0.22</td>
<td>0.35</td>
<td>0.58</td>
<td>6.25</td>
<td>7.81</td>
<td>9.35</td>
</tr>
<tr>
<td>4</td>
<td>0.21</td>
<td>0.30</td>
<td>0.48</td>
<td>0.71</td>
<td>1.06</td>
<td>7.78</td>
<td>9.49</td>
<td>11.14</td>
</tr>
<tr>
<td>5</td>
<td>0.41</td>
<td>0.55</td>
<td>0.83</td>
<td>1.15</td>
<td>1.61</td>
<td>9.24</td>
<td>11.07</td>
<td>12.83</td>
</tr>
<tr>
<td>6</td>
<td>0.68</td>
<td>0.87</td>
<td>1.24</td>
<td>1.64</td>
<td>2.20</td>
<td>10.64</td>
<td>12.59</td>
<td>14.45</td>
</tr>
<tr>
<td>7</td>
<td>0.99</td>
<td>1.24</td>
<td>1.69</td>
<td>2.17</td>
<td>2.83</td>
<td>12.02</td>
<td>14.07</td>
<td>16.01</td>
</tr>
<tr>
<td>8</td>
<td>1.34</td>
<td>1.65</td>
<td>2.18</td>
<td>2.73</td>
<td>3.49</td>
<td>13.36</td>
<td>15.51</td>
<td>17.53</td>
</tr>
<tr>
<td>9</td>
<td>1.73</td>
<td>2.09</td>
<td>2.70</td>
<td>3.33</td>
<td>4.17</td>
<td>14.68</td>
<td>16.92</td>
<td>19.02</td>
</tr>
<tr>
<td>10</td>
<td>2.16</td>
<td>2.56</td>
<td>3.25</td>
<td>3.94</td>
<td>4.87</td>
<td>15.99</td>
<td>18.31</td>
<td>20.48</td>
</tr>
</tbody>
</table>

In Table 6 for the number of degrees of freedom and significance level of 0.05, the critical value is 16.92. This value shows at which $\chi^2$ value H0 hypothesis is acceptable. [24]

If $\chi^2 < 16.92$ (illustrated in figure 2) than the H0 hypothesis is acceptable, which means there is no evidence that the data doesn’t follow Poisson distribution.

Figure 2: Graphical representation of Chi-squared test

After analyzing faults on all CB’s, results are presented in Table 7.

Table 7: Results of Chi-square goodness of fit test

<table>
<thead>
<tr>
<th>Number of CB’s where H0 hypothesis is accepted (data follow Poisson distribution)</th>
<th>Number of CB’s where H0 hypothesis is rejected</th>
</tr>
</thead>
<tbody>
<tr>
<td>148</td>
<td>19</td>
</tr>
<tr>
<td>89 %</td>
<td>11 %</td>
</tr>
</tbody>
</table>

Results are showing that in most cases (89%), data are following Poisson distribution. Analysis of short circuit faults will continue in the future periods to determine will the bigger amount of data increase fit to Poisson distribution.

IV. Conclusion

In this paper, new data are used to check the correctness of methods used in previous research. New data are showing that with an increased number of samples, fit to Weibull distribution becomes greater. Better fit to the Weibull distribution becomes obvious by observing the values of the correlation coefficient of Weibull distribution with old and new data. Shape and scale parameters are showing that survival time is different from previous assessment and that CB’s RUL is a little bit greater than in previous research. Chi-squared Goodness of fit shows that almost 90% of current data of short circuit faults are following Poisson distribution. With Poisson distribution, CB’s probability of failure in the next period could be very easily assessed.

This paper proves that it is justifiable to use Weibull and Poisson distribution for CB’s remaining useful life estimation. With these two methods, CB’s RUL could be calculated very fast and easy which could be later used for other studies such as risk assessment, power station reliability assessment, determining critical points in the power system, or justification of CB replacement.

Research in this field will be continued by gathering data from other power operators in the Power Industry of Serbia to better understand the problem of the CB aging process by using voltage drop values and short circuit faults.

References Références Referencias


16. A. Ünsal, B. Mumyakmaz, N. S. Tunaboylu „Predicting the failures of transformers in a power system using the poisson distribution: a case study”, ELECO 2005 International Conference on Electrical and Electronics Engineering, Bursa, 2005


Failure Prediction of Induction Motors: A Case Study using CSLGH900/6-214, 5.8 MW, 11 kV/3ph/50 Hz Sag Mill Motor at Goldfields, Damang Mine

C. K. Amuzuvi & H. Warden

Abstract - This paper proposes a generalised feed-forward artificial neural network model that fulfils the failure prediction of a three phase 5.8MW, 11 kV Slip-Ring SAG Mill Induction Motor at Goldfields Ghana Limited, Damang Mine. It provides a general understanding of three phase induction motors, faults associated with induction motors and also emphasizes the use of intelligent systems, particularly artificial neural network, a modern failure prediction technology of induction motors. Site analysis and motor data (Current, Power and Winding Temperatures) collection were conducted at the Damang Mine. Simulation results are presented using MATLAB software (2017a) package to develop the fault prediction model. The proposed feed-forward neural network used the Levenberg-Marquardt and Bayesian Regularisation in training. The use of Log sigmoid and Tan sigmoid was also employed as the activation functions of the hidden layer, with hidden layer size kept at 10. Simulation and calculations are done in real time on load measurement from the SAG Mill motor. Analysis of the model's output performance were conducted by correlation of coefficient of network performance, R and Mean Squared Error, MSE. The proposed implemented model resulted in an increase in the SAG Mill motor availability, an improved reliability and a great impact on safety of employees and equipment. It is therefore, worthwhile, to invest in the deployment of this model to augment the condition monitoring needs of the SAG Mill motor and other such equipment in the plant.

Keywords: SAG mill induction motor, feed-forward neural network, multilayer perceptron.

I. Introduction

Damang Gold Mine, a subsidiary of Goldfields International is a world class mining operation consisting of a 25 MTPA open pit mining and a 5.2 MTPA Carbon in Leach (CIL) metallurgical plant. Located in the south western part of Ghana, 300 km by road from the capital of Ghana, Accra, the mine exploits oxide and fresh hydrothermal mineralization in addition to Witwatersrand – style transitional paleo placer gold. The plant is designed to treat 5.2 MTPA of gold ore from a blend of approximately 20% oxide ore and 80% fresh ore sourced from various open pit mining operations.

Process feed for the 12-month period of 2016 comprised 4.3 Mt at a yield of 1.17 g/t for a 148 koz of gold.

The plant has 2×5.8 MW ball mill and sag mill, a 1×600 kW primary gyratory crusher, 1×375 kW pebble crusher, 8×CIL tanks and a secondary crushing plant with a maximum electric power draw of 17.5 MW at peak times. The mine uses a lot if induction motors at the crushing circuit, milling circuit, CIL circuit, elution circuit, tailings, etc., because of its strength, mechanical simplicity and adaptability to a variety of applications [1]. The plant is often faced with issues associated with burnt induction motors. Unfortunately, the exact causes are not clearly known.

Induction motors are the mainstay for every industry. They are widely used in transportation, mining, petrochemical, manufacturing and in almost every other field using electrical power. These motors are simple, efficient, highly robust and rugged, thus, offering a very high degree of reliability. However, like any other machine, they are susceptible to faults, which if left unmonitored, might lead to catastrophic failure of the machine in the long run especially due to heavy duty cycles, poor working environment alongside with installation and manufacturing factors.

In a bid to detect fault and avoid complete breakdown of induction motors with its concomitant production losses, on-line condition monitoring of the induction motors must be implemented for effective operation of these machines. With increasing demands for reliability and efficiency, fault prediction in induction motors has become necessary, particularly in industries that make use of these rotary equipment of which the Damang Gold Mine is no exception [2]. Various fault conditions of induction motors as well as methods of their detection and prediction are presented in this paper.

a) Some Impacts of the Occurrence of Faults

With the mines current maintenance cost of electrical motors on the high, the mine must come up with strategies to bring the overall cost of engineering maintenance down. Figure 1(a) is a graph showing annual motor change-out from 2012 to 2016 and Figure
1(b) is the probability of occurrence of faults in an operating induction motor [2]. Research has shown that, failures associated with induction motors are often caused by rotor, stator, and bearing failures, etc. [2].

With the current price of gold on the downside, the maintenance department is under intense pressure to efficiently maintain the plant machinery to continue to stay in business. Table 1 show gold prices from 2012 to 2016 respectively. This research work seeks to identify and assess in detail, all the various root causes of induction motor failures in the mine and suggest a means of accurately predicting future failures.

![Annual Motor Change Out](image)

**Figure 1:** (a) Graph Showing Number of Annual Motor Change – Out; and (b) Probability of Occurrence of Faults

**Table 1:** Gold Price from 2012 to 2016 Year

<table>
<thead>
<tr>
<th>Year</th>
<th>Closing Price</th>
<th>Year Open</th>
<th>Year High</th>
<th>Year Low</th>
<th>Year Close</th>
<th>% Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012</td>
<td>$1,668.86</td>
<td>$1,590.00</td>
<td>$1,790.00</td>
<td>$1,537.50</td>
<td>$1,664.00</td>
<td>5.68%</td>
</tr>
<tr>
<td>2013</td>
<td>$1,409.51</td>
<td>$1,681.50</td>
<td>$1,692.50</td>
<td>$1,192.75</td>
<td>$1,201.50</td>
<td>−27.79%</td>
</tr>
<tr>
<td>2014</td>
<td>$1,266.06</td>
<td>$1,219.75</td>
<td>$1,379.00</td>
<td>$1,144.50</td>
<td>$1,199.25</td>
<td>−0.19%</td>
</tr>
<tr>
<td>2015</td>
<td>$1,158.86</td>
<td>$1,184.25</td>
<td>$1,298.00</td>
<td>$1,049.60</td>
<td>$1,060.20</td>
<td>−11.59%</td>
</tr>
<tr>
<td>2016</td>
<td>$1,251.92</td>
<td>$1,075.20</td>
<td>$1,372.60</td>
<td>$1,073.60</td>
<td>$1,151.70</td>
<td>8.63%</td>
</tr>
</tbody>
</table>

(Source: [3])
b) **Induction Motor**

An induction motor is a type of asynchronous alternating current (AC) motor, where power is supplied to the rotating device (rotor) by means of electromagnetic induction. There are two types, namely wound or slip-ring induction motor and squirrel-cage induction motor.

Squirrel-cage induction motors are the preferred choice for industries due to their low cost, high reliability, absence of slip-rings and brushes, which eliminate the risk of sparking thereby, making them explosion proof with high efficiency over a wide range of power outputs.

They also have the ability of speed control. From a constant frequency source, they operate as constant speed drives. For continuous speed control over a wide speed range, a solid-state variable-frequency converter provides an indirect source of supply [4].

c) **Induction Motor Failure**

Induction motors are rugged, low cost, low maintenance, reasonably small sized, reasonably highly efficient and operating with an easily available power supply. They are reliable in operations but are subject to different types of undesirable faults.

**Figure 2:** (a) Block Diagram Presentation of Internal Faults; and (b) Block Diagram Representation of External Faults

Sources of induction motor faults may be internal or external. In Figure 2(a) and Figure 2(b) [2], block diagrams of internal and external faults are depicted. The most vulnerable parts for fault in the induction motor are bearing, stator winding, rotor bar, and shaft. Besides, due to non-uniformity of the air gap between stator-inner surface and rotor-outer surface, motor faults occur [5]. Faults in induction motors can be categorized as:
1. Electrical-related faults due to unbalance supply voltage or current, single phasing, under or over voltage or current, reverse phase sequence, earth fault, overload, inter-turn short-circuit fault, and crawling;

2. Mechanical-related faults due to broken rotor bar, mass unbalance, air gap eccentricity, bearing damage, rotor winding failure and stator winding failure; and

3. Environmental-related faults such as ambient temperature, external moisture as well as vibrations of machine due to reasons like installation defect and foundation defect affect the performance of induction motor.

Figure 3(a) [5] show the rotor and parts of a broken rotor bar and Figure 3(b) a rotor with mass unbalance fault, with a hole drilled into one bar.

Industrial processes make use of a large number of asynchronous motors even in sensitive applications. Consequently, a defect can induce high losses in terms of cost and can be dangerous in terms of security and safety. Motor failures are mostly directly or indirectly caused by insulation breakdown, bearing wear or extensive heating of different motor parts involved in motor operation [6]. Multiple faults may occur simultaneously in an induction motor, which may result in unbalanced stator currents and voltages, oscillations in torque, reduction in efficiency and torque, overheating and excessive vibration. Normally, electric motors do not fail suddenly. It happens over time and regular inspection will detect a problem before a serious situation develops. Three main components of electric motors that experience faults are the stator, rotor and bearings. These faults may be a growing one with only small effects on the operation, a partial non-catastrophic one with emergency operation possible or a catastrophic one with total drive breakdown [6]. Incipient fault detection is preferably done to find faults before complete motor failure in order to avoid service downtime and large losses.

**d) Condition Monitoring and Its Necessity**

Induction motors are the main workhorse of industrial prime movers due to their ruggedness, low cost, low maintenance, reasonably small size, reasonably high efficiency, and operating with an easily available power supply. About 50% of the total generated power of a nation is consumed by these induction motors[5]. These statistics gives an idea regarding the use of huge number of induction motors, but they have some limitations in their operating conditions. If these conditions are exceeded, then some premature failure may occur in the stator or/and rotor. This failure, in many applications in industry, may lead to a shut down, even, the entire industrial process resulting in loss of production time and money. It is, therefore, an important issue to avoid any kind of failure of an induction motor. Operators and technicians of induction motors are under continual pressure to prevent unscheduled downtime and also to reduce maintenance cost of motors.

Maintenance of electrical motors can be done in three forms: breakdown maintenance, fixed-time maintenance, and condition-based maintenance. In breakdown maintenance, the strategy is ‘run the motor until it fails’ which means maintenance action is taken only when the motor gets breakdown. In this case, though the motor may run comparatively for a long time before the maintenance is done, when breakdown occurs, it is necessary to replace the entire machine, which is much costlier compared to replacing or repairing the faulty parts of the motor. Also, it causes loss of productivity due to downtime.

In fixed-time maintenance, the motor is required to stop for inspection, which causes long downtime. Also, trained and experienced technical persons are required to recognise each and every fault correctly. All these necessitate the condition-based maintenance of the motor. In this form of maintenance, the motor is allowed to run normally and action is taken at the very first sign of an incipient fault.

In condition monitoring, when a fault has been identified, sufficient data is required for the plant operator for the best possible decision making on the...
correct course of action. If the data is insufficient, there remains the chance for wrong diagnosis of fault, which leads to inappropriate replacement of components, and if the root of the problem is not identified properly, the replacement or any other action taken already will succumb to the same fate. In condition monitoring, signals from the concerned motor are continuously fed to the data acquisition system and the health of the motor is continuously evaluated during its operation for which it is also referred as online condition monitoring of the motor, and hence, it is possible to identify the faults even while they are developing. The operator/technician can take preparation for the preventive maintenance and can arrange for necessary spare parts in advance, for repairing. Thus, condition monitoring can optimise maintenance schedule and minimise motors downtime and thereby increase the reliability of the motor. Advantages of using condition monitoring can be mentioned point wise as follows:

1. Prediction of motor failure;
2. Optimisation of the maintenance schedule of the motor;
3. Reduction of maintenance cost;
4. Reduction of the downtime of the machine; and
5. Improvement of the reliability of the motor.

Condition monitoring and fault detection are usually carried out by investigating the corresponding anomalies in the machine current, voltage and leakage flux. Other methods include monitoring the core temperature, bearing vibration level and pyrolysated products. Fault conditions such as insulation defects and bearing degradation may also be diagnosed [2].

e) Failure Prediction Methods or Techniques
According to [7], online failure prediction aims to identify situations that will evolve into a failure. Classification of failure prediction methods are usually based on the type of input data used, namely: data from failure tracking, symptom monitoring, detected error reporting and undetected error auditing. System monitoring, however, is mostly used as it is effective and offers reliable data based on analysis of time series and/or type of symptoms. In order to build high availability systems based on failure prediction, methods are developed not only to capture, select, or interpret essential data and predict future system states but also to provide proactive recovery and failure avoidance schemes, which build on these predictions and help to self-manage the system.

Thus, it has become necessary to diagnose motor faults for effective maintenance plans by management, so as to avoid complete failure of systems or machines in the future. Using baseline characteristics of a healthy motor as a reference data, any deviation in motor operating characteristics obtained from system monitoring may be used to perform fault detection and diagnosis, irrespective of unavoidable manufacturing defects in the system. Depending on the region of fault occurrence, five main categories of faults, namely: stator faults, eccentricity faults, rotor faults, bearing faults and vibration faults are diagnosed based on various failure prediction methods [2].

i. Vibration Spectrum Analysis
This technique is used to detect bearing faults. High frequency components of vibration are created due to friction or forces occurring in the rolling element bearing in electrical machines under normal conditions. In case of a defect in the bearings or breaks in the lubrication layer between the friction surfaces, shock pulses are produced.

The method analyses the vibration spectrum of an induction machine using piezoelectric accelerometer, which works on Fast Fourier Transform to extract from a time domain signal, the frequency domain representation. In diagnosing bearing fault, the harmonic vibration spectrum of the healthy motor and that with defective bearing is analysed individually. Upon comparison, it is realized that the vibration amplitude for faulty motor is larger than that of a healthy motor. Dynamic simulation of motor running with bearing fault to analyse frequency spectrum of electromagnetic torque produced by the faulty motor may provide similar result when compared with its vibration spectrum.

ii. Park Vector Approach and Complex Wavelets
Park vector transformation approach is used to diagnose stator faults on a three-phase induction motor due to the impact of fault on the machine current. This technique uses Park’s Transform to derive a two-dimensional Park’s current vector components, which are expressed as functions of the phase currents of the three-phase induction motor. Thus, the locus of instantaneous spatial vector sum of the measured three phase stator currents forms the basis for Park’s vector.

This maps a circle, which has its centre at the origin of the coordinates. This locus is distorted by stator winding faults and thus provides easy fault diagnosis. In other words, a graphical representation of the Park’s current vector for a faulty motor gives an elliptical shape, which is a distortion of the circularly shaped Park’s current vector representation of a healthy motor. The amount of distortion of the circular shape depends on the level of stator fault of the motor. Simulation and experimental results are finally analysed using complex wavelets.

iii. Motor Current Signature Analysis (MCSA)
This technique can be used to detect rotor faults and eccentricity. In case of a fault, current harmonics in the stator current, caused by a backward rotating field in the air gap, are analysed by MCSA. This requires only one current sensor, whose function is based on signal processing techniques like the Fast Fourier Transform (FFT).
An equipment set-up, which comprises current transformer, signal conditioning unit, data collector/analysyer and computer, is used for measuring the motor current. Data is acquired by performing FFT on the stator current. The data obtained, is analysed after FFT is normalized as a function of the first harmonic amplitude. Conversely, harmonic contents or percentage amplitude for harmonics, increase with increase in the level of faults, like the number of broken rotor bars and eccentricity.

iv. Intelligent Techniques

Several intelligent techniques like Fuzzy logic systems, Artificial Neural Networks and Neuro-Fuzzy Systems usually have three prime steps for induction motor condition monitoring. These are: i) Signature extraction; ii) Fault detection; and iii) Fault severity estimation.

Apart from the above-mentioned techniques, some other methods for incipient fault detection of induction motors are the finite element method, vibration testing and analysis, Concordia transform, external magnetic field analysis, multiple reference frames theory, power decomposition technique, zero crossing method and modal analysis method. This work, however, makes use of the artificial neural network for failure prediction of induction motors.

f) Artificial Neural Network

According to [8], Artificial Neural Network (ANN) is a non-linear mapping structure inspired by observed process in natural network of neurons in the human brain. It consists of highly interconnected simple computational units called neurons. It imitates the learning process of the human brain and can process problems, which involve complex, non-linear, imprecise and noisy data. It is ideally suited for modelling and predicting the outcome of new independent input data after training.

ANNs are parallel computational models consisting of densely interconnected adaptive processing units. They are used for a wide variety of applications where statistical methods are traditionally employed. ANN is therefore being recognised as a powerful tool for data analysis. By their adaptive nature, “learning by example” replaces “programming” in solving problems. This feature makes such computational models very appealing especially in application domains, where a problem to be solved is not understood fully but training data is readily available. Back propagation algorithm is the most widely used learning algorithm in an ANN. Various types of ANN include Multilayered Perceptron, Radial Basis Function and Kohonen networks. In fact, majority of the networks are more closely related to traditional mathematical and/or statistical models, such as non-parametric pattern classifiers, clustering algorithms, non-linear filters, and statistical regression models than they are to neurobiology models.

ANNs are constructed with layers of units. All units in a particular layer perform similar tasks. The first and last layers of a multilayer ANN consist of input units (independent variables) and output units (dependent or response variables) respectively. All other units (hidden units) make up the hidden layer. The behaviour of a unit is governed by an input function and an output or activation function. These functions are normally the same for all units within the whole ANN. Input into a node is a weighted sum of outputs from nodes connected to it. There exists a threshold term, which is a baseline input to a node in the absence of any other inputs. A weight is termed inhibitory if it is negative as it decreases net input, otherwise it is called excitatory.

Each unit takes its net input and applies an activation function to it. In instances where the inputs and outputs are binary encoded, the threshold function becomes very useful. The activation function mainly maps the outlying values of the obtained neural input back to a bounded interval. The activation function shows a great variety. However, the most common choice is the sigmoid function since it maps a wide domain of values into the interval.

i. Development of an ANN Model

A neural network forecasting model is developed by the following steps:

1. Variable selection;
2. Formation of training, testing and validation sets;
3. Neural network architecture; and
4. Model building.

Suitable variable selection procedures are used to select the input variables, important for modelling or forecasting variable(s) under study in the first step. This is followed by the formation of three distinct data sets called training, testing and validation sets. These data sets are used by the neural network not only to learn current data patterns (training set) and evaluate the overall ability of the supposedly trained network (testing set), but also to check the performance of the trained network using the validation set. The third step defines the network structure, which includes a number of hidden layers and hidden nodes as well as the number of output nodes and the activation function. The next step involves model building.

The model of a very popular and frequently used multilayer feed-forward neural network (FFNN) or multilayer perceptron (MLP) learned by back propagation algorithm is constructed based on supervised procedure or on examples of data with known output. The examples presented are assumed to implicitly contain the information necessary to establish the relation for building the model. An MLP allows prediction of an output object for a given input object. Its non-linear elements or neurons are arranged in
successive layers with a unidirectional flow of information from input layer to output layer through hidden layer(s). With adequate data, only one hidden layer is always sufficient for an MLP as it can learn to approximate virtually any function to any degree of accuracy. MLPs are therefore also known as universal approximates. Generally, learning methods in neural networks are classified into three basic types, namely; supervised learning, unsupervised learning or reinforced learning. A neural network learns off-line if the learning phase and the operation phases are distinct. On-line learning occurs when it learns and operates at the same time. Supervised learning is usually performed off-line based on training data, whereas unsupervised learning is performed on-line based on given data. In reinforced learning, data is usually not given, but generated by interactions with the environment.

ii. Architecture of Neural Networks

The two most widely used ANN architecture are the feed-forward networks and the feedback or recurrent networks. Other types of ANN architecture include stochastic network, physical network, bi-directional network, Elman and Jordan network, Hopfield network, self-organising map and long short-term memory networks. Feed-forward networks have no feedback loops and are extensively used in pattern recognition. Thus, signals are allowed to travel one way only; from input to output. The output layer does not affect that same layer. In feedback networks however, signals do not travel in one way only due to the presence of a feedback loop. In addition, their state changes continuously (dynamic) until an equilibrium point is reached. They remain at this point until the input changes and a new equilibrium needs to be found.

The MLP network is trained using a supervised learning algorithm like the backpropagation algorithm. The backpropagation algorithm uses data to adjust the network’s weights and thresholds so as to reduce the error in its prediction on the training set. It computes how fast the error, which is the difference between the actual and the desired activity, changes due to an alteration in: i) the activity of an output unit; ii) the total input received by an output unit; iii) weight on the connection into an output unit; and iv) the activity of a unit in the previous layer.

According to [9], some of the uses and applications of Artificial Neural Networks are for; classification, pattern matching, pattern completion, optimisation; control, function approximation/times series modelling, and data mining.

g) Related Works

Lizarraga-Morales et al. [10] proposed a novel FPGA-based methodology for early broken rotor bar (BRB) detection and classification through homogeneity estimation. Obtained results demonstrated the high efficiency of the proposed methodology as a deterministic technique for incipient BRB diagnosis in induction motors, which can detect and differentiate among half, one, or two BRBs with very high accuracy.

Kayri [11] did a comparative study on the predictive ability of Bayesian regularization with Levenberg-Marquardt artificial neural networks. Analysis were done by sum squared error (SSE), sum squared weight (SSW) and correlation of regression and concluded that the Bayesian regularization training algorithm shows better performance than the Levenberg-Marquardt algorithm.

Araujo et al. [12] provides an analysis about early incipient and recurring failures in three-phase induction motor bearings when driven by pulse width modulation inverters, focusing on a real industrial process. Over the investigation, it was concluded that the presence of common-mode currents at the verified levels could cause damages to the motor bearings, which was confirmed when the machine stopped working due to another bearing failure.

Yu et al. [13] developed a model-based remaining useful life (RUL) prediction method for induction motor with stator winding short circuit fault. The induction motor model with stator winding short circuit fault is introduced based on reference frame transformation theory. A particle filter method is used to realize unknown parameter estimation and RUL prediction. Simulation results were provided to validate the proposed method.

Kraleti et al. [14] presented a paper on model-based diagnostics and prognostics of three-phase induction motor for vapour compressor applications. Faults under consideration were incipient electrical faults: insulation degradation and broken rotor bars. Two online approximators were used to discover the system parameter degradation and facilitate fault isolation, or root-cause analysis, and a time to failure (TTF) prediction before the occurrence of a failure.

Ghate and Dudul [15] developed the radial-basis-function- multilayer- perceptron cascade connection neural-network based fault detection scheme for the small and medium sizes of three-phase induction motors. Simple statistical parameters of stator current were considered as input features and experimental results showed ability of the proposed classifier for detecting faults such as stator winding inter-turn short and/or rotor eccentricity. The network was tested for good classification accuracy with enough robustness to noises. The classifier was then found to be suitable for real world applications.

The use of ANN’s in predicting failure of the 5.8 MW 11 kV motor on nominal load provides a new area of research. The network is a generalised feed-forward network and the input data samples are current, winding temperatures and power all in the time domain.
II. Methods Used

In designing a model for the failure prediction of a 3 phase 5.8 MW, 11 kV slip-ring SAG Mill induction motor at Goldfields Ghana Limited, Damang mine, Artificial Neural Networks was employed in modelling and simulations on the data collected (power, current and winding temperatures) from the company. The materials used in this research for the collection of the motor data was the Citect and Laptop computer with MATLAB software (2017a) for modelling and simulation of the data.

Figure 4(a) [15] shows a generalized flow chart of ANN-based fault classification of induction motors. Designing ANN models follows a number of systemic procedures. In general, there are five basics steps: (1) collecting data, (2) pre-processing data, (3) building the network, (4) train, and (5) validate and test the performance of model as shown in Figure 4(b).

Figure 4: (a) General Flow of ANN-Based Fault Classification of Induction Motors; and (b) Basic Flow Chart for Designing Artificial Neural Network Model
b) **Data Collection**

Collecting and preparing sample data is the first step in designing ANN models. As it is outlined in Figure 4(b), measurement data of the SAG Mill motor power (MW), motor current (A), and winding temperatures (°C) with the corresponding motor condition i.e. motor healthy or motor faulty (MH/MF) for the Damang mine for a 93-day period from 6th January, 2019 to 8th April, 2019 was collected through the Citect as shown in Figure 5 (a). A total of $5 \times 879$ data samples were collected. Figure 5(b) show graphs of trends of the SAG Mill motor current and power.

*Figure 5:* (a) Trends of SAG Mill Motor Current, Power and Winding Temperatures; and (b) Graph Showing Trends of SAG Mill Motor Current and Power
c) **Data Pre-Processing**

After data collection, data pre-processing procedures are conducted to train the ANNs more efficiently. The procedure is normalisation of data. Normalization procedure before presenting the input data to the network is generally a good practice, since mixing variables with large magnitudes and small magnitudes will confuse the learning algorithm on the importance of each variable and may force it to finally reject the variable with the smaller magnitude [16]. Figure 6(a) and Figure 6(b) are graphs showing SAG Mill motor winding temperatures and winding temperatures after normalisation. A total of $5 \times 837$ data samples were considered healthy after normalisation.

Data samples which were out of range after normalisation were taken to be faulty data samples. This data samples totalled $5 \times 42$ faulty data samples. Figure 7 is a graph showing faulty data samples.
d) Building the network

At this stage, the designer specifies the number of hidden layers, neurons in each layer, transfer function in each layer, training function, weight/bias learning function, and performance function. In this work, the generalised feed-forward neural network was used.

Feed-Forward Neural Network with Backpropagation Algorithm

In feed-forward neural networks, otherwise known as multilayer perceptrons, the input vector of independent variables is related to the target (SAG Mill motor condition) using the architecture depicted in Figure 8. This figure shows one of the commonly used networks, namely; the layered feed-forward neural network with one hidden layer. Here, each single neuron is connected to those of a previous layer through adaptable synaptic weights. Knowledge is usually stored as a set of connection weights, and then, the weights are adjusted so that the network attempts to produce the desired output. The weights after training contain meaningful information, whereas before training they are random and have no meaning.

![Diagram of Feed-Forward Network](image)

Figure 8: Architecture of Feed-Forward Network

e) Training the network

Training is the process of modifying the network using a learning mode, in which an input is presented to the network along with the desired output. During the training process, the weights are adjusted in order to make the actual outputs (predicted) close to the target.
(measured) outputs of the network. In this study, 70% of the data was used for training. Two different types of training algorithms were investigated for developing the feed-forward network. These are Levenberg-Marquardt algorithm and Bayesian Regularisation algorithm. MATLAB provides built-in transfer functions, which are used in this study; Linear (purelin), Hyperbolic Tangent Sigmoid (tansig) and Logistic Sigmoid (logsig).

**f) Validating and Testing the Network**

The next step is to validate and test the performance of the developed model. At this stage, unseen data are fed to the model. For this case study, 15% of SAG Mill motor data was used for validating and another 15% used for testing the ANN models. Validation data generalise the network validation and stops training before overfitting, which occurs when a network memorises the training data but not learn to generalise new inputs.

In order to evaluate the performance of the developed ANN models quantitatively and verify whether there is any underlying trend in performance of ANN models, statistical analysis involving mean squared error were conducted. MSE provides information on the short-term performance, which is a measure of the variation of predicted values around the measured data. The lower the MSE, the more accurate is the estimation. The expressions for the aforementioned statistical parameter is:

\[
\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (I_p - I_i)^2
\]

where
- \(I_p\) denotes the predicted power of SAG Mill motor in MW;
- \(I_i\) denotes the measured power of SAG Mill motor in MW; and
- \(n\) denotes the number of observations.

On the other hand, regression is a statistical analysis assessing the correlation between two variables. The regression line equation can be written as:

\[
y = a + bx
\]

\[
b = \frac{N \sum XY - (\sum X)(\sum Y)}{\left(N \sum X^2 - (\sum X)^2\right)}
\]

\[
a = \frac{\sum Y - b(\sum X)}{N}
\]

where
- \(a\) = the y intercept when \(x = 0\);
- \(b\) = the slope/gradient of the line;
- \(N\) = number of data samples;
- \(X\) = first group; and
- \(Y\) = second group and regression coefficient.

\[R = \frac{\sum XY - \frac{\sum X \sum Y}{N}}{\sqrt{\left(\frac{\sum X^2}{N}\right) - \left(\frac{(\sum X)^2}{N}\right)}}\]

\[R^2 = \frac{(\sum XY - \frac{\sum X \sum Y}{N})^2}{\sum X^2 - \frac{(\sum X)^2}{N}}\]

\[R^2 = \frac{(\sum XY - \frac{\sum X \sum Y}{N})^2}{\sum Y^2 - \frac{(\sum Y)^2}{N}}\]

**g) Programming the Neural Network Model**

MATLAB is a numerical computing environment and also a programming language. It allows easy matrix manipulation, plotting of functions and data, implementation of algorithms, creating user interfaces and interfacing with programs in other languages. The Deep Learning Toolbox (formerly Neural Network Toolbox) provides a framework for designing and implementing deep neural networks with algorithms, pretrained models, and apps. Apps and plots help you visualize activations, edit network architecture, and monitor training progress (The Math Works, 2019).

Figure 9(a) show the screen captions of the FFNN ANN training window obtained using the “nntraintool” GUI toolbox in MATLAB. Figure 9(b) show the flow chart for the development of the feed forward network using MATLAB.
The proposed methodology is implemented using a microprocessor to achieve online failure detection. In addition to the cost-effectiveness of the microprocessor implementation, it is flexible and its reconfigurability allows changes and refinements while in operation.

Figure 10 shows the block diagram of the proposed methodology implementation. The data acquisition system receives current, power and three winding temperature signals from the sensors connected to the power supply to the stator windings of the motor. Signal processing is performed by the microprocessor and the result further analysed by a postprocessor decision-making block that simply states the motor condition in two possible values, i.e., MH (a healthy motor) and MF (a faulty motor), making the process online with no expert technician required for the diagnosis.
III. RESULTS AND DISCUSSION

The results of MATLAB simulations using Artificial Neural Network tool box of SAG Mill motor current, temperature and power data from Goldfields Damang Mine are presented here.

a) Simulation Results using Feed-Forward Network

In this section, results of using current and winding temperature readings representing three sides of the SAG Mill motor is used as the input to the network with Mill motor power as the target of the network. Two training algorithms i.e. Levenberg-Marquardt (LM) and Bayesian Regularization (BR) were used in training the network. Simulation results of correlation coefficient for network performance (R), mean squared error (MSE) against epochs, error histograms and training state plot for model validation are presented here.

i. Simulation Results of FFNN Using Levenberg-Marquardt Training Algorithm
Figure 11: (a) Correlation Coefficient for Network Performance, R (LM); and (b) Mean Squared Error (MSE) against Epochs (LM)
ii. Simulation Results of FFNN Using Bayesian Regularisation Training Algorithm

Figure 12: (a) Error Histogram (LM); and (b) Training State Plot for Model Validation (LM)

Figure 13: Correlation Coefficient for Network Performance, R (BR)
Figure 14: (a) Mean Squared Error (MSE) against Epochs (BR); and (b) Error Histogram (BR)
b) Discussion of Simulation Results

This section also presents discussions of the MATLAB simulated results using FFNN. Table 2 shows the computed values of mean squared error (MSE) and correlation coefficient of network performance, $R$. It shows values of MSE and $R$ for different number of data samples for training, validation and testing of the generated FFNN. The data samples range from 100, 200, 300, 400 and 500.

**Table 2:** Statistical Error Parameters of Developed FFNN Models for Different Data Sample Size

<table>
<thead>
<tr>
<th>Number of Data Samples</th>
<th>Levenberg - Marquardt Algorithm</th>
<th>Bayesian Regularisation Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A/F - LOGSIG</td>
<td>A/F - TANSIG</td>
</tr>
<tr>
<td></td>
<td>MSE</td>
<td>R</td>
</tr>
<tr>
<td>100</td>
<td>1.48E-04</td>
<td>0.99724</td>
</tr>
<tr>
<td>200</td>
<td>1.03E-04</td>
<td>0.99894</td>
</tr>
<tr>
<td>300</td>
<td>1.33E-04</td>
<td>0.99904</td>
</tr>
<tr>
<td>400</td>
<td>8.93E-04</td>
<td>0.99565</td>
</tr>
<tr>
<td>500</td>
<td>0.0064</td>
<td>0.96889</td>
</tr>
</tbody>
</table>

In this study, the network was decided to consist of one hidden layer with 10 neurons. The criterion $R$ and MSE were selected to evaluate the networks to find the optimum solution. The complexity and size of the network was also an important consideration and therefore smaller ANN’s had to be selected. A regression analysis between the network response and the corresponding targets was performed to investigate the network response in more detail. Thus, LM and BR were selected. The R-values in Table 3 represent the correlation coefficient between the outputs and targets. The R-values did not increase beyond 10 neurons in the hidden layer. Consequently, the network with 10 neurons in the hidden layer would be considered satisfactory. From all the networks trained, few ones could provide the low error condition, from which the simplest network was selected. The results showed that, the training algorithm of LM was sufficient for predicting SAG Mill motor failures. There is a high correlation between the predicted values by the ANN model and the measured values collected from normal real time running of 5.8 MW, 11 kV SAG Mill motor, which imply that the model succeeded in prediction of SAG Mill motor failures.

It can be observed in Figs. (11a, b, 12a, b and 13) that, the ANN provided the best accuracy in modelling induction motor failures with correlation coefficients of 0.999 and 0.998 for LM and BR respectively. Generally, the ANN offers the advantage of being fast, accurate and reliable in the prediction of...
approximation affairs, especially when numerical and mathematical methods fail. There is also a significant simplicity in using ANN due to its power to deal with multivariate and complicated problems.

The measured values collected from real time, on load running of the 5.8 MW, 11 kV SAG Mill motor showed some linearity between the current, temperatures and the power. The power of the SAG Mill motor at nominal load ranges from 4 MW – 5.6 MW with the current and temperatures reading 300 A – 349 A and 80°C – 109°C respectively.

From Table 2, it can be seen that, the ANN showed good R and MSE values when data samples of 300 was used. This was the same for LM and BR, while using log-sigmoid and tan-sigmoid as transfer functions for the hidden layer. The results for R-values for data samples of 300 were 0.99904, 0.99911, 0.99831 and 0.99701 respectively, while MSE values were 1.33E-04, 1.74E-04, 3.70E-04 and 3.03E-04 respectively. This simulation was repeated for data samples of 100, 200, 400 and 500. It was observed that, increasing the number of data samples resulted in bad R-values. Data samples of 100 gave better results than 200, 200 than 400 and 400 than 500 in that order.

Training stops after 251 iterations. At this position, performance of network, $150 \times 10^{-4}$, gradient decrease to $3.55 \times 10^{-4}$ and also the value of $\mu = 10^{-7}$ as shown in Figure 12(b). Validation performance reached the minimum at epoch 201. The training continued for 51 more iterations and stopped at epoch 251. The gradient and $\mu$ increased gradually as shown in Figure 12(b).

From the error histogram shown in Figure 12(a), most errors occurred between −0.04 to +0.05. Errors also occurred at 0.065, 0.087 and 0.094 of the training data on the histogram, and also represents the point for which output 4.5 – target value 4.6, output 4.8 – target 4.9 and output 5.1 – target 5.2 on the training correlation coefficient for network performance, plot shown in Figure11(a).

c) Discussions on Using the Network for Prediction

Two matrices of $5 \times 669$ and $5 \times 31$ constructed by power, current and three winding temperature values normalised sample data of SAG Mill motor at healthy and faulty on load condition respectively as input are used to analyse network performance. Among them, 70%, 15% and rest data are used as training, cross validation and testing data. The target of the network is 1 or 0, with 1 indicating healthy motor condition and 0 indicating faulty motor condition. For any output value between 1 and 0 represents the probability of fault condition, in training the network, there was 1 hidden layer with 10 neurons and tansigmoid as the transfer function. The output layer had 1 neuron and the transfer function was logsigmoid.
Table 3: Detection Accuracy

<table>
<thead>
<tr>
<th>TOTAL NUMBER DATA SAMPLES</th>
<th>HEALTHY</th>
<th>FAULTY</th>
<th>ACCURACY</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TD</td>
<td>FD</td>
<td>TD</td>
</tr>
<tr>
<td>Levenberg-Marquardt</td>
<td>169/169</td>
<td>0/169</td>
<td>10/10</td>
</tr>
<tr>
<td>Bayesian Regularization</td>
<td>169/169</td>
<td>0/169</td>
<td>9/10</td>
</tr>
</tbody>
</table>

Figure 16: (a) Plot of Confusion Matrix Using Levenberg-Marquardt Algorithm; and (b) Plot of Confusion Matrix Using Bayesian Regularization Algorithm
d) Summary of Findings

The findings as regards simulations of data samples measured on the 5.8 MW, 11 kV SAG Mill motor at the Goldfields Ghana Ltd., Damang Mine from 6th January, 2019 to 8th April, 2019 are summarised as follows:

1. A smaller Feed-Forward Neural Network size of 4-10-1 provides optimum performance for prediction of SAG Mill motor failures;

2. Though Bayesian Regularisation training algorithm has not been extensively used in failure prediction of three phase slip-ring induction motor as compared to Levenberg-Marquardt, yet it gives acceptable results in terms of accuracy but at a relatively low efficiency;

3. Data samples of 100, 200, 300, 400 and 500 were used in this work. Data samples of 300 with Levenberg-Marquardt training algorithm and tansigmoid activation function of the hidden layer provided the best results for R-values and MSE;

4. The network stopped training at 251 iterations, a network performance of $150 \times 10^{-4}$ at this position. The gradient decreases to $3.55 \times 10^{-4}$ and $\mu = 10^{-7}$. Validation performance reaches minimum at epoch 201; and

5. The network with Levenberg-Marquardt training algorithm can detect healthy and faulty conditions of the SAG Mill motor with 100% accuracy and 99.4% using Bayesian Regularization as the training algorithm.

IV. Conclusions and Recommendations

a) Conclusions

From the results and discussions, the following conclusions can be drawn:

1. The proposed Feed-Forward Neural Network with Levenberg-Marquardt training algorithm is capable of predicting imminent faults on load 5.8MW, 11 kV SAG Mill three phase slip-ring induction motor at Goldfields Ghana Ltd., DamangMine with 100% accuracy;

2. Correlation coefficient of network performance, $R$ and mean squared error, MSE proved to be very good statistical tools for artificial neural network model analysis; and

3. Bayesian Regularisation training algorithm proved to be a good alternative to Levenberg-Marquardt algorithm in failure prediction networks.

b) Recommendations

The following are recommended based on the conclusions drawn:

1. Similar research could be carried out on the Ball Mill motor and other important motors at the plant;

2. Since it will be very difficult to set up a prototype for the 3 phase 5.8 MW, 11 kV slip-ring induction motor taking into consideration the size and cost, MATLAB/SIMULINK and Finite Element Method Magnetics could be considered in generating signals for this research; and

3. Wavelet techniques and Fuzzy logic could be used to find exact location of fault, identification and evaluation of fault severity.

References Références Referencias


MONSDA: - A Novel Multi-Objective Non-Dominated Sorting Dragonfly Algorithm

Pradeep Jangir

Abstract- This novel article presents the multi-objective version of the recently proposed Dragonfly Algorithm (DA) known as Non-Dominated Sorting Dragonfly Algorithm (NSDA). This proposed NSDA algorithm works in such a manner that it first collects all non-dominated Pareto optimal solutions in achieve till the evolution of last iteration limit. The best solutions are then chosen from the collection of all Pareto optimal solutions using a crowding distance mechanism based on the coverage of solutions and swarming strategy to guide dragonflies towards the dominated regions of multi-objective search spaces. For validate the efficiency and effectiveness of proposed NSDA algorithm is applied to a set of standard unconstrained, constrained and engineering design problems. The results are verified by comparing NSDA algorithm against Multi objective Colliding Bodies Optimizer (MOCBO), Multi objective Particle Colliding Bodies Optimizer (MOPSO), non-dominated sorting genetic algorithm II (NSGA-II) and Multi objective Symbiotic Organism Search (MOSOS). The results of proposed NSDA algorithm validates its efficiency in terms of Execution Time (ET) and effectiveness in terms of Generalized Distance (GD), Diversity Metric (DM) on standard unconstrained, constraint and engineering design problem in terms of high coverage and faster convergence.

Keywords: non-dominated; crowing distance; NSDA algorithm; multi-objective algorithm; economic constrained emission dispatch.

I. Introduction

Optimization is a work of achieving the best result under given limitation or constraints. Now a day, optimization is used in all the fields like construction, manufacturing, controlling, decision making, prediction etc. The final target is always to get feasible solution with minimum use of resources. In this field computers make a revolutionary impact on every field as it provides the facility of virtual testing of all parameters that are involved in a particular design with less involvement of human efforts, benefits in less time consuming, human efforts and wealth as well.

Today we use computer-aided design where a designer designs a virtual system on computer and gives only command to test all parameters involved in that design without even the need for a single prototype. A designer only to design and simulate a system and set all the parameter limitation for the computer.

Computer-aided design technique becomes more effective with the additional feature of automatic generation of solutions after it’s mathematically formulation of any system or design problem. Auto generation of solution, this feature is come into nature with the development of algorithms. In past years, real world designing problems are solved by gradient descent optimization algorithms. In gradient descent optimization algorithm, the solution of mathematically formulated problem is achieved by obtaining its derivative. This technique is suffered from local minima stagnation [1, 2] more time consuming and their solution is highly dependent on their initial solution.

The next stage of development of optimization algorithms is population based stochastic algorithms. These algorithms had number of solutions at a time so embedded with a unique feature of local minima avoidance. Later population based algorithms are developed to solve single objective at a time either it may be maximization or minimization on accordance the problems objective function. Some popular algorithms for single objective problems are Moth-Flame optimizer (MFO) [3], Bat algorithm (BA) [4], Particle swarm optimization (PSO) [5], Ant colony optimization (ACO) [6], Genetic algorithm (GA) [7], Cuckoo search (CS) [8], Mine blast algorithm (MBA) [9], Krill Herd (KH) [10], Interior search algorithm (ISA) [11] etc. These algorithms have capabilities to handle uncertainties [12], local minima [13], misleading global solutions [14], better constraints handling [15] etc. To overcome these difficulties different algorithms are enabled with different powerful operators. As mention above here is only objective then it is easy to measure the performance in terms of speed, accuracy, efficiency etc. with the simple operational operators.

In general, real world problems are nonlinear and multi-objective in nature. In multi-objective problem there may be some objectives are consisting of maximization function while some are minimization function. So now a day, multi-objective algorithms are in firm attention.

Let’s take an example of buying a car, so we have many objectives in mind like speed, cost, comfort level, space for number of people riding, average fuel consumption, pick up time required to gain particular speed, type of fuel requirement either it is diesel driven, petrol driven or both etc. To simply understand multi-objective problem, from Fig. 1, we consider two objectives, first cost and second comfort level. So we go for sole objective of minimum cost possible then we...
have to deny comfort level objective and vice-versa. It means real world problems are with conflicting objectives. So as, we are disabled to find an optimal solution like single objective problems. About multi-objective algorithm and its working is detailed described in next portion of the article.

Fig. 1: Car-buying decision-making problem (Hypothetical Optimal solutions)

The No free lunch [16] theorem that logically proves that none of the only algorithm exists equally efficient for all engineering problem. This is the main reason that it allows all researcher either to propose new algorithm or improve the existing ones. This paper proposed the multi-objective version of the well-known dragonfly algorithm (DA) [17]. In this paper non-sorted DA (NSDA) is tested on the standard un-constraint and constraint test function along with some well-known engineering design problem, their results are also compared with contemporary multi-objective algorithms Multi objective Colliding Bodies Optimizer (MOCBO)[18], Multi objective Particle Swarm Optimizer (MOPSO)[19-20], Non-dominated Sorting Genetic Algorithm (NSGA) [21-23], non-dominated sorting genetic algorithm II (NSGA-II)[24] and Multi objective Symbiotic Organism Search (MOSOS)[25]that are widely accepted due to their ability to solve real world problem.

The structure of the paper can be given as follows: - Section 2 consists of literature; Section 3 includes the proposed novel NSDA algorithm; Section 4 consists of competitive results analysis of standard test functions as well as engineering design problem and section 5 includes real world application, finally conclusion based on results and future scope of work is drawn.

II. Literature Review

As the name describes, multi-objective optimization handles simultaneously multiple objectives. Mathematically minimize/maximize optimization problem can be written as follows:

\[
\text{Minimize/maximize: } \quad F\vec{n}(\vec{x}) = \{f_{n_1}(\vec{x}), f_{n_2}(\vec{x}), \ldots, f_{n_0}(\vec{x})\} \\
\text{Subject to: } \quad p_i(\vec{x}) \geq 0, \quad i = 1,2,\ldots,q \quad (2.1) \\
\text{subject to: } \quad t_i(\vec{x}) = 0, \quad i = 1,2,\ldots,r \quad (2.2) \\
U_{lb}^i \leq x_i \leq U_{ub}^i, \quad i = 1,2,\ldots,k \quad (2.3)
\]

Where \(q\) is the number of inequality constraints, \(r\) is the number of equality constraints, \(k\) is the number of variables, \(p_i\) is the \(i^{th}\) inequality constraints, \(n_0\) is the number of objective functions, \(t_i\) indicates the \(i^{th}\) equality constraints, and \([U_{lb}^i, U_{ub}^i]\) are the boundaries of \(i^{th}\) variable.

Obviously, relational operators are ineffective in comparing solutions with respect to multiple objectives.
The most common operator in the literate is Pareto optimal dominances, which is defined as follows for minimization problems:

\[
\forall n \in \{1,2,\ldots,k\}: f_n(x) \leq f_n(y) \land \exists n \in \{1,2,\ldots,k\}: f_n(x) < f_n(y)
\]

where \(x = (x_1, x_2, \ldots, x_k)\) and \(y = (y_1, y_2, \ldots, y_k)\).

For maximization problems, Pareto optimal dominance is defined as follows:

\[
\forall n \in \{1,2,\ldots,k\}: f_n(x) \geq f_n(y) \land \exists n \in \{1,2,\ldots,k\}: f_n(x) > f_n(y)
\]

where \(x = (x_1, x_2, \ldots, x_k)\) and \(y = (y_1, y_2, \ldots, y_k)\).

These equations show that a solution is better than another in a multi-objective search space if it is equal in all objective and better in at least one of the objectives. Pareto optimal dominance is denoted with \(<\) and \(>)\). With these two operator’s solutions can be easily compared and differentiated.

Population based multi-objective algorithm’s solution consists of multiple solution. But with multi-objective algorithm we cannot exactly determine the optimal solution because each solution is bounded by other objectives or we can say there is always conflict between other objectives. So the main function of stochastic/population based multi-objective algorithm is to find out best trade-offs between the objectives, so called Pareto optimally set [26-28].

The principle of working for an ideal multi-objective optimization algorithm is as shown in Fig. 2.

**Step No. -1** Find maximum number of non-dominated solution according to objective, it expresses the number of Pareto optimal set so as shows higher coverage.

**Step No. -2** Choose one of the Pareto optimal solution using crowding distance mechanism that fulfills the objectives.

![Fig. 2: Multi-objective optimization (Ideal) procedure.](image-url)
Now a day recently proposed sole objective algorithms are equipped with powerful operators to provide them a capability to solve multi-objective problems as well. In the same manner we proposed NSDA algorithm in a hope that it will perform efficiently for multi-objective problems. These are: Multi-objective GWO [29], Multi-objective Bat Algorithm [30], Multi-objective Bee Algorithm [31], Pareto Archived Evolution Strategy (PAES) [32], Pareto-frontier Differential Evolution (PDE) [33], Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D) [34], Strength-Pareto Evolutionary Algorithm (SPEA) [35, 36] and Multi-objective water cycle algorithm with unconstraint and constraint standard test functions [37][38]. Performance measurement for approximate robustness to Pareto front of multi-objective optimization algorithms in terms of coverage, convergence and success metrics.

The computational complexity of NSDA algorithm is order of $O(mn^2)$ where $N$ is the number of individuals in the population and $M$ is the number of objectives. The complexity for other good algorithms in this field: NSGA-II, MOPSO, SPEA2 and PAES are $O(mn^2)$. However, the computational complexity is much better than some of the algorithms such as NSGA and SPEA which are of $O(mn^3)$.

III. Non-Dominated Sorting Dragonfly Algorithm (NSDA)

Dragonfly Algorithm (DA) with sole objective was proposed by Mirjalili Seyedali in 2015 [17]. It is basically a stochastic population based, nature inspired algorithm. In this algorithm the basic strategy based on swarming nature of dragonflies for exploration and exploitation. DA algorithm originated from the static and dynamic swarming behaviors of dragonflies. These two swarming behaviors are similar to the basic stage of working of any optimization algorithm in all meta-heuristic algorithms as: exploration and exploitation. Dragonflies build small number of group and fly in different directions in search of food is known as static swarm, this function is very similar to exploration phase in meta-heuristic techniques. Whereas, dragonflies make a big group and fly in only direction for either attacking to prey or migration to other place is known as dynamic swarm, this function is very similar to exploitation phase.

**Mathematical modelling of Dragonfly Algorithm:**

Each portion of Dragonfly Algorithm is formulated by algebraic equations are:

1. For Separation part formulating equation:

$$\text{SEP.}_j = \sum_{i=1}^{N} L_i - L_j$$

2. For Alignment part formulating equation:

$$\text{Alig.}_j = \frac{\sum_{i=1}^{N} L_i}{N}$$

3. For cohesion part formulating equation:

$$\text{Coh.}_j = \frac{\sum_{i=1}^{N} L_i}{N} - L$$
4. For Attraction towards a food source part formulating equation:
   \[ F_j = L^+ - L \]  
   (3.4)

5. For Attraction towards a food source part formulating equation:
   \[ E_j = L^- + L \]  
   (3.5)

6. Step vector is formulating equation:
   \[ \Delta L_{t+1} = (sSep. _j + aAlig. _j + cCoh. _j + fF_j + eE_j) + w\Delta L_t \]  
   (3.6)

7. Position vector is calculated using equation
   \[ L'^d_{t+1} = L_t + \Delta L_{t+1} \]  
   (3.7)

8. Position of dragonfly updated using equation
   \[ L_{t+1} = L_t + \text{Levy}(L) \ast L_t \]  
   (3.8)

Where:
- L = Location of the current individuals
- N = Neighboring individuals
- \( L^+ \) = positions of food source
- \( L^- \) = positions of enemy
- s = separation weight
- a = alignment weight
- c = cohesion weight
- f = food weight
- e = enemy weight
- w = inertia weight
- t = iteration counter
- d = dimension of position vectors that levy flight step calculated

![Dragonfly Algorithm principle](image)

**Fig. 4: Dragonfly Algorithm principle**

Basic working of NSDA algorithm is as follows:

- **Stage 1**
  - First of all, initialize the population of dragonflies
  - Randomly generated sets of dragonflies & position vectors are represented in matrix for convenience to understand
  - Then fitness of step vector & position vectors calculated on an according as objective function

- **Stage 2**
  - Position of dragonflies are updated as a function of levy flight motion and so as value of position vector is decided

- **Stage 3**
  - The value of absolute distance is achieved which is basically a distance between the current best solution to the final optimal solution
  - Step vector is a function of both static and dynamic swarming behavior of dragonflies where some constant weight is assigned to the step vector function according to their swarming nature
  - Termination counter in integrated to limit/forcefully stop the search in uncertain search space (max. iteration counter to forcefully converge the search to optimal one)
Size of the position vector matrix is continuously reduced over the course of iteration due to directed search to find global best solution.

- Continuously position of the dragonflies are updated towards the optimal one via a Levy function or position vector equation for each iteration.

**Step 4**

- Likewise, multi-objective optimization the NSDA algorithm is made to capable to store the pare to optimal solutions in a collection set and make it as flexible to change solution over the course of iteration.

- Solution is assigned a rank according to their ability as if a solution is not dominated by other solution is assigned rank 1, dominated by only solution assigned rank 2 and so on & if collection set is full (archive size) over predefined size then some solutions that are less non-dominated (according to fitness value) in nature are directed to be out from the collection set according to the crowding distance mechanism.

This collection set is similar to the term achieve used in MOSOS and NSGA-II. It is a repository to store the best non-dominated solutions obtained so far. The search mechanism in NSDA is very similar to that of DA in which solutions are improved using step vectors. Due to the existence of multiple best solutions, however, the best dragon flies position should be chosen from the collection set.

In order to select solutions from the archive to establish tunnels between solutions, we employ a leader selection mechanism. In this approach, the crowding distance between each solution in the archive is first selection and the number of solutions in the neighbourhood is counted as the measure of coverage or diversity. We require the NSDA to select solutions from the less populated regions of the archive using the following equation to improve the distribution of solutions in the archive across all objectives.

This subsection proposes multi-objective version of the DA algorithm called NSDA algorithm. The non-dominated sorting has been of the most popular and efficient techniques in the literature of multi-objective optimization. As its name implies, non-dominated sorting sort Pareto optimal solutions based on the domination level and give them a rank. This means that the solutions that are not dominated by any solutions is assigned with rank 1, the solutions that are dominated by only one solution are assigned rank 2, the solutions that are dominated by only two solutions are assigned rank 3, and so on. Afterwards, solutions are chosen to improve the quality of the population base on their rank. The better rank, the higher probability to be chosen. The main drawback of non-dominated sorting is its computational cost, which has been resolved in NSGA-II.

The success of the NSGA-II algorithm is an evidence of the merits of non-dominated sorting in the field of multi-objective optimization. This motivated our attempts to employ this outstanding operator to design another multi-objective version of the DA algorithm. In the NSDA algorithm, solutions are updated with the same equations presented in equation 3.9. In every iteration, however, the solutions to have optimal position of dragonflies are chosen using the following equation:

$$P_i = \frac{c}{\text{Rank}_i}$$

where $c$ is a constant and should be greater than 1 and $\text{Rank}_i$ is the rank number of solutions after doing the non-dominated sorting.

This mechanism allows better solutions to contribute in improving the solutions in the population. It should be noted that non-dominated sorting gives a probability to dominated solutions to be selected as well, which improves the exploration of the NSDA algorithm. Flow chart of NSDA algorithm is represented as Fig. 5.

**Constraint Handling Approach:**

With the extended literature survey we find that the population based algorithms are the common way to solve the multi-objective problems as they are more commonly provides the global solution and capable of handling both continuous and combinational optimization problem with a very high coverage and convergence. Multi-objective problems are subjected to various type of constraints like linear, non-linear, equality, inequality etc. So with these problems embedded it is very difficult to find simple and good strategy to achieve considerable solutions in the acceptable criterion. So in this paper NSDA algorithm uses a very simple approach to get feasible solutions. In this mechanism, after generating number of solutions at each generation, all the desirable constraint checked and then some solution that fulfills the criterion of acceptable solution are selected and collected them in achieve. Afterward non dominated solutions added in archive as we find more suitable solution to get acceptable solution. So as if achieve is full then less dominated solutions are removed. Finally according to crowding distance mechanism all these solutions (more suitable position of dragonflies) from archive is selected to get desired solution.
IV. Results Analysis on Test Functions

For determine the performance of proposed NSDA algorithm is applied to:

- A set of unconstrained and constraint standard multi-objective test functions
- Tested on well-known engineering design problems
- Non-linear, highly complex practical application known as formulation of economic constrained emission dispatch (ECED) with stochastic integration of wind power (WP) in the next section

NSDA algorithm is tested on seventeen different multi-objective case studies, including eight unconstrained test functions, five constrained test functions, and four real world engineering design problem, later algorithm is applied to the main application economic constrained emission dispatch with wind power (ECEDWP). These can be classified into four groups given below:
MONSDA: A Novel Multi-Objective Non-Dominated Sorting Dragonfly Algorithm

- Standard multi-objective unconstrained test functions (KUR, FON, ZDT1, ZDT2, ZDT3, ZDT4, SCHN1, and SCHN2)
- Standard multi-objective constrained test functions (TNK, OSY, BHN, SRN, and CONST)
- Real world engineering multi-objective design problem (Four bar truss design, welded beam design, speed reducer and disk brake design problem)
- Modeling of ECEDWP problem

Mathematical representation of these standard test functions are given in Appendix 1. (Multi-objective unconstrained test functions), 2. (Multi-objective constrained test functions), 3. (Engineering multi-objective design problem) with distinct characteristics like non-linear, non-convex, discrete pareto fronts and convex etc. are selected to measure the performance of proposed NSDA algorithm. To deal with real world engineering design problem is really a typical task with unknown search space, in this article we includes four different engineering problems are considered and performance is compared with various well known algorithms like MOWCA, NSGA-II, MOPSO, PAES and μ-GA multi-objective algorithms. Each algorithm is separately runs fifteen times and numeric results are listed in tables below. To measure the quality of obtained results we match their coverage of obtained true pareto front with respect to their original or true pareto fronts.

For numeric as well as qualitative performance of proposed NSDA algorithm on various case studies we consider Generational Distance (GD) given by Veldhuizen in 1998 [39]for measuring the deviation of the distance between true pareto front and obtained pareto front, Diversity metric (Δ) also known as matrix of extreme solutions in true pareto front and obtained solution achieved and the nearest true Pareto optimal solutions. Where “d” is the average of all distances between true pareto front and obtained pareto fronts. All the statistical results are shown in Table 1 represents a greater robustness and accuracy of NSDA algorithm in terms of mean and standard deviation with the help of GD, diversity matrix along with computational time. However, proposed NSDA algorithm shows very competitive results in comparison with the MOSPO, MOCBO and MOSOS algorithms and in some cases these algorithms performs better than proposed one. Pareto front obtained by proposed NSDA algorithm shows almost complete coverage with respect to true pareto front.

\[
\Delta = \frac{d_i + d_m + \sum_{i=1}^{n_{PFs}} |d_i - \bar{d}|}{d_i + d_m + (n-1)d}
\]

where \(d_i\) shows the Euclidean distance (calculated in the objective space) between the \(i^{th}\) Pareto optimal solution achieved and the nearest true Pareto optimal solution in the reference set, \(n_{PFs}\) is the total number of achieved Pareto optimal solutions.

\[
S = \frac{1}{n_{PFs}} \sum_{i=1}^{n_{PFs}} (d_i - \bar{d})^2
\]

where \(d_i\) is the average of all \(d_i\), \(n_{PFs}\) is the total number of achieved Pareto optimal solutions, and \(d_i = \min\{(f_i^1(x) - f_i^1(\bar{x})) + (f_i^2(x) - f_i^2(\bar{x}))\} \) for all \(i,j=1,2,\ldots,n\). Smallest value of “S” metric gives the global best non-dominated solutions are uniformly distributed, thus if numeric value of \(d_i\) and \(d\) are same then value of “S” metric is equal to zero.

**a) Results on unconstrained test problems**

Like as above mentioned, the first set of test problems consist of unconstrained standard test functions. All the standard unconstrained test functions mathematical formulation is shown in Appendix A. Later, the numeric results are represented in Table 1 and best optimal pareto front is shown in Fig. 6.

All the statistical results are shown Table 1 suggests that the NSDA algorithm effectively outperforms with most of the unconstraint test functions compare to the MOSOS, MOCBO, MOSPO and NSGA-II algorithm. The effectiveness of proposed non-dominated version of DA (NSDA algorithm) can be seen in the Table 1, represents a greater robustness and accuracy of NSDA algorithm in terms of mean and standard deviation with the help of GD, diversity matrix along with computational time. However, proposed NSDA algorithm shows very competitive results in comparison with the MOSPO, MOCBO and MOSOS algorithms and in some cases these algorithms performs better than proposed one. Pareto front obtained by proposed NSDA algorithm shows almost complete coverage with respect to true pareto front.

### Table 1: Results of the multi-objective NSDA algorithms (using GD, Δ, CT) on the unconstrained test functions employed

<table>
<thead>
<tr>
<th>Algorithm Function ↓</th>
<th>PFs</th>
<th>NSDA MEAN±SD</th>
<th>MOSOS MEAN±SD</th>
<th>MOCBO MEAN±SD</th>
<th>MOSPO MEAN±SD</th>
<th>NSGA-II MEAN±SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>KUR</td>
<td>GD</td>
<td>0.00729±0.00032</td>
<td>0.0075±0.00042</td>
<td>0.0083±0.00062</td>
<td>0.015±0.00075</td>
<td>0.0301±0.00043</td>
</tr>
<tr>
<td>Δ</td>
<td></td>
<td>0.02704±0.01025</td>
<td>0.0295±0.0122</td>
<td>0.0357±0.0236</td>
<td>0.0991±0.031</td>
<td>0.0362±0.0240</td>
</tr>
<tr>
<td>CT</td>
<td></td>
<td>7.65853±0.04369</td>
<td>10.7413±0.822</td>
<td>7.9531±0.5823</td>
<td>8.0532±0.621</td>
<td>20.4368±3.102</td>
</tr>
<tr>
<td>GD</td>
<td></td>
<td>0.00173±0.00032</td>
<td>0.0019±0.0002</td>
<td>0.0022±0.0003</td>
<td>0.0042±0.000</td>
<td>0.0026±0.0003</td>
</tr>
<tr>
<td>Function</td>
<td>Δ</td>
<td>CT</td>
<td>GD</td>
<td>Δ</td>
<td>CT</td>
<td>GD</td>
</tr>
<tr>
<td>----------</td>
<td>------</td>
<td>-----</td>
<td>-----</td>
<td>------</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>FON</td>
<td>0.29805±0.03758</td>
<td>0.3875±0.0062</td>
<td>0.3955±0.0068</td>
<td>0.4158±0.008</td>
<td>0.3987±0.0082</td>
<td></td>
</tr>
<tr>
<td>ZDT-1</td>
<td>0.35589±0.00875</td>
<td>0.3803±0.0122</td>
<td>0.3825±0.0125</td>
<td>0.3876±0.024</td>
<td>0.3905±0.0220</td>
<td></td>
</tr>
<tr>
<td>ZDT-2</td>
<td>0.35879±0.01478</td>
<td>0.4585±0.0073</td>
<td>0.4795±0.0079</td>
<td>0.6543±0.024</td>
<td>0.7003±0.0089</td>
<td></td>
</tr>
<tr>
<td>ZDT-3</td>
<td>0.69874±0.23568</td>
<td>0.6537±0.0052</td>
<td>0.65325±0.002</td>
<td>0.8234±0.018</td>
<td>0.7386±0.0474</td>
<td></td>
</tr>
<tr>
<td>ZDT-4</td>
<td>0.50078±0.01578</td>
<td>0.5295±0.1312</td>
<td>0.5302±0.1356</td>
<td>0.8582±0.164</td>
<td>0.5502±0.1360</td>
<td></td>
</tr>
<tr>
<td>SCHN-1</td>
<td>0.00904±0.00070</td>
<td>0.0028±0.0024</td>
<td>0.0031±0.0032</td>
<td>0.0032±0.003</td>
<td>0.0034±0.0042</td>
<td></td>
</tr>
<tr>
<td>SCHN-2</td>
<td>0.04476±0.00189</td>
<td>0.0705±0.0215</td>
<td>0.0932±0.0228</td>
<td>0.1497±0.022</td>
<td>0.3096±0.0217</td>
<td></td>
</tr>
</tbody>
</table>

**Graphs:**
- NSDA "KUR" (True PF vs. Obtained PF)
- NSDA "ZDT-1" (True PF vs. Obtained PF)
- NSDA "ZDT-2" (True PF vs. Obtained PF)
- NSDA "ZDT-3" (True PF vs. Obtained PF)
- NSDA "ZDT-4" (True PF vs. Obtained PF)
b) Results on constrained test problems

The next set of standard test functions consisting of constrained functions. For constrained test function it should be necessary that NSDA algorithm has a capability of handling constraints so algorithm is equipped with a death penalty function to search agents that violate any of the constraints at any level [41]. For comparing the results of different algorithms, we have utilized GD and \( \Delta \) metrics.

Table 2: Results of the multi-objective NSDA algorithms on constrained test problems

<table>
<thead>
<tr>
<th>Algorithm Function</th>
<th>PFs</th>
<th>NSDA MEAN±SD</th>
<th>MOSOS MEAN±SD</th>
<th>MOCBO MEAN±SD</th>
<th>MOPSO MEAN±SD</th>
<th>NSGA-II MEAN±SD</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>GD</td>
<td>0.14466±0.00210</td>
<td>0.1508±0.0040</td>
<td>0.1528±0.0051</td>
<td>0.1576±0.0062</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TNK</td>
<td>0.57896±0.05587</td>
<td>0.1206±0.0423</td>
<td>0.1242±0.0512</td>
<td>0.1286±0.0522</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CT</td>
<td>10.7895±0.04748</td>
<td>15.1286±0.063</td>
<td>11.0104±0.052</td>
<td>12.0212±0.054</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GD</td>
<td>0.10054±0.00020</td>
<td>0.1196±0.0031</td>
<td>0.1210±0.0041</td>
<td>0.1282±0.0042</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OSY</td>
<td>0.54789±0.05679</td>
<td>0.5354±0.0616</td>
<td>0.5422±0.0712</td>
<td>0.5931±0.0721</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CT</td>
<td>15.5578±0.02047</td>
<td>20.2124±0.032</td>
<td>12.2104±0.030</td>
<td>14.6420±0.042</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GD</td>
<td>0.14458±0.00375</td>
<td>0.1436±0.0062</td>
<td>0.1498±0.0076</td>
<td>0.1644±0.0078</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BHN</td>
<td>0.44587±0.03789</td>
<td>0.4288±0.0625</td>
<td>0.4798±0.0721</td>
<td>0.4975±0.0632</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CT</td>
<td>07.5254±0.04587</td>
<td>16.2664±0.054</td>
<td>9.1544±0.0420</td>
<td>9.7452±0.0464</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GD</td>
<td>0.05001±0.01478</td>
<td>0.0988±0.0014</td>
<td>0.1018±0.0015</td>
<td>0.1125±0.0026</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SRN</td>
<td>0.20458±0.00090</td>
<td>0.2295±0.0017</td>
<td>0.2352±0.0019</td>
<td>0.2730±0.0023</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CT</td>
<td>7.24456±0.0102</td>
<td>12.3254±0.012</td>
<td>7.3251±0.0082</td>
<td>9.2134±0.0083</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GD</td>
<td>0.32145±0.04002</td>
<td>0.5162±0.0021</td>
<td>0.5202±0.0034</td>
<td>0.5854±0.0036</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CONST</td>
<td>0.7055±0.00076</td>
<td>0.7122±0.0072</td>
<td>0.7235±0.0083</td>
<td>0.7344±0.0084</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CT</td>
<td>16.3556±0.00054</td>
<td>10.0112±0.003</td>
<td>5.2252±0.0028</td>
<td>6.4766±0.0035</td>
</tr>
</tbody>
</table>

Fig. 6: Best Pareto optimal front of KUR, FON, ZDT1, ZDT2, ZDT3, ZDT4, SCHN1 and SCHN2 obtained by the NSDA algorithm.
Table 2 suggests that the NSDA algorithm comparatively performs better than other four algorithms for most of the standard constrained test functions employed. The best Pareto optimal fronts in Fig. 7 also helps in proving since all the Pareto optimal solutions exactly follow the true pareto fronts obtained from by NSDA algorithm.

CONST function consists of concave front with linear front, OSY is similar to CONST but consists of many linear regions with different slopes while TNK almost similar to wave shaped. These also suggests that NSDA algorithm has a capability to solve various type of constraint problem. All the constraint test functions are mathematically given in Appendix B.

c) Results on constrained engineering design problems

The third set of test functions is the most complicated one and consists of four real engineering design problems. Mathematical model of all the four engineering design problem are given in Appendix C. Same as before both GD and diversity matrix is employed to measure the performance of NSDA algorithm with respect to other algorithms to solve them, numeric results are given in Tables and Figure respectively shows the best optimal front obtained by NSDA algorithm.

i. Four-bar truss design problem

The statistical results of four bar truss design problem [42] in given in Table 3 and best optimal front is given in Fig. 8. It consists of two minimization objectives displacement and volume with four design control variable mathematically given in Appendix C.

Table 3: Results of the multi-objective NSDA algorithm on four-bar truss design problem in terms mean and standard deviation

<table>
<thead>
<tr>
<th>Methods</th>
<th>GD MEAN±SD</th>
<th>S MEAN±SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSDA</td>
<td>0.1756±0.0235</td>
<td>1.8717±0.1205</td>
</tr>
<tr>
<td>MOWCA</td>
<td>0.2076±0.0055</td>
<td>2.5816±0.0298</td>
</tr>
<tr>
<td>NSGA-II</td>
<td>0.3601±0.0470</td>
<td>2.3635±0.2551</td>
</tr>
<tr>
<td>MOPSO</td>
<td>0.3741±0.0422</td>
<td>2.5303±0.2275</td>
</tr>
<tr>
<td>μ-GA</td>
<td>0.9102±1.7053</td>
<td>8.2742±16.831</td>
</tr>
<tr>
<td>PAES</td>
<td>0.9733±1.8211</td>
<td>3.2314±5.9555</td>
</tr>
</tbody>
</table>
ii. Speed-reducer design problem

The statistical results of speed reducer design problem[43] is given in Table 4 and best optimal front is given in Fig. 9. It is a well-known mechanical design problem consists of two minimization objectives stress and weight with seven design control variable mathematically given in Appendix C.

Table 4: Results of the multi-objective NSDA algorithm on speed-reducer design problem in terms mean and standard deviation

<table>
<thead>
<tr>
<th>Methods</th>
<th>GD MEAN±SD</th>
<th>S MEAN±SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSDA</td>
<td>0.95578±0.32468780</td>
<td>1.578354±05.947475</td>
</tr>
<tr>
<td>MOWCA</td>
<td>0.98831±0.17894217</td>
<td>16.68520±2.6969443</td>
</tr>
<tr>
<td>NSGA-II</td>
<td>9.843702±7.0810303</td>
<td>02.7654494±3.534978</td>
</tr>
<tr>
<td>μ-GA</td>
<td>3.117536±1.6781086</td>
<td>47.80098±32.8015157</td>
</tr>
<tr>
<td>PAES</td>
<td>77.99834±4.2102608</td>
<td>16.20129±4.26842769</td>
</tr>
</tbody>
</table>

Fig. 8: Pareto optimal front obtained by the NSDA Algorithm for “Four–bus truss design problem”

Fig. 9: Pareto optimal front obtained by the NSDA Algorithm for “Speed Reducer design problem”
iii. **Welded-beam design problem**

The statistical results of welded beam design problem [44] is given in Table 5 and best optimal front is given in Fig. 10. It is a well-known mechanical design problem consists of two minimization objectives fabrication cost and deflection of beam with four design control variable mathematically given in Appendix C.

**Table 5:** Results of the multi-objective NSDA algorithms on welded-beam design problem in terms mean and standard deviation

<table>
<thead>
<tr>
<th>Methods</th>
<th>GD MEAN±SD</th>
<th>Δ MEAN±SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSDA</td>
<td>0.03325±0.01693</td>
<td>0.75844±0.03770</td>
</tr>
<tr>
<td>MOWCA</td>
<td>0.04909±0.02821</td>
<td>0.22478±0.09280</td>
</tr>
<tr>
<td>NSGA-II</td>
<td>0.16875±0.08030</td>
<td>0.88987±0.11976</td>
</tr>
<tr>
<td>pae-ODEMO</td>
<td>0.09169±0.00733</td>
<td>0.58607±0.04366</td>
</tr>
</tbody>
</table>

![Graph](image)

**Fig. 10:** Pareto optimal front obtained by the NSDA Algorithm for “Welded Beam Design problem”

iv. **Disk brake design problem**

The statistical results of welded beam design problem [44] is given in Table 6 and best optimal front is given in Fig. 11. It is a well-known mechanical design problem consists of two minimization objectives stopping time and mass of brake of a disk brake with four design control variable mathematically given in Appendix C.

**Table 6:** Results of the multi-objective NSDA algorithms on the Disk brake design problem in terms mean and standard deviation

<table>
<thead>
<tr>
<th>Methods</th>
<th>GD MEAN±SD</th>
<th>Δ MEAN±SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSDA</td>
<td>0.0587±0.27810</td>
<td>0.43551±0.08237</td>
</tr>
<tr>
<td>pae-ODEMO</td>
<td>2.6928±0.24051</td>
<td>0.84041±0.20085</td>
</tr>
<tr>
<td>NSGA-II</td>
<td>3.0771±0.10782</td>
<td>0.79717±0.06608</td>
</tr>
<tr>
<td>MOWCA</td>
<td>0.0244±0.12314</td>
<td>0.46041±0.10961</td>
</tr>
</tbody>
</table>
Fig. 11: Pareto optimal front obtained by the NSDA Algorithm for “Disk brake design problem”

Due to high complexity of engineering design problem it is really hard to gain results alike true pareto front but we can clearly see that optimal pareto obtained by NSDA algorithm is covers almost whole solutions that are the actual/true solutions of an engineering design problem. From all above tested function we can conclude that problem either it consists of constraints or unconstraint problem NSDA algorithm shows its capability to solve any kind of linear, non-linear and complex real world problem. So in the next section we attached a highly non-linear complex real problem to show its effectiveness regarding the real world complex application with many objectives.

d) Formulation of Economic Constrained Emission Dispatch (ECED) with integration of Wind Power (WP)

In case of wind power generation the output power of wind generator is calculated with the help of a stochastic variable wind speed \( v \) (meter/seconds). Wind speed is a variable function so there probability distribution plays a very important role. Wind speed mathematically formulated as two-parametric Weibull distribution function, probability density function (PDF) and cumulative distribution function (CDF) as follows:

\[
s(v) = \frac{k}{c} \left( \frac{v}{c} \right)^{k-1} \exp\left[-\left(\frac{v}{c}\right)^k\right], \quad v \geq 0
\]

\[
S(v) = 1 - \exp\left[-\left(\frac{v}{c}\right)^k\right], \quad v \geq 0
\]

Where, \( S(v) \) and \( s(v) \) are CDF and PDF respectively. Shape factor and scale factor are \( k \) and \( c \) respectively.

The wind speed and output wind power are related as:

\[
P_{\text{wind}} = \begin{cases} 
0, & v < v_{\text{in}} \text{ or } v \geq v_{\text{out}} \\
p_{\text{rated}} \cdot \frac{v - v_{\text{in}}}{v_{\text{rated}} - v_{\text{in}}}, & v_{\text{in}} \leq v < v_{\text{rated}} \\
p_{\text{rated}} v_{\text{in}}, & 0 \leq P_{\text{wind}} < P_{\text{rated}}
\end{cases}
\]

Where, \( v_{\text{rated}} \) and \( p_{\text{rated}} \) are the rated speed of wind and rated power output. \( v_{\text{out}} \) and \( v_{\text{in}} \) are cut-out and cut-in speed of wind respectively. The CDF of \( P_{\text{wind}} \) in the boundary of \([0, P_{\text{rated}}]\) on an accordance with the speed range of wind can be formulated as:

\[
S(P_{\text{wind}}) = 1 - \exp\left\{-\left(1 + \frac{v_{\text{in}} P_{\text{rated}}}{v_{\text{in}} P_{\text{rated}}} P_{\text{wind}}^{\frac{v_{\text{in}}}{c}}\right)\right\} + \exp\left[-\left(\frac{v_{\text{out}}}{c}\right)^k\right],
\]

Above equation is very meaningful to calculate the ECED problems with speculative wind power with variable speed.
ii. Modeling of ECEDWP problem

As wind power is formulated as system constraint, so the objective function of economic emission dispatch problem (EEDP) stays on unchanged as classical EEDP:

Fuel cost objective is given by:

\[ S(P_i) = \sum_{i}^n (a_i + b_i P_i + c_i P_i^2) \]  

(4.5)

where, the thermal power generators cost coefficients are \(a_i, b_i, c_i\) for \(i\)-th generator. Sum of the total fuel cost of the system and \(N\) is the total number of generators.

Total Emission is calculated by:

\[ E(P_i) = \sum_{i}^n [(a_i + \beta_i P_i + \gamma_i P_i^2) \cdot 10^{-2}] + \delta_i \cdot \exp (\varphi_i \cdot P_i) \]  

(4.6)

where, \(\alpha_i, \beta_i, \gamma_i, \delta_i\) and \(\varphi_i\) are emission coefficients with valve point effect taking into consideration for \(i\)-th thermal generator.

iii. System Constraints

As wind power generation is considered as system constraint with the summation of stochastic variables the classical power balance constraint changes to fulfill the predefined confidence level.

\[ P_i \sum_{i=1}^N (P_i + P_{wind}) \geq P_D + P_{loss} \geq \eta_{pbc} \]  

(4.7)

where, \(\eta_{pbc}\) is confidence level that a power system must follow the load demand and so as it is selected nearer to unity as values lesser than unity represents high operational risk. \(P_{loss}\) represents system losses can be calculated by B-coefficient method given below:

\[ P_{loss} = \sum_{i=1}^N \sum_{j=1}^N (P_{ij} - P_{ij}) \geq \beta_0 \]  

(4.8)

So as to change above described power balance constrained equation into deterministic form can be solved as:

\[ P_i \{P_{wind} < P_D + P_{loss} - \sum_{i=1}^N P_i \} = F(P_D + P_{loss} - \sum_{i=1}^N P_i) \leq 1 - \eta_{pbc} \]  

(4.9)

Assume that the wind turbine have same speed and same direction and combination of Eqs. (4) and (9), the power balance constraint is represented as:

\[ P_D + P_{loss} - \sum_{i=1}^N P_i \leq \frac{e^{\frac{c_{rated} - v_{in}}{v_{rated-v_{in}}}}}{\ln [\eta_{pbc} + \exp (\frac{v_{in}^{\text{rated}}}{v_{in}^{\text{rated-v_{in}}}})]} \]  

(4.10)

iv. Reserve capacity system constraint

So as to reduce the impact of stochastic wind power on system, up and down spinning reserve needs to be maintained [22]. Such reserve constraints formulated as [15] and [16] respectively:

\[ P_i \{ \sum_{i=1}^N (P_i^{\text{max}} - P_i) \geq P_{dr} + t_u * P_{wind} \} \geq \eta_{urc} \]  

(4.11)

\[ P_i \{ \sum_{i=1}^N (P_i - P_i^{\text{min}}) \geq t_d * (P_rated - P_{wind}) \} \geq \eta_{drc} \]  

(4.12)

where, \(P_{dr}\) represents the reserve demand of conventional thermal power plant system and it generally keeps the maximum value of thermal unit, \(P_i^{\text{max}}\) and \(P_i^{\text{min}}\) are maximum and minimum output level of operational generators of \(i\)-th unit, \(\eta_{drc}\) and \(\eta_{urc}\) are predefined down and upper confidence level parameter respectively, \(t_u\) and \(t_d\) are the demand coefficients of up and down spinning reserves.

v. Generational capacity constraint

The real output power is bounded by each generators upper and lower bounds given as:

\[ P_{\text{Minimum}} \leq P_i \leq P_{\text{Maximum}} \]  

(4.13)

V. 40-Operational Thermal Generating Unit

a) Case study I- 40 thermal-generator lossless system without wind power

In this case forty operational generating unit is consider without integration of wind power means all the generating units are coal fired. Input parameters like generators operating limit, fuel cost coefficients and emission coefficients are given in Appendix D extracted from [45]. System is considered lossless and its solution is compared with three well known multi-objective algorithms like SMODE [45], NSGA-II [45] and MBFA [46] in terms of various objectives such as best cost, best emission and best compromise between both objectives. Best compromise solution is then obtained
by the fuzzy based method [47]. Total power demand for this system is 10500 MW. Results obtained by NSDA algorithm is added to table 7 and best pareto front obtained by NSDA algorithm is represented in Fig. 12.

Table 7: Results of the multi-objective NSDA algorithms for case study I- 40 thermal-generator lossless system without wind power

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Best emission</td>
<td>Best Cost</td>
<td>Best compromise</td>
<td>Best emission</td>
</tr>
<tr>
<td>Cost ($/h)</td>
<td>156,700</td>
<td>124,230</td>
<td>126,180</td>
<td>128,490</td>
</tr>
<tr>
<td>Emission (tons/h)</td>
<td>66,799</td>
<td>96,578</td>
<td>99,671</td>
<td>93,002</td>
</tr>
</tbody>
</table>

![Graph](image)

**Fig. 12:** Pareto optimal front obtained by the NSDA Algorithm for “40 thermal-generator lossless system without wind power”

b) Case study II- 40 thermal-generator lossless system with wind power

All the conditions are remaining same as case study I like input parameters and power demand. While integrating with wind power plant, the total rated output power of wind farm is set to 1000 MW [45, 47]. Statistical results obtained by NSDA algorithm is reported in Table 8 and best optimal front is represented in Fig. 13.

Table 8: Results of the multi-objective NSDA algorithms for case study II- 40 thermal-generator lossless system with wind power

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Best emission</td>
<td>Best Cost</td>
<td>Best Compromise Point</td>
<td>Best emission</td>
</tr>
<tr>
<td>$\sum P_G$</td>
<td>10,241.72</td>
<td>10,242.09</td>
<td>10,241.63</td>
<td>10,243.6</td>
</tr>
<tr>
<td>$P_w$</td>
<td>254.24</td>
<td>257.91</td>
<td>258.37</td>
<td>255.68</td>
</tr>
<tr>
<td>Cost</td>
<td>153,830</td>
<td>124,830</td>
<td>124,830</td>
<td>154,000</td>
</tr>
<tr>
<td>Emission</td>
<td>54,855</td>
<td>68,855</td>
<td>68,855</td>
<td>78,890</td>
</tr>
</tbody>
</table>

© 2020 Global Journals
VI. Result Discussion

In almost all the cases that we consider in this article where NSDA algorithm proved its effectiveness in both prospective quantitative and qualitative. From plots also evident that NSDA algorithm follows the exact pareto front similar to the true pareto front for all constrained, unconstrained and complex engineering design problem. So as for real world application of economic emission dispatch problem and its integration with stochastic wind power generation. So for this application Wilcoxon test (statistical test) is performed. In Table 9 the signed rank test is presented in third row of each results whereas the calculation time is represented in forth row. For this test null hypothesis cannot be rejected at 5% level for numeric value ‘0’ while null hypothesis is rejected at 5% level with the value of ‘1’. Where NSDA algorithm performs superior to other algorithms that are considered for comparative purpose. NSDA algorithm shows good performance in both coverage and convergence as main mechanism that guarantee convergence in DA and NSDA algorithms are continuously shrink its virtual limitation using Levy strategy in the movement of dragonflies for their random walk. Both mechanism emphasizes convergence and exploitation proportional to maximum number of

Table 9: Results of Wilcoxon test and simulation/computational time or speed

<table>
<thead>
<tr>
<th>Case Study</th>
<th>Cost</th>
<th>NSDA</th>
<th>NSGAII [45]</th>
<th>Case Study</th>
<th>Cost</th>
<th>NSDA</th>
<th>NSGAII [45]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Study I</td>
<td>Best</td>
<td>119310</td>
<td>124,380</td>
<td>Best</td>
<td>118,689</td>
<td>122,610</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Worst</td>
<td>127,568</td>
<td>147,760</td>
<td>Worst</td>
<td>146,685</td>
<td>173,060</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Mean</td>
<td>124,830</td>
<td>131,710</td>
<td>Mean</td>
<td>123,010</td>
<td>134,880</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Wilcoxon test (H/P)</td>
<td>1/5.40e-10</td>
<td>1/5.77e-10</td>
<td>Wilcoxon test (H/P)</td>
<td>1/5.77e-10</td>
<td>0.9785</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Simulation speed (s)</td>
<td>11.89</td>
<td>11.89</td>
<td>Simulation speed (s)</td>
<td>11.89</td>
<td>11.89</td>
<td></td>
</tr>
</tbody>
</table>

Case Study II

<table>
<thead>
<tr>
<th>Cost</th>
<th>NSDA</th>
<th>NSGAII [45]</th>
<th>Emission</th>
<th>NSDA</th>
<th>NSGAII [45]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best</td>
<td>93,002</td>
<td>123,010</td>
<td>Best</td>
<td>56,509</td>
<td>73,894</td>
</tr>
<tr>
<td>Worst</td>
<td>194,830</td>
<td>123,010</td>
<td>Worst</td>
<td>179,099</td>
<td>158,250</td>
</tr>
<tr>
<td>Mean</td>
<td>141,800</td>
<td>123,010</td>
<td>Mean</td>
<td>104,258</td>
<td>102,120</td>
</tr>
<tr>
<td>Wilcoxon test (H/P)</td>
<td>1/5.55e-10</td>
<td>1/5.55e-10</td>
<td>Wilcoxon test (H/P)</td>
<td>1/5.55e-10</td>
<td>1/5.55e-10</td>
</tr>
<tr>
<td>Simulation speed (s)</td>
<td>154.78</td>
<td>154.78</td>
<td>Simulation speed (s)</td>
<td>127.57</td>
<td>127.57</td>
</tr>
</tbody>
</table>
VII. Conclusion

In this paper the non-dominated sorting dragonfly algorithm-multi-objective version of recently proposed dragonfly algorithm (DA) is proposed known as NSDA algorithm. This paper also utilizes the static and dynamic swarming strategy for exploration purpose used in its parent DA version. NSDA algorithm is developed with equipping dragonfly algorithm with crowding distance criterion, an archive and dragonflies position (accordance to ranking) selection method based on Pareto optimal dominance nature. The NSDA algorithm is first applied on 17 standard test functions (including eight unconstraint, five constraint and four engineering design problem) to prove its capability in terms of qualities and quantities showing numerical as well as convergence and coverage of pareto optimal front with respect to true pareto front. Then after NSDA algorithm is applied to real world complex ECEDWP problem where algorithm proves its dominance over algorithm is first applied on 17 standard test functions. The qualitative results are stored and represented in numeric results expresses that NSDA algorithm has a advantage of high coverage, which is the result of the selection of position of dragonflies and archive selection procedure. All the position are updated according to their fitness value that enable the algorithm to direct the search space in right direction to find the best solution without trapped in local solution. Archive selection criteria follow all the rules of the entrance and exhaust of any value in it for each iteration and updated when its size full. Solutions of higher fitness in archive have higher probability to thrown away first to improve the coverage of the pareto optimal front obtained during the optimization process.

References


**Appendix A:** Unconstrained multi-objective test problems utilized in this work

**KUR:**

Minimize:

\[ f_1(x) = \sum_{i=1}^{2} -10 \exp \left( -0.2 \sqrt{x_i^2 + x_{i+1}^2} \right) \]

\[ f_2(x) = \sum_{i=1}^{2} \left[ |x_i|^{0.8} + 5 \sin(x_i^3) \right] \]

\[-5 \leq x_i \leq 5 \quad 1 \leq i \leq 3\]

**FON:**

\[
\begin{align*}
\text{minimize} & \quad f_1(x) = 1 - \exp \left[ -\sum_{i=1}^{n} \left( x_i - \frac{1}{\sqrt{n}} \right)^2 \right] \\
& \quad f_2(x) = 1 - \exp \left[ -\sum_{i=1}^{n} \left( x_i + \frac{1}{\sqrt{n}} \right)^2 \right] \\
& \quad \quad -4 \leq x_i \leq 4 \quad 1 \leq i \leq n
\end{align*}
\]

**ZDT1:**

Minimise:

\[ f_1(x) = x_1 \]

Minimise:

\[ f_2(x) = g(x) \times h(f_1(x), g(x)) \]

Where:

\[
G(x) = 1 + \frac{9}{N-1} \sum_{i=2}^{N} x_i h(f_1(x), g(x)) = 1 - \sqrt{\frac{f_1(x)}{g(x)}}
\]

\[ 0 \leq x_i \leq 1, 1 \leq i \leq 30 \]
ZDT2:
Minimise: \[ f_1(x) = x_1 \]
Minimise: \[ f_2(x) = g(x) \times h(f_1(x), g(x)) \]
Where: \[ G(x) = 1 + \frac{g(x)}{N-1} \sum_{i=1}^{N} h(f_1(x), g(x)) = 1 - \left( \frac{f_1(x)}{g(x)} \right)^2 \]
\[ 0 \leq x_i \leq 1, 1 \leq i \leq 30 \]

ZDT3:
Minimise: \[ f_1(x) = x_i \]
Minimise: \[ f_2(x) = g(x) \times h(f_1(x), g(x)) \]
Where: \[ G(x) = 1 + \frac{1}{29} \sum_{i=2}^{N} h(f_1(x), g(x)) = 1 - \frac{f_1(x)}{g(x)} - \left( \frac{f_1(x)}{g(x)} \right) \sin(10\pi f_1(x)) \]
\[ 0 \leq x_i \leq 1, 1 \leq i \leq 30 \]

ZDT4:
Minimise: \[ f_1(x) = x_i \]
Minimise: \[ f_2(x) = g(x) \times h(f_1(x), g(x)) \]
\[ h(f_1(x), g(x)) = 1 - \left( \frac{f_1(x)}{g(x)} \right) g(x) = 91 + \sum_{i=2}^{10} \left( x_i^2 - 10 \times \cos(4\pi x_i) \right) \]

SCHN-1:
Minimise: \[ f_1(x) = x_i^2 \]
Minimise: \[ f_2(x) = (x - 2)^2 \] Where, value of can be from 10 to 10^5.
\[ -A \leq x \leq A \]

SCHN-2:
Minimise: \[ f_1(x) = \begin{cases} 
-x, & \text{if } x \leq 1 \\
-x^2 + 2x, & \text{if } 1 < x \leq 3 \\
4 - x, & \text{if } 3 < x \leq 4 \\
-x, & \text{if } x > 4 
\end{cases} \]
Minimise: \[ f_2(x) = (x - 5)^2 \]
\[ -5 \leq x \leq 10 \]

Appendix B: Constrained multi-objective test problems utilised in this work

TNK:
Minimise: \[ f_1(x) = x_1 \]
Minimise: \[ f_2(x) = x_2 \]
Where:
\[ g_1(x) = -x_1^2 - x_2^2 + 1 + 0.1 \cos(16 \arctan \left( \frac{x_1}{x_2} \right)) \]
\[ g_2(x) = 0.5 - (x_1 - 0.5)^2 - (x_2 - 0.5)^2 \]
\[ 0.1 \leq x_i \leq \pi, 0 \leq x_i \leq \pi \]
BNH:
This problem was first proposed by Binh and Korn [48]:
Minimise:
\[ f_1(x) = 4x_1^2 + 4x_2^2 \]
Minimise:
\[ f_2(x) = (x_1 - 5)^2 + (x_2 - 5)^2 \]
Where:
\[ g_1(x) = (x_1 - 5)^2 + x_2^2 - 25 \]
\[ g_2(x) = 7.7 - (x_1 - 8)^2 - (x_2 + 3)^2 \]
\[ 0 \leq x_1 \leq 5, 0 \leq x_2 \leq 3 \]

OSY:
The OSY test problem has five separated regions proposed by Osyczka and Kundu [49]. Also, there are six constraints and six design variables.
Minimise:
\[ f_1(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 + x_6^2 \]
Minimise:
\[ f_2(x) = -[25(x_1^2 - 2)^2 + (x_2 - 1)^2 + (x_3 - 1)^2 + (x_4 - 4)^2 + (x_5 - 1)^2] \]
Where:
\[ g_1(x) = 2 - x_1 - x_2 \]
\[ g_2(x) = -6 + x_1 + x_2 \]
\[ g_3(x) = -2 - x_1 + x_2 \]
\[ g_4(x) = -2 + x_1 - 3x_2 \]
\[ g_5(x) = -4 + x_4 + (x_3 - 3)^2 \]
\[ g_6(x) = 4 - x_6 - (x_5 - 3)^2 \]
\[ 0 \leq x_1 \leq 10, 0 \leq x_2 \leq 10, 1 \leq x_3 \leq 5, 0 \leq x_4 \leq 6, 1 \leq x_5 \leq 5, 0 \leq x_6 \leq 10 \]

SRN:
The third problem has a continuous Pareto optimal front proposed by Srinivas and Deb [50].
Minimise:
\[ f_1(x) = 2 + (x_1 - 2)^2 + (x_2 - 1)^2 \]
Minimise:
\[ f_2(x) = 9x_1 - (x_2 - 1)^2 \]
Where:
\[ g_1(x) = x_1 + x_2^2 - 255 \]
\[ g_2(x) = x_1 - 3x_2 + 10 \]
\[ -20 \leq x_1 \leq 20, -20 \leq x_2 \leq 20 \]

CONSTR:
This problem has a convex Pareto front, and there are two constraints and two design variables.
Minimise:
\[ f_1(x) = x_1 \]
Minimise:
\[ f_2(x) = (1 + x_2)/(x_1) \]
Where:
\[ g_1(x) = 6 - (x_2 + 9x_1), g_2(x) = 1 + x_2 - 9x_1 \]
\[ 0.1 \leq x_1 \leq 1, 0 \leq x_2 \leq 5 \]
Appendix C: Constrained multi-objective engineering problems used in this work

Four-bar truss design problem:

The 4-bar truss design problem is a well-known problem in the structural optimisation field [42], in which structural volume ($f_1$) and displacement ($f_2$) of a 4-bar truss should be minimized. As can be seen in the following equations, there are four design variables ($x_1$-$x_4$) related to cross sectional area of members 1, 2, 3, and 4.

Minimise: 

$$f_1(x) = 200 \cdot (2 \cdot x(1) + \sqrt{2 \cdot x(2)} + \sqrt{x(3)}) + x(4)$$

Minimise: 

$$f_2(x) = 0.01 \cdot \left( \left( \frac{2}{x(1)} \right) + \frac{2 \cdot \sqrt{x(2)}}{x(2)} \right) - \left( (2 \cdot \sqrt{x(2)})/x(3) + (2/x(1)) \right)$$

$$1 \leq x_1 \leq 3.14142 \leq x_2 \leq 3.14142 \leq x_3 \leq 3.1 \leq x_4 \leq 3$$

Speed reducer design problem:

The speed reducer design problem is a well-known problem in the area of mechanical engineering [43], in which the weight ($f_1$) and stress ($f_2$) of a speed reducer should be minimized. There are seven design variables: gear face width ($x_1$), teeth module ($x_2$), number of teeth of pinion ($x_3$ integer variable), distance between bearings 1 ($x_4$), distance between bearings 2 ($x_5$), diameter of shaft 1 ($x_6$), and diameter of shaft 2 ($x_7$) as well as eleven constraints.

Minimise: 

$$f_1(x) = 0.7854 \cdot x(1) \cdot (x(2))^2 \cdot (3.3333 \cdot x(3)^2 + 14.9334 \cdot x(3) - 43.0934) - 1.508 \cdot x(1) \cdot (x(6)^2 + x(7)^2) + 7.4777 \cdot (x(6)^3 + x(7)^3) + 0.7854 \cdot (x(4) \cdot x(6)^2 + x(5) \cdot x(7)^2)$$

Minimise: 

$$f_2(x) = \left( \frac{((745 \cdot x(4))/(x(2) \cdot x(3))^2 + 16.96)}{(0.1 \cdots (6)^3)} \right)$$

Where:

$$g_1(x) = \frac{27}{(x(1) \cdot (x(2)^2 \cdot x(3) - 1)}$$

$$g_2(x) = \frac{397.5}{(x(1) \cdot x(2)^2 \cdot x(3)^2 - 1)}$$

$$g_3(x) = \frac{1.93 \cdot x(4)^3}{(x(2) \cdot x(3) \cdot x(6)^4 - 1)}$$

$$g_4(x) = (1.93 \cdot x(5)^3)/(x(2) \cdot x(3) \cdot x(7)^4 - 1)$$

$$g_5(x) = ((\sqrt{((745 \cdot x(4))/(x(2) \cdot x(3))^2 + 16.96))/((110 \cdot x(6)^3)) - 1$$

$$g_6(x) = ((\sqrt{((745 \cdot x(5))/(x(2) \cdot x(3))^2 + 157.5e6})/(85 \cdot x(7)^3)) - 1$$

$$g_7(x) = ((x(2) \cdot x(3))/40 - 1)$$

$$g_8(x) = (5 \cdot x(2)/x(1)) - 1$$

$$g_9(x) = (x(1)/12 \cdot x(2)) - 1$$

$$g_{10}(x) = ((1.5 \cdot x(6) + 1.9)/x(4)) - 1$$

$$g_{11}(x) = ((1.1 \cdot x(7) + 1.9)/x(5)) - 1$$

$$2.6 \leq x_1 \leq 3.60.7 \leq x_2 \leq 0.817 \leq x_3 \leq 28,7.3 \leq x_4 \leq 8.3,7.3 \leq x_5 \leq 8.3,2.9 \leq x_6 \leq 3.9$$

$$5 \leq x_7 \leq 5.5$$

Welded beam design problem:

The welded beam design problem has four constraints first proposed by Ray and Liew [44]. The fabrication cost ($f_1$) and deflection of the beam ($f_2$) of a welded beam should be minimized in this problem. There are four design variables: the thickness of the weld ($x_1$), the length of the clamped bar ($x_2$), the height of the bar ($x_3$) and the thickness of the bar ($x_4$).

Minimise: 

$$f_1(x) = 1.10471 \cdot x(1)^2 \cdot x(2) + 0.04811 \cdot x(3) \cdot x(4) \cdot (14.0 + x(2))$$

Minimise: 

$$f_2(x) = 65856000/(30 \cdot 10^6 \cdot x(4) \cdot x(3)^3)$$

Where:

$$g_1(x) = tau - 13600$$

$$g_2(x) = sigma - 30000$$

$$g_3(x) = (x(1) - x(4))$$
Displacement of a disk brake design problem:
The disk brake design problem has been proposed by Ray and Liew [44]. The
objectives to be minimized are: stopping time ($f_1$) and mass of a brake ($f_2$) of a disk brake. As can be seen in
following equations, there are four design variables: the inner radius of the disk ($x_1$), the outer radius of the disk ($x_2$),
the engaging force ($x_3$), and the number of friction surfaces ($x_4$) as well as five constraints.

Minimise: $f_1(x) = 4.9 * (10^{-5}) * ((x(2)^2 - x(1)^2) * (x(4) - 1))$

Minimise: $f_2(x) = (9.82 * (10^6)) * ((x(2)^2 - x(1)^2)) / ((x(2)^3 - x(1)^3) * ... * x(4) * x(3))$

Where:

$g_1(x) = 20 + x(1) - x(2)$

$g_2(x) = 2.5 * (x(4) + 1) - 30$

$g_3(x) = (x(3)) / (3.14 * (x(2)^2 - x(1)^2)^2) - 0.4$

$g_4(x) = (2.22 * 10^{-3} * x(3) * (x(2)^3 - x(1)^3)) / ((x(2)^2 - x(1)^2)^2) - 1$

$g_5(x) = 900 - (2.66 * 10^2 * x(3) * x(4) * (x(2)^3 - x(1)^3)) / ((x(2)^2 - x(1)^2))$

$55 \leq x_1 \leq 80, 75 \leq x_2 \leq 110, 1000 \leq x_3 \leq 3000, 2 \leq x_4 \leq 20$
Abstract- The output fluctuation of wind power system has brought huge hidden dangers to the grid. In recent years, the application of energy storage devices to stabilize the fluctuation has been greatly developed. In this paper, a control strategy for wind power fluctuation based on hybrid energy storage of battery and super-capacitor is proposed. Due to the performance characteristics of battery and super-capacitor, a low-pass filter is designed to separate the output into low frequency for battery, and high frequency for super-capacitor. A voltage and current double closed-loop coordination controller is further designed to realize the frequency division mixed energy throughput of the battery and the super-capacitor. The simulation results show that the proposed hybrid energy storage system effectively suppresses the power fluctuation of wind power system and prolongs the service life of the battery.

Keywords: wind power generation, hybrid energy storage, power fluctuation, smooth control.

I. INTRODUCTION

In order to alleviate energy crisis and improve ecological environment, the development and utilization of new energy has been worldwide concerned, among which the wind power generation technology has been rapidly developed. However, wind energy, as a natural clean energy, has great volatility and randomness under the influence of weather. Large-scale wind power grid connection has a certain impact on the safe and stable operation of the power system. At present, matching corresponding energy storage devices is usually adopted in the wind power generation system to effectively smooth the power fluctuation of wind energy.[1-3].

At present, the energy storage system based on battery and super-capacitor is mainly used to smooth the wind power fluctuation. Literature[4] proposed an energy storage structure of dual battery pack that separated charging and discharging processes, and designed a control strategy for power fluctuation to keep the battery running within the optimum discharge depth, thus prolonging the service life of the battery. Sun G W[5] utilized storage battery to suppress the power fluctuation of the wind farm, and realized real-time system adjustment by studying the space vector modulation algorithm of PWM converter. However, a single energy storage device cannot fully meet the comprehensive performance requirements of the system, and the combination of super capacitors and battery can improve the power regulation capacity of the energy storage system[6]. In literature[7], super capacitor voltage low-frequency suppression method is adopted to distribute the smoothing power required by super-capacitor and battery respectively. The battery set is divided into three independent units to alleviate the current imbalance and reduce the loop current ripple. In literature[8], the sliding average filtering algorithm is adopted to separate the power required by the flat suppression of the battery, which effectively reduces The Times of charging and discharging of the battery and improves the operation economy of the energy storage system. Literature[9] proposed an energy storage technology based on wavelet packet decomposition to smooth power fluctuations. Power fluctuation signals are decomposed at multiple scales by wavelet packet decomposition theory. Low-frequency fluctuations are directly connected to the grid, while high-frequency fluctuations are further decomposed to different energy storage devices through wavelet packet decomposition for smoothing.

In this paper, for the combined wind storage system, a control strategy based on hybrid energy storage to smooth out wind power fluctuations is studied. Through a low-pass filter the fluctuation of power is separated into high frequency and low frequency, complying with the super capacitor and battery characteristics respectively, to enhance the control capacity and the service life of the battery energy storage system, proposing a voltage and current double closed-loop coordination controller where two kinds of energy storage devices share the voltage outer loop. Finally the validity of the proposed control strategy is validated by computer simulation.

II. STRUCTURE OF THE WIND STORAGE SYSTEM

The energy storage system can cut the peak load, fill the valley load and reduce the power fluctuation when wind power is connected to the power system, which is conducive to large-scale access of wind power, improving the stability of the grid, and carrying out planned dispatching of wind power generation[10]. For
the current mainstream doubly fed induction generator, adopting a centralized hybrid energy storage system which is directly connected to the ac bus in parallel. Its structure is shown in figure 1.

![Fig. 1: Structure of wind turbine-energy storage system](image)

The energy storage devices are all connected to the dc bus of the energy storage converter through bi-directional DC-DC Converter, as shown in figure 2. The circuit has a Boost state and a Buck state. When the wind energy is insufficient, the energy storage device is required to provide energy. Energy flows from the energy storage device to the dc bus. When there is surplus of wind energy, the energy storage device is required to absorb energy. Energy flows from the dc bus to the energy storage device. Meanwhile T1 tube is turned on and T2 tube is turned off, the converter works in Buck state, and the energy storage device is charged.

![Fig. 2: Bidirectional DC-DC convertor](image)

The energy storage converter is connected between the dc bus and the common ac bus, and PQ control is adopted to obtain the active power and reactive power required by the system (where the reference value of reactive power is set as 0). When there is surplus of wind energy, the energy storage system absorbs energy, and the energy storage converter is in the rectifying state. When there is short of wind energy, the energy storage system releases energy, and the energy storage converter is in the state of inverter. Its circuit structure diagram is shown in figure 3.

![Fig. 3: Equivalent circuit model energy storage convertor](image)

The expressions of active power and reactive power of the energy storage converter are as follows \(^{[11]}\)

\[
\begin{align*}
P &= E_{sd}i_d \\
Q &= -E_{sd}i_q
\end{align*}
\]

(1)

Where \(i_d\) , \(i_q\) is the component of the three-phase current on the ac side in the \(d\) and \(q\) axis, and is the component of the three-phase voltage on the ac side in the \(d\) axis. Active power and reactive power can be
controlled by controlling \( i_d \) and \( i_q \). When the energy storage converter is connected to the grid, the phase and frequency of the ac side can be obtained through the phase-locked loop. Through the energy storage system to specify the throughput active power \( P_{\text{HESS}} \) and reactive power \( Q_{\text{HESS}} \), the reference value of the current loop \( i_{d\text{ref}} \) and \( i_{q\text{ref}} \) can be obtained after calculation. \( u_d \), \( u_q \) can be obtained by comparing and controlling the current loop with \( i_d \), \( i_q \), and \( u_{\text{ref}} \) can be obtained by Park transform in version, and then generating SVPWM wave to control the on and off of the switch tube.

### III. Power Fluctuation Restrain based on Hybrid Energy Storage System

Current specification standards of wind power fluctuations is in accordance with Technical Provisions for Wind Farm Access to Power System issued by State Grid Corporation of China, in February 2009. This document clearly specifies the maximum variation of output power when the wind farm is connected to the grid, including the variation of 1min and 10mins. The specific data are shown in table 1 below.

**Table 1:** The recommendation value of the maximum variation rate of the wind farm

<table>
<thead>
<tr>
<th>Capacity of Wind Farms (MW)</th>
<th>Maximum Change in 1 min (MW)</th>
<th>Maximum Change in 10 mins (MW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;30</td>
<td>6</td>
<td>20</td>
</tr>
<tr>
<td>30-150</td>
<td>capacity/5</td>
<td>capacity/1.5</td>
</tr>
<tr>
<td>&gt;150</td>
<td>30</td>
<td>100</td>
</tr>
</tbody>
</table>

#### a) Energy flow of wind storage system

The main output power of the wind storage system comes from the power generated by the draught fan, so the main factor affecting the wind storage system is wind speed. The power balance relationship of wind storage system is as follows

\[
P_o = P_w - P_{\text{HESS}} \\
P_{\text{HESS}} = P_b + P_c
\]

(2)

Where \( P_{\text{HESS}} \) is the throughput power of the hybrid energy storage system, \( P_b \) and \( P_c \) is the throughput power of the battery and ultra-capacitor respectively, and \( P_o \) is the grid-connected power of the whole system. When the wind speed is relatively high, the energy storage system needs to absorb power to smooth the fluctuating power, while the wind speed is relatively low, the energy storage system emits power to stabilize the power fluctuation.

#### b) Low pass filter for frequency division

The first order low-pass filter is designed to separate the power frequency of the energy storage system into low frequency for battery and high frequency for super-capacitor. The circuit schematic diagram is shown in figure 4, where \( U_1 \) is the input signal, \( U_2 \) is the output signal, \( R \) is the filter resistance, and \( C \) is the filter capacitor.

**Fig. 4:** Circuit diagram of one-order low pass filter

The differential equation of the circuit is shown in equation (3).

\[
RC \frac{dU_2}{dt} + U_2 = U_1
\]

(3)

The transfer function is

\[
H(s) = \frac{1}{1 + \tau s}
\]

(4)

Where \( s \) is the filter operator, \( \tau \) is the filter time constant, \( \tau = 1 / 2\pi f_c \), and \( f_c \) is the filter cut-off frequency.

When the filter is applied to the power distribution of the energy storage system, the input signal \( U_1 \) is the expected power value of the energy storage through put \( P^*_{\text{HESS}} \), and the output signal \( U_2 \) is the reference power value of the battery throughput \( P^*_b \). The expressions are

\[
P^*_b = \frac{1}{1 + \tau s} P^*_{\text{HESS}}
\]

(5)

\[
P^*_c = P^*_{\text{HESS}} - P^*_b = \frac{\tau s}{1 + \tau s} P^*_{\text{HESS}}
\]

(6)

In the expression, \( P^*_c \) is the reference value of the power suppressed by super-capacitor. According to the characteristics of each energy storage device, the battery response time is the key factor. The power with a frequency higher than 0.1Hz and the power with a frequency lower than 0.1Hz is designed to be absorbed by a super-capacitor and a battery respectively, so take \( f_c = 0.1Hz \), \( \tau = 1.6s \).
c) Coordinated control strategy of voltage and current double closed loop

The following voltage and current double closed-loop frequency division coordination control strategy is further proposed, as is shown in fig 5. The super-capacitor and the battery share a voltage outer loop. The current inner loop reference value $I_{ref}$ is obtained by the PI controller, and is subsequently divided into high frequency part and low frequency part, as the reference value of the battery current loop and the reference value of the super-capacitor current loop respectively.

\[
\frac{d}{dt} \begin{bmatrix} I \\ U_{dc} \end{bmatrix} = \begin{bmatrix} 0 & -\frac{1(1-D)}{L} \\ -\frac{1-D}{C} & -\frac{1}{RC} \end{bmatrix} \begin{bmatrix} I \\ U_{dc} \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} U_{bar}
\]

Fig. 5: Voltage and current double closed-loop coordination control strategy

When the circuit is in Boost state, T1 tube is turned off and T2 tube is working. The DC voltage is controlled by adjusting its duty cycle $D$, where $D = \frac{T_{on}}{T}$, and $T_{on}$ is the conduction time of T2 tube in one cycle. Provided that the DC bus voltage is $U_{dc}$, the terminal voltage of the energy storage device is $U_{bar}$, and the current flowing into the energy storage side is $I$, when the T2 tube is working, the state equation can be obtained as follows:

\[
L \frac{dI}{dt} = U_{bar} - (1-D)U_{dc}
\]

Thus, the circuit's current loop control equation and duty cycle adjustment equation under Boost state are:

\[
D = \frac{(K_p + K_i / s)(I_{ref} - I) - U_{bar} + U_{dc}}{U_{dc}}
\]

Where $K_p$ and $K_i$ are the proportional and integral current loop parameters of the PI controller.

Similarly, when the circuit is in Buck state, the current loop control equation and duty cycle expression are:

\[
L \frac{dI}{dt} = -U_{bar} + D*U_{dc}
\]

\[
D = \frac{(K_p + K_i / s)(I_{ref} - I) + U_{bar}}{U_{dc}}
\]

IV. SIMULATION RESULTS

Based on Matlab / Simulink, an integrated wind and energy storage grid-connected system is established. The external system uses a single-machine infinite system. The main parameters of the fan are 100kW fan capacity and 690V rated voltage, and the main parameters of the energy storage system are 300Ah battery capacity, 0.5R battery internal resistance, 70F super-capacitor capacity and 800V DC bus reference voltage.

Assume that the active power output expectation of the system, namely the grid dispatch value, is constant within a second time scale, the reactive power is zero, and the wind speed is variable. The output power of the wind turbine and the power to
the grid stabilized by the energy storage system are shown in Fig 6, which verifies that the power to the grid remains basically unchanged after smoothed by the designed hybrid energy storage stabilization system, meeting the grid dispatching requirements.

![Graph showing power comparison](image)

**Fig. 6:** Comparison of suppressing power under variable wind speed

Figures 7 and 8 compare the actual power and reference power of the battery and super-capacitor, illustrating that the energy storage device can be charged and discharged according to the reference value of power distribution, and the proposed hybrid energy storage system can well suppresses wind power fluctuations.

![Graph showing power comparison](image)

**Fig. 7:** Actual and reference power of battery

**Fig. 8:** Actual and reference power of capacitor

Fig. 9 and 10 show the changes of SOC values of the battery and super-capacitor when the system is connected to the grid. It shows that the smooth battery SOC curve has a small amplitude variation range without repeated charge and discharge. The super-capacitor SOC changes rapidly with repeated charge and discharge for many times, and the charge and discharge depth is also larger than that of the battery. Therefore, the effectiveness of the hybrid energy storage control strategy is verified, which can effectively reduce the charging and discharging times of the battery and prolong its service life.

![Graph showing SOC comparison](image)

**Fig. 9:** SOC curve of battery

**Fig. 10:** SOC curve of super capacitor

V. **Conclusion**

In this paper, a wind power grid-connected system based on hybrid storage of battery and super-capacitor is established, and a power fluctuation smoothing strategy based on voltage and current double closed-loop frequency division coordinated control is proposed. The actual power characteristics and SOC change curve of the battery and super-capacitor are observed in the simulation under the conditions of constant wind speed and variable wind speed. The results verify that the control strategy can effectively smooth the fluctuation of wind power, reduce charging and discharging times of the battery, and prolong its service life.
References Références Referencias


