A Multi-Layer Artificial Neural Network Approach for Runway Configuration Prediction

Md Shohel Ahmed\textsuperscript{1}, Sameer Alam\textsuperscript{2}, and Michael Barlow\textsuperscript{1}

\textsuperscript{1}School of Engineering and Information Technology, University of New South Wales, Canberra, Australia. 
\textsuperscript{2}School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore.

Abstract—Runway configuration selection is a complex decision-making process which involves the interaction of several factors, many of them stochastic in nature. Runway configuration identifies the set of runways that can be used, under certain operating conditions, for a given time constraint. The runway configuration has a major influence on the runway operating efficiency and the overall capacity of the runway system. This paper proposes an adapted Multi-layer Artificial Neural Network (MANN) model to predict the runway configuration and assignment of the appropriate runway to flights that lead to maximum runway capacity. The proposed model is trained and tested on Amsterdam Schiphol airport data with 1789 arrival and departure spread over two days. Several factors, including weather and wind, were identified and incorporated in the training of the MANN. In this research, different classes of ANN technique i.e., feed-forward back-propagation, cascaded feed-forward, distributed time delay back-propagation and recurrent back-propagation has been applied and evaluated for predicting runway configuration. The validation and testing of the model are carried on with a subset of data using cross-validation and Mean Square Error to evaluate runway configuration prediction. The hourly throughput of each configuration is projected from the runway configuration capacity envelope of the Schiphol airport. Results demonstrate the viability and benefits of machine learning approach for predicting runway configuration in an operational environment.

Keywords—Machine learning, neural network, Airport runway configuration, runway capacity.

I. INTRODUCTION

The steady growth of air traffic demand and the limited capacities of the runways at airports has led to airspace congestion and prolonged delays in terminal airspace leading to a significant economic loss to airlines as well as Air Navigation Service Providers(ANSPs). It is estimated that by 2035, more than 20 airports, in Europe alone, will be operating at 80\% or more of capacity on a daily basis, resulting in delays of up to 5-6 minutes (2013 Challenges of Growth Report, Eurocontrol). Several research initiatives have been undertaken under EU’s Single European Sky ATM Research (SESAR) [1] for high-performing airport operations to increase runway and airport throughput under project PJ02 (EARTH) which focuses on developing, validating and delivering separation and procedures to improve runway and airport throughput considering wake vortex, weather, the environment, and noise while taking account of different levels of traffic demand, future aircraft capabilities and airport configurations. Similar initiatives have taken under increase arrivals and departures at high-density airports program at US FAA’s NextGen [2] to maximize runway capacity. Notable amongst them, from runway/terminal airspace capacity enhancement, are traffic optimization on single/multiple runways, minimum-pair separations based on required surveillance performance, enhanced arrival procedures enabled by satellite technologies, wake turbulence separation optimization, enhanced terminal area for efficient curved operations etc.

Major airports around the world have multiple runways and Air Traffic Controllers (ATCs) have to select a subset of these runways as active runways for arrival and departure management. This set of active runways, associated traffic direction and assigned nature of the operation (arrival, departure and mix mode) is known as the runway configuration. The choice of the configurations influences airports capacity to serve the demand for arrivals and departures. The assignment of the runway and the choice of the runway configuration plays a significant role in determining the overall runway capacity. Given the complex interplay of factors including wind, weather, noise abatement, arrival and departure sequence, navigation aids at runway etc., it is challenging to forecast which runway configuration is best to serve the arrival-departure mix such that it can maximize the runway capacity for a given look-ahead time. A better prediction of runway capacity is essential for strategic flight planning [3], smooth air traffic flow management [4], and coordination of arrival and departure flights. For efficient runway operation, accurate and reliable prediction of runway configuration i.e., assignment of appropriate runway to the arrival and departure flight is crucial.

However, there are no precise rules that dictate the selection of active runways. ATC consider many interrelated factors including weather (wind and visibility), predicted arrival and departure demand, environmental considerations such as noise abatement procedures, and coordination of flows with neighboring airports [5] for identifying which runways will be active in a given configuration. The major factors that influence the runway configuration selection include metrological conditions (wind, visibility, temperature, QNH, cloud ceiling), operation time (noise distribution), aircraft class, flight type (IFR, VFR, SVFR etc.) and distribution of the aircraft in a given arrival or departure sequence. Since an ATC selects a runway configuration, in anticipation/prediction to the changes in the above factors, a wrong selection may cause unnecessary delays, either airborne or on the ground, and may contribute to inefficiencies in air traffic management. The availability of a particular configuration might be restricted by the metrological
Fig. 1: Schematic diagram of the runways of Amsterdam airport, Schiphol. Six runways are situated in a parallel, cross, converging and diverging orientation.

As a case study, we have used one-day traffic data for arrivals and departures at Amsterdam Schipol international airport. The airport has 6 active runways. During an inbound peak, Schiphol airport usually operates 2 runways for landings and 1 runway for takeoffs. Similarly, during an outbound peak, Schiphol usually operates 2 runways for takeoffs and 1 for landings. If inbound and outbound times overlap, one can expect a $2 + 2$ configuration, with 2 runways open for landings and 2 for takeoffs. Figure 1 shows the schematic representation of the Amsterdam Schiphol airport runways and their orientation.

The motivation behind this work is to predict the appropriate runway configuration before an hour lookahead time that helps to projects the maximum runway capacity. The method is divided into two main steps: Preprocessing and Machine Learning:

1) The Pre-processing includes data extraction, data redundancy reduction, and normalization. Normalization of data means transforming the data format in such a way that machine learning algorithm can easily and accurately process.

2) The Machine learning includes training, validation, and testing on real-world data-sets.

The paper is organized as follows: Section II presents a background of the runway configuration prediction and runway assignment methods. Section III discusses the runway configuration selection factors. The proposed approach is discussed in section IV. Section V illustrates the data set organization. The machine learning model is presented in section VI. The experimental design is discussed in section VII. Result analysis is presented in section VIII. Conclusions and future work are summarized in Section IX.

### TABLE I: Variables of the runway configuration that influence the selection of runway configuration.

<table>
<thead>
<tr>
<th>$X_i$</th>
<th>Variables of the runway configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>Operation time</td>
</tr>
<tr>
<td>$x_2$</td>
<td>Traffic distribution (heavy aircraft)</td>
</tr>
<tr>
<td>$x_3$</td>
<td>Traffic distribution (medium aircraft)</td>
</tr>
<tr>
<td>$x_4$</td>
<td>Traffic distribution (small aircraft)</td>
</tr>
<tr>
<td>$x_5$</td>
<td>Wind Direction</td>
</tr>
<tr>
<td>$x_6$</td>
<td>Wind Speed</td>
</tr>
<tr>
<td>$x_7$</td>
<td>Visibility</td>
</tr>
<tr>
<td>$x_8$</td>
<td>Cloud Ceiling</td>
</tr>
<tr>
<td>$x_9$</td>
<td>Scattered Cloud</td>
</tr>
<tr>
<td>$x_{10}$</td>
<td>Broken Cloud</td>
</tr>
<tr>
<td>$x_{11}$</td>
<td>Temperature</td>
</tr>
<tr>
<td>$x_{12}$</td>
<td>Dew Point</td>
</tr>
<tr>
<td>$x_{13}$</td>
<td>QNH</td>
</tr>
</tbody>
</table>

**II. BACKGROUND**

The availability of the runways at major airports might be restricted by prevailing meteorological conditions and traffic scenario. To date, the runway configuration selection decision is made by applying previous experience and rules of thumb, and with limited automation assistance [6]. Despite the lack of research on the problem of runway configuration selection in a stochastic operating environment, some noteworthy research on this problem are found. Several of the research have been motivated on the development of tools assuming knowledge of their respective capacities, prevailing operating conditions, expected traffic demand and feasibility of the configuration that suggests the optimal sequencing of runway configurations [7], [8], [9], [10].

In [5], a discrete-choice based model was proposed that uses empirical observations to characterize the configuration selection process. In [11], a decision support tool was proposed to improve the selection of runway configurations and the balancing of arrival and departure service rates. In [12], a dynamic programming model was used to reduce the congestion that dynamically selects runway configurations and balances the arrival and departure rates at a busy airport. A major challenge to the practical implementation of such approaches is the incorporation of the dynamic weather data and objectives of the human operators. For instance, runway configuration changes require significant coordination among the air traffic controllers, pilots, and other ground resources that increase the ATC workload.

Recent research has focused on machine learning based techniques [13]. Most of the approaches attempt to reduce the cost by predicting the trajectories, predicting the airport’s arrival rates or forecasting the traffic demand. The Support Vector Machines (SVM) and Ensemble Bagging Decision Tree (BDT) machine learning method attempt to select the runway configuration and predict the arrival rates [14]. This approach models the problem of the runway configuration selection process using retrieved operational data, quantifying the effect of factors that influence the air traffic controllers.

**III. FACTORS AFFECTS THE RUNWAY CONFIGURATION**

The combinations of the active runways at an airport that are used for arrivals and departures at any time is known as the runway configuration. The combination of the runways for
a configuration is typically selected in the form $A_1, A_2 \mid D_1, D_2$ where $A_1$ and $A_2$ are the arrival runways, and $D_1$ and $D_2$ are the departure runways. A variety of factors are considered to choose the appropriate runway configuration. These factors may impact runway capacity directly or indirectly. For example, runway operation time is an important factor due to noise abatement procedures [15]. Aircraft wake category (e.g., heavy, medium, light), flight type (i.e., arrival/departure) and traffic mix (i.e., distribution of the aircraft) influences the separation time between the consecutive aircraft as well as runway occupancy time which in turn affects the runway capacity significantly [16]. Suitable configuration selection guides the aircraft to plan for the appropriate path towards the terminal airspace that contributes to the runway capacity [17].

Wind speed and wind direction influence the choice of a runway. Over a certain threshold, crosswinds, headwinds, and tailwind may not be favourable for aircraft landing and takeoff [18]. Headwinds decrease the aircraft ground speed. Therefore, for aircraft flying at a given airspeed profile, with given in-trail spacing on final approach, the presence of a headwind increases the inter-arrival time spacing, and thus reduces the runway throughput. However, headwind has a possible advantage too. Headwind helps to lift off the aircraft fast and reduce the take off run.

Therefore, an ATC always chooses the configuration that is favourable for the wind conditions. The cloud ceiling and the visibility at the airport, influence the choice of runway configuration. The runway is not operable at low visibility conditions. Low visibility reduces the capacity of the runway since the separation of the aircraft is increased. In addition to winds, cloud, and visibility, airport capacity is also affected by temperature (i.e., wet or icy runway conditions). These result in longer taxi times (due to longer breaking distance), which in turn causes longer final approach spacing, consequently reducing the runway capacity. Furthermore, airborne icing conditions cause ground delays for departures. In addition, extremely hot temperatures can make take-off impossible because of insufficient lift and extremely cold temperatures slow ramp operations [19].

IV. PROPOSED APPROACH

Figure 2 shows the flowchart of the proposed approach. There are two major parts of the proposed prediction model, the preprocessing (traffic data, weather data), and the machine learning. The data preprocessing includes the extraction of the traffic data, weather data, the reduction of the similar data pattern and the feature scaling. The machine learning includes training, testing and the validation of the data. All the major factors that influence runway configuration and consequently the runway capacity are processed [20]. Table I lists all the variables processed for each sample data.

A. Runway Configuration for Amsterdam Schiphol Airport

Data pre-processing is an essential part of a machine learning algorithm. Data pre-processing ensures that information being readied are accurate and consistent, so the dimension reduction and application of the artificial neural network can be valid. The dataset used for this research is extracted from European airspace air traffic flight data between 30th June 2016 to 1st July 2016. We extract the flight data (arrival and departures) for Amsterdam Schiphol International airport, which is one of the busiest airports in the world. We choose Schiphol airport for this research because it has six runways in different layout and orientation and the weather at the airport changes quite frequently. Based on the traffic data we computed all the runway configurations on July 1, 2016, at Amsterdam, Schiphol airport. We noticed that the runway

**Fig. 2: Proposed MANN machine learning approach.**

**Fig. 3: Runway Configuration Capacity Coverage Chart Amsterdam, Schiphol.** The x-axis shows the percentages of active time of the configuration and y-axis shows the movement per hour.
configuration has been changed 20 times within 24 hours, making it a suitable candidate for the runway configuration prediction problem.

1) Capacity Coverage Chart and Capacity Envelope: The Capacity Coverage Chart (CCC) for the runway configuration is produced from the historical data that is presented in figure 3. In CCC, the capacity of each configuration and the percentage of the activation time are presented. One interesting point to note that most of the configuration remains active less than one hour. For example, 18R; 18C | 18L and 18R | 18L were active 5.5% of the 24 hours time.

We then developed the Runway Capacity Envelope for Amsterdam Schiphol Airport. This envelope defines the boundary of the maximum throughput capacities that can be achieved for a given runway configuration under a range of arrival and departure mix. The runway configuration capacity envelop is presented in figure 4. The hourly capacity for each configuration indicates that various factors (i.e., weather, traffic pattern, time of the day) affect the capacity for each configuration. Therefore, Amsterdam Schiphol Airport becomes an interesting airport to the researcher for studying the runway configuration prediction for runway capacity maximization.

2) Weather Data for Amsterdam Schiphol Airport: Weather observations and forecasts data for Amsterdam Schiphol airport on July 1, 2016, is collected and processed. METAR (Meteorological terminal area routine) and TAF (Terminal area forecasts) data is collected from observational data and an algorithm was developed for automatic decoding and processing of weather information. The following weather information for a 24 hour period was compiled: issue time, wind direction, wind speed, wind gust, wind direction variation, visibility, runway visual range, weather during time of observation, cloud ceiling, air temperature, dew-point, QNH (pressure measured at airport with adjustment made to suit aeronautical use), weather during the past hour but not at time of observation, wind shear information and trend-type landing forecast.

Wind speed and direction are vital factors that impact runway configuration. Figure 5 shows the actual wind speed and direction observed at Amsterdam, Schiphol on June 30 to July 1, 2016. The arrows correspond to the wind direction, frequency, and the speed limit. The colour square boxes represent the runway directions. The observation denotes that majority of the winds components corresponds to the operating conditions in which all the runways are usable. However, the crosswind components for the runway 18R/36L, 18L/36R and 18C/36C are higher comparing to the other runways.

Notice the capacity coverage chart at figure 3, 18R;18C | 18L and 18R | 18L were active around 5.5% of time. On the other hand, runway 24 was active highest percentage of time in different configurations. Because it had lowest crosswind component. The headwind for runway 24 helps to lifts off the aircraft sooner; consequently, this result in a lower ground speed and shorter take-off run for the aircraft to become airborne.

V. DATA SET ORGANIZATION

Suppose that we wish to visualize n observations for a runway configuration with measurements on a set of p features, \( X_1, X_2, ..., X_p \), as part of an exploratory data analysis. So the data matrix for the configuration become \( A = [n \times p] \). Since a runway configuration is activated at the different time of the day. The total number of flight observation for that configuration become high. For example, 18R; 18C | 24; 18L process 464 flight on the day at Schiphol. So the matrix for the configuration considering 13 variables become \([13 \times 464] \) that is high dimensional. We process a total of 40480 flight from

![Runway Configuration Capacity Envelopes (Armsterdam, Schiphol )](image1)

![Wind Speed and direction observed at Amsterdam, Schiphol airport on July 1, 2016. The arrows present the wind speed and direction. The rectangle shape shows the runway direction.](image2)
Europe, where 1789 arrivals and departures are encountered in Amsterdam, Schiphol airport runway.

1) Data Similarity Measurement: In this study, we observe that weather data have a similar pattern for a long time. For example, metrological condition (i.e., wind speed, wind direction, visibility, and cloud ceiling) remain same for couple of hours. The redundant data reduces the efficiency of the machine learning algorithm. Therefore, a data mining technique known as similarity measurement is applied to estimate the data similarity and reduce the redundant data for the machine learning algorithm. Similarity measurement is an effective data mining technique to measure the similitude among the data [21]. Common intervals used to mapping the similarity are [−1, 1] or [0, 1], where 1 denotes the maximum of similarity. Similarity of two two series \( X = x_1, \ldots, x_n, Y = y_1, \ldots, y_n \) can be estimated as:

\[
t_{\text{sim}}(X, Y) = \sqrt{\frac{1}{n} \sum_{i=1}^{n} 1 - \frac{|x_i - y_i|}{|x_i| + |y_i|}} \tag{1}
\]

Figure 6 illustrates the effectiveness of the similarity measurement technique that removes the redundant data from the total data set. The similarity threshold used was 0.99; it means that only the most similar data are reduced. The similarity measurement technique reduces dataset from 1789 to 507. An interesting point to notice that 507 unique observation representing the 1789 observation without affecting the machine learning model prediction accuracy.

2) Feature Scaling: Machine learning algorithms e.g., Neural Networks (NN) natively process numeric data. Therefore, processed data need to be scaled in a standard format. The data normalization is a useful process for feature scaling. When numeric data values are normalized, neural network training is often more efficient, which leads to a better predictor. However, if the numeric data is not normalized, and the magnitudes of two predictors might be far apart, then a change in the value of a neural network weight has a far more relative influence on the data value with larger magnitudes. Normalisation scales all variables in the range [0, 1] and removes variation in data that might be non-causal in predicting the output. For the variable \( X=(X_1, \ldots, X_n) \), normalized data \( Z_i \) of the \( i_{th} \) element is estimated as

\[
Z_i = \frac{X_i - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}} \tag{2}
\]

where \( X_{\text{min}} \) and \( X_{\text{max}} \) are the minimum and maximum value of the variable X respectively.

VI. PROPOSED MACHINE LEARNING MODEL

In order to achieve an unbiased performance of the machine learning algorithm, a multi-layer artificial neural network (MANN) is used [22]. Multilayer architecture gives neural networks the potential of being global approximators. In this experiments, a specific class of neural networks is used, referred to as feed-forward networks i.e., Multi-Layer Perceptrons (MLP). The units (neurons) are arranged in a way in such networks so that all units in successive layers are fully connected. MLP has one input layer, one or several hidden layers, and an output layer. In our case, we have only one target value to predict, so the output layer has only one unit. The network is referred as MANN.

Figure 7 illustrates the MANN architecture. The proposed MANN is a supervised learning technique which has three layers i.e., an input layer, hidden layers, and output layers. The hidden layer will naturally be used to undertake more complicated tasks than perceptrons. A supervised learning algorithm analyzes the training data (i.e., input feature data and the corresponding target output) and produces an inferred
function, which can be used for mapping new inputs. The advantage of a multi-layer neural network is that each prediction cell is able to learn the hidden and non-linear dependencies from the training data. In this research, different classes of neural network technique i.e., feed-forward backpropagation, cascaded feed-forward, distributed time delay backpropagation and recurrent backpropagation has been applied.

Backpropagation is a method used to calculate the error contribution of each neuron after a batch of data is processed. In backpropagation, the parameters of primary interest are $w_{kj}^{l}$, the weight between node $j$ in layer $l_k$ and node $i$ in layer $l_{k-1}$, and $b_i^k$, the bias for node $i$ in layer $l_k$. Given input $\{x_i|i \in [1,n]\}$ and the hidden layer node number $m$, the network output can be predicted as,

$$y = \sum_j m w_j \varphi(\sum_i n w_{ji} x_i + b_{j-h}) + b_{h-o} \quad (3)$$

where $w_{ji}$ is the weight between the $i^{th}$ input and the $j^{th}$ hidden node, $w_{j}$ is the weight between the $j^{th}$ hidden node and the output node, $b_{j-h}$ is the bias to the $j^{th}$ hidden layer, $b_{h-o}$ is the bias to the output layer.

1) Backpropagation Technique for Learning: The backpropagation algorithm uses the gradient descent method that involves calculating the derivative of the squared error function with respect to the weights of the network. The error function is described in the following subsection.

For each neuron $j$, its output $u_j$ is defined as

$$u_j = \varphi \left( \sum_k w_{kj} x_k \right) \quad (4)$$

$w_{kj}$ denotes the weight between neurons $k$ and $j$ and $x_k$ is input of the $k^{th}$ neuron. The activation function $\varphi$ is non-linear and differentiable. The activation function calculates a layer’s output from its net input. In this study, the tan-sigmoid function is used as the activation function. This function is a good tradeoff for neural networks, where speed is important [23].

$$\varphi(z) = \text{tansig}(z) = \frac{2}{1 + e^{-2z}} - 1 \quad (5)$$

where $z$ is the summation of the weighted input values to the processing node. The processing nodes constitute a set of fully interconnected layers, except that there are no interconnections between nodes within the same layer i.e., the standard feed-forward backpropagation algorithm.

2) Performance Measurement (Error Function): Typically, the training process requires an error function that quantifies the difference between the predicted output of the machine learning algorithm and the true value $y$ for an input $\vec{x}$ over a set of the input-output pair $(\vec{x}, y)$. Here, we use Mean Square Error (MSE) to evaluate our runway configuration prediction model performance. MSE has two desirable advantages. The first is to ensure that all values are positive. If one or more values are negative, the sum of all the values could be unrealistically small and a poor representation of the actual variation between predicted and observed values. The second advantage of squaring is to give more weight to larger differences, which ensures that a large value for MSE signifies large data variations. Smaller values for MSE indicate closer agreement between predicted and observed results i.e., better accuracy of prediction, and an MSE of 0.0 indicates perfect prediction. The MSE function is defined for the $N$ input-output pairs $X = (\vec{x}_1, y_1), \ldots, (\vec{x}_N, y_N)$ as

$$E(x_i) = |f_i - y_i| \quad (6)$$

So, the mean square error can be estimated as

$$MSE = \frac{1}{N} \sum_i N E^2(i) \quad (7)$$

where $N$ is the number of data, $f_i$ the predicted value returned by the model and $y_i$ the actual target value of the data point $i$.

VII. EXPERIMENTAL DESIGN

Selecting an appropriate architecture is, in general, the first step to take when designing a machine learning based prediction system. In this study, we have simulated different neural network structure for the pre-processed data. The model uses four networks architecture i.e., feed-forward backpropagation, feed-forward distributed time delay, recurrent backpropagation, and the cascade forward backpropagation.

During the training of the networks, the data is divided into three subsets. The first subset is the training set (70% of the total data), which is used for computing the gradient and updating the network weights and biases. The second subset is the validation set (15% of the total data). The error on the validation set is monitored during the training process. The network weights and biases are saved at the minimum of the validation set error. The third data set is test set (15% of the total data). The sharing of data was made randomly for training, testing, and validation. The test set error is not used during training, but it is used to compare different models.

Table II shows the experiment parameters used. The training function used is trainlm. The trainlm is a network training
configuration considering not just the demand and weather at that time-period, but also the trend of the metrological condition of the airport within the next time window. In this model, the historical data has been used to train the machine learning model and tested with the metrological and traffic data of the particular flights. The hourly capacity of the predicted configuration is projected based on the information provided in the capacity envelop.

Figure 8 shows the network architecture of the feed-forward neural network. The presented network shows 13-10-1 structure where 13 neurons are connected to the input variables, 10 neurons are used for hidden layer and finally one output neuron. This structure demonstrates the highest accuracy of the prediction model. Figure 9 illustrates the performance of the feed-forward neural network. X-axis presents the epochs and Y-axis presents the mean square error for training, testing and data validation. The neural network is trained using standard backpropagation with a learning rate set to 0.01. As training progresses, the figure shows the errors associated with the values of the current weights and biases. Notice that the error on the training set, testing and on the validation set are varied after 10 epoch. With the above setup, the error reduces with the epochs of training and converges after the 61 epoch. The best validation performance found is 0.00017826.

Figure 10 shows the network architecture of recurrent backpropagation network. Recurrent neural networks are similar to feedforward networks, however, that each layer has a recurrent connection with a tap delay associated with it. This allows the network to have an infinite dynamic response to time series input data.

Figure 11 shows the performance of the recurrent backpropagation network. The error reduces with the epochs of training and converges after the 23 epoch. The best validation performance found is 0.00024469. From the above experiment, the feedforward backpropagation and the recurrent backpropagation network works well for our runway configuration prediction model. Note that, in the feed-forward backpropagation network, the validation data set get the higher error comparing to the testing and training data set which is expected. Table IV presents the machine learning prediction versus actual configuration and corresponding configuration throughput. The target value represents the normalized value of each runway configuration. The prediction value is the estimated value of the machine learning algorithm. Based on the predicted value the neural network decides the feasible configuration. Randomly 20 samples are taken from the test data set and decoded into real configurations. The table lists the all the unique configurations. The predicted value of the network is close to the target value. The hourly throughput of each
configuration is projected from the capacity coverage chart of the Schiphol airport. Notice that, configuration 18R; 18C | 24 projects the highest throughput i.e., 116. On the other hand, configuration 18R | 18L estimates 64 movement which is lowest. The capacity varies due to the number of active runways and influence of the factors mentioned in section III.

TABLE IV: Machine learning prediction versus actual configuration target and corresponding configuration throughput.

<table>
<thead>
<tr>
<th>S.N</th>
<th>Target Configuration</th>
<th>MANN Prediction</th>
<th>Capacity/hour</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>18R</td>
<td>24</td>
<td>18R</td>
</tr>
<tr>
<td>2</td>
<td>18R; 18C</td>
<td>24; 18L</td>
<td>18R; 18C</td>
</tr>
<tr>
<td>3</td>
<td>18R; 18C</td>
<td>24</td>
<td>18R; 18C</td>
</tr>
<tr>
<td>4</td>
<td>18R; 18C</td>
<td>18L</td>
<td>18R; 18C</td>
</tr>
<tr>
<td>5</td>
<td>18R</td>
<td>18L</td>
<td>18R</td>
</tr>
</tbody>
</table>

In conclusion, we also believe that more research is needed to get the insight of the neural network behavior in forecasting the runway configuration and the capacity before definite conclusions are drawn.

REFERENCES