Journal of Open Aviation Science (2024), Vol.**2**
doi:10.59490/joas.2024.8253

JOAS

EDITORIAL

Reviews and Responses for Data-Driven Prediction of Aircraft Holding Times Using OpenSky Data

Authors: Michele Vella, Jason Gauci, and Alexiei Dingli

Reviewers: Max Li, Ramon Dalmau, and Timothé Krauth

Editor: Manuel Waltert

1. Original paper

DOI for the original paper: https://doi.org/10.59490/joas.2024.7890

2. Review - round 1

2.1 Reviewer 1

The authors investigate the use of OpenSky data for predicting aircraft airborne holding times, focusing on the four holding stacks within the TMA surrounding London Heathrow Airport. Although understanding / predicting airborne hold times would be useful for a variety of purposes (e.g., better en route metering, flow control settings, ATFM regulation settings, etc.) , I struggled to see the methodological advancements in the paper: The authors use standard ML approaches (RNN, LSTM), and furthermore the training process was a bit confusing. I provide further detailed comments below:

(1.) The study leverages data from April to August 2023. However, this misses the majority of seasonal trends, e.g., winter-time schedules. The schedule that is being run will impact the number of flights, etc. seeking to arrive to LHR/LGW. This is particularly true for larger hub airports in Europe, where a significant portion of them are slot controlled, and these slots are re-done per season. It would be great if the authors could comment on some of the limitations of their study, particularly as it pertains to data time frame, generalizability, and how future studies could build on this paper but with more comprehensive data.

(2.) A relatively minor point regarding Figure 1 – is this geographically representative of the TMA surrounding LHR and the local London airports? It seems to be a relatively simplistic approximation with a simple circle. Especially seeing how the authors would like to *predict* airborne holding patterns, it would be interesting if they examine the role that airspace geometry potentially plays in how often an aircraft is asked to hold. For example, perhaps a simpler airspace geometry allows for the definition of specific holding stacks / boxes that can be a bit more organized, compared to the case with more complex airspace. If the authors could touch on within the discussion, or perhaps run a new experiment with more realistic TMA boundaries, that would be great. Thanks!

(3.) I am not quite sure what is the purpose of Figure 2. Unless if this heatmap representation is supposed to provide insights into, e.g., distributions of flight positions that are eventually used for

the actual prediction algorithm, what is the advantage of this heatmap view over, e.g., just plotting the trajectories themselves? This figure in its current form would need a legend as well, it is not clear what the different color scales mean. I would suggest that the authors provide a motivation for why this visualization is needed.

(4.) It seems to me that using airport delay as a factor to predict airborne holding times is a bit backwards: Generally airborne holding occurs when there is an unexpected reduction in arrival capacity at the airport, and/or (in rare instances) there is a lack of physical space to continue accepting aircraft on the taxiways, tarmac, etc. Both of these factors can show up in airport delay statistics, but just because an airport has a lot of delays does not necessarily indicate that a lot of airborne holding will happen. It would be great if the authors could better justify their use of delays as a predictor / factor here, rather than something further back in the chain, e.g., current airport arrival rates, etc.

(5.) Figure 3 provides intuition to how airborne holding is detected in the trajectory data. I am surprised that the authors simply used the position of the aircraft in relations to the bounding boxes drawn / visualized to detect holding patterns. Since the authors have full access to the entire trajectory, it seems to me that a much more robust and generalizable approach would have been to detect holding as a function of, e.g., turn rates, heading, etc., i.e., physical parameters of the trajectories, and not just the positions? It is also unclear if the authors consider aspects such as path stretching, additional vectoring, as airborne holding as well. While, strictly speaking, these are more speed control tactics, it would increase the in-air time. This point alos needs to be clarified.

(6.) A minor editing point in Figure 4 – it is relatively clear that the authors modified using, e.g., a paint tool or drawing tool, to add the entry angles into a holding stack. It looks a bit unpolished, especially the uneven/asymmetric arcs representing an angle. I would strongly suggest the authors re-do this figure with more polished modifications.

(7.) This is perhaps just a phrasing / wording comment, but it is quite strong to say "ML algorithms are only capable of handling numerical values." Of course, the authors indicate that the ML algorithms utilized in this paper require some kind of numerical encoding / embedding scheme to be performed prior to ingesting the trajectory data. The authors should clarify that ML algorithms *can* handle non-numerical values, but some kind of transformation must be done first.

(8.) In the paragraph starting on line 150, the authors mention the wake turbulence categories which are used for aircraft separation and sequencing. It would be great if there could be additional discussion on the impact of wake category re-categorization, since I believe RECAT-EU would indeed apply to flights coming into LHR. If the authors could comment on how these additional categories might impact their results, that would be great.

(9.) Table 9 is a bit confusing to me. Why are different window lengths used for the same algorithm when applied to different holding stacks? This seems like it may overfit to a specific holding stack at LHR, which would throw the ability of the ML prediction algorithm to generalize in doubt. Also, I do not see any sensitivity analyses performed on the window lengths – how do we know that for, e.g., the BIG holding stack, that 15 is the optimal window length to use?

2.2 Reviewer 2

The paper presents an application of machine learning to predict aircraft holding times at London Heathrow Airport. The methodology is well-structured and includes the use of both LightGBM and LSTM models.

Specific Comments 1. Introduction

- It might be beneficial to elaborate on how the proposed models could integrate into existing air

traffic control systems. For instance, would real-time data feeds like Delay Index be necessary? Is the Delay Index available in real time or only post-operations?

- A clearer explanation of the interaction between air traffic controllers and these predictions would enhance the reader's understanding. Additionally, discussing the timeframe of predictions would add depth. For example, are predictions limited to an hour in advance, or could they feasibly extend to several hours? Clarifying whether the current look-ahead time is operationally sufficient would provide valuable context.

2. Related Works The literature review could benefit from additional references to contextualize the novelty of this work. For instance, the study presented at last year's OpenSky symposium—*On the Causes and Environmental Impact of Airborne Holdings at Major European Airports* (Dalmau, R.; Very, P.; and Jarry, G., 2023).

3. Methodology

- The methodology section would benefit from an introductory paragraph summarizing its structure and objectives, placed immediately after the "3. Methodology" heading.
- Further clarification on the extraction of information from textual METARs is recommended. Did the authors utilize existing tools or develop custom regular expressions for this purpose?
- There appears to be some ambiguity regarding data access. AeroDataBox is cited as a source, but their website indicates the data is not freely available. It would be helpful to clarify the nature of the data access.
- Regarding the detection of holdings, why did the authors choose not to utilize the method available in the *Traffic* library? A brief comparison of this approach with the one adopted in the study, particularly in terms of predictive performance, would be insightful—even if only discussed qualitatively.
- The use of one-hot encoding (OHE) in LightGBM might deserve reconsideration as it is not recommended by the authors of that algorithm. While the (assumed) rationale of maintaining consistency with neural networks is understandable, exploring alternative encoding methods, such as embeddings, could enhance generalizability.
- A minor but important clarification: using the landing runway to compute wind components during real-time inference might pose challenges, as the runway assignment is only known retrospectively. How is this addressed in the implementation?
- Referring to the task as a "time series problem" is somewhat broad. A more precise term, such as "time series regression" or "regression with time series features," might better describe the methodology. All in all, both regression and time series problems are regression problems.
- The choice of LSTM over simpler alternatives like GRU could be better justified. Is the complexity of the prediction task high enough to necessitate the added sophistication of LSTMs?
- Table 3 introduces separate models for tasks with a single varying parameter. Instead, conditioning the model on this parameter (e.g., look-ahead time, or the holding stack identifier) could allow for improved generalization as the model would be trained with more data.
- LightGBM is used effectively in the study, but it would be helpful to discuss why it was chosen over other GBDT implementations, such as CatBoost or XGBoost. Were these alternatives evaluated?
- Regarding numerical feature scaling, while min-max scaling is effective, ensuring features approximate a Gaussian distribution might enhance neural network performance. Exploring a power transformation followed by standardization could be a worthwhile consideration.

- Dropout and early stopping are used to mitigate overfitting. Was overfitting observed in preliminary experiments? Providing a brief explanation would be helpful.

4. Results and Discussion The explanation of feature importance could be more detailed. For instance, which specific method was used? The reference to "permutation importance" in Section 3.4.2.1 suggests this might refer to features rather than parameters. Clarifying this terminology would improve readability. Additionally, exploring Shapley values might offer more insights into marginal feature contributions. 5. Conclusions and Future Work The conclusions section could place greater emphasis on operational aspects. For instance, what changes, if any, would be required to integrate this model into current operations? Who would use the predictions, and at what point in the decision-making process? Addressing whether the current model is ready for operational deployment—or highlighting areas for improvement—would underscore the practical novelty of the work.

2.3 Reviewer 3

L61: Does this approach work well? What are the strengths and limitations of [4]?

L65: What are the inputs of the model? Are they aircraft trajectories, flight parameters, or something else?

L70: Does this result require improvement?

L72: Is the input of the model an image of a trajectory, including the trajectory and the background map?

Related Work Section: This section could be significantly improved. While it conveys that predicting holding patterns using machine learning is an interesting topic, the methods presented lack detailed explanations. It is unclear how these methods were used in developing your model, what their strengths and limitations are, and which gap in the literature your work addresses. Furthermore, predicting holding patterns with machine learning likely involves substantial data collection, processing, and feature extraction, but this is not clearly articulated. Specifically, it is unclear what types of inputs the methods employed, which would provide a more seamless transition to your methodology. Currently, the reader learns about the data you collected, but the rationale behind selecting this specific data remains unclear.

L82: Does ADS-B data refer to traffic surveillance data?

L85: How did you determine the bounding box?

L90: Is presenting a traffic object here essential? Consider whether this detail is necessary.

L111: It would be helpful to present the general idea of this algorithm. Is its sole purpose to detect flights performing holding patterns and classify them according to specific holding stacks?

L114: Is this the description of the algorithm introduced in L111? This is not entirely clear.

L116: Are there no false positives caused by sharp turns or tromboning maneuvers? Is this criterion applied exclusively within the vicinity of a given holding stack?

L123: Why are two separate algorithms needed for right- and left-turn holding patterns? Is the only difference a +20° or -20° heading change? If so, is it necessary to define them as two distinct algorithms? Additionally, since the direction of holding is typically fixed for a specific holding stack, what happens if a stack has a right-turn holding pattern (e.g., Lambourne)? If the holding direction is indeed fixed, could the algorithm simply use the absolute heading difference instead of separate algorithms?

L127: Providing validation results for this algorithm would be helpful. How confident are you in its accuracy for detecting holding patterns?

L130: How would you explain this observation? Why is it relevant to your study?

L133: Are the manipulations mentioned here those explained later in the chapter?

L136: Do you use the term "dependent variable" in subsequent sections?

L140: This is an awkward phrasing. In machine learning, "numerical values" typically refer to "continuous variables."

L145: Is a score of 30 considered good or bad? Is the score a continuous or integer value? If it's an integer, do you treat it as continuous or categorical?

L140–149: This paragraph is poorly structured. It's unclear which variables were processed, which ones underwent one-hot encoding, and how ATMAP processing relates to the calculation of runway headwind/crosswind. Clarify these points and improve the logical flow.

L159: To clarify, you retrieved the aircraft type via the Traffic library, then determined each flight's wake category and engine type? Aircraft type information is often missing in Traffic—how did you address this issue?

L165: Are these models applied to both datasets, or is each model dedicated to a specific dataset?

L180: First, what is the time length of your input? Are you predicting the average holding time for the next 15 minutes based solely on the previous 15-minute average? Are you using a longer observation window? Second, are you predicting the average holding time for the entire airspace or for specific holding stacks? Lastly, regarding your train/test split, if input-output pairs are correctly aligned (e.g., each input corresponds to the subsequent time period's output), wouldn't shuffling the data be possible? If the observation distribution in the last month differs significantly from prior months due to weather, construction, or other factors, how does your approach handle this? While time series sometimes don't allow random splits, it's unclear if your work falls into this category.

L190: Is it necessary to detail the grid search procedure and hyperparameter space if the final hyperparameter values are not shared?

L197: Does a window length of 1 correspond to a 15-minute average? Additionally, the inability to shuffle your training dataset seems related to using grid search for determining window length. Consider mentioning this earlier.

L201: Are you predicting multiple future timestamps? If so, this should be clarified earlier.

L208: Are five timesteps used? Does the regression model also has sequential data? If not, what are those 5 time steps?

L209: What is the architecture of your model? How many layers does it have, and what is the size of the hidden state? What do the 50 neurons represent? The current description lacks sufficient detail for replication.

L217: Is the exact same architecture used for both array-like and sequential inputs? Unlike for the LGBM model, did you not search for the optimal observation window? Additionally, learning rate is a fundamental parameter—what value did you use?

L241: It seems the LSTM setup contributes to the poor results. Additionally, LSTMs are primarily used for sequential input data, which does not appear to align with this scenario.

L246: What feature importance analysis was performed?

L255: For clarity, consider converting results into hours and minutes.

L257: For the LSTM model, what is the input window length?

L264: The base version of SMOTE may not be suitable for oversampling time series data.

Results and Discussion Section: This section primarily describes results but provides limited explanations. The discussion could be enhanced by elaborating on the factors contributing to good and poor results and providing perspectives on operational implications. Furthermore, limitations of the study are not adequately addressed.

3. Response - round 1

We would like to first thank the reviewers for their attentive reading of our contribution as well as the editor who allowed us to make some corrections. The lines mentioned in the responses refer to the latest updated paper. The Github repository has been updated with the latest coding and datasets.

3.1 Response to reviewer 1

1. The study leverages data from April to August 2023. However, this misses the majority of seasonal trends, e.g., winter-time schedules. The schedule that is being run will impact the number of flights, etc. seeking to arrive to LHR/LGW. This is particularly true for larger hub airports in Europe, where a significant portion of them are slot controlled, and these slots are re-done per season. It would be great if the authors could comment on some of the limitations of their study, particularly as it pertains to data time frame, generalizability, and how future studies could build on this paper but with more comprehensive data.

Response

This issue has been rectified by increasing the dataset to a year (April 2023 till March 2024), to assess seasonal trends. The results have also been updated after retraining the ML models.

2. A relatively minor point regarding Figure 1 – is this geographically representative of the TMA surrounding LHR and the local London airports? It seems to be a relatively simplistic approximation with a simple circle. Especially seeing how the authors would like to *predict* airborne holding patterns, it would be interesting if they examine the role that airspace geometry potentially plays in how often an aircraft is asked to hold. For example, perhaps a simpler airspace geometry allows for the definition of specific holding stacks / boxes that can be a bit more organized, compared to the case with more complex airspace. If the authors could touch on within the discussion, or perhaps run a new experiment with more realistic TMA boundaries, that would be great. Thanks!

Response

This figure is taken directly from another paper study. The actual London TMA boundary is more complex and is not centred on Heathrow (Refer to TMA boundary shown here). The image has been updated, and the circle has been removed so that it does not confuse the reader.

In this study, the London TMA boundary is only considered to determine where – in terms of time – an aircraft is in relation to the boundary e.g. 30 minutes from the boundary. The London TMA is modelled by applying a bounding box (with the TMA co-ordinates taken from UK AIP) to the raw data using the Traffic API.

The TMA airspace is a fixed geometry, which include fixed locations of the holding stacks. To examine

if geometry plays a role in how often an aircraft is asked to hold, it would be necessary to repeat the experiment for different TMAs.

3. I am not quite sure what is the purpose of Figure 2. Unless if this heatmap representation is supposed to provide insights into, e.g., distributions of flight positions that are eventually used for the actual prediction algorithm, what is the advantage of this heatmap view over, e.g., just plotting the trajectories themselves? This figure in its current form would need a legend as well, it is not clear what the different colour scales mean. I would suggest that the authors provide a motivation for why this visualization is needed.

Response

In this case, a heatmap is used to give a graphical representation of the quantity of aircraft at different positions. To give a better understanding of the heatmap, a legend has been added which shows how different colours map to a range of flights. The bounding box shows the locations to which the raw dataset was limited to when downloading it from Open Sky Network.

4. It seems to me that using airport delay as a factor to predict airborne holding times is a bit backwards: Generally airborne holding occurs when there is an unexpected reduction in arrival capacity at the airport, and/or (in rare instances) there is a lack of physical space to continue accepting aircraft on the taxiways, tarmac, etc. Both of these factors can show up in airport delay statistics, but just because an airport has a lot of delays does not necessarily indicate that a lot of airborne holding will happen. It would be great if the authors could better justify their use of delays as a predictor / factor here, rather than something further back in the chain, e.g., current airport arrival rates, etc.

Response

In the initial stage of the study several predictors based on the authors' intuition of what can impact airborne holding were considered. Following the experiment, and after conducting a feature importance analysis, airport delay was found to rank number 13 out of all the predictors. Other metrics like 'Number of landings in the last hour' were also taken into account to have a clearer view of the traffic situation at London Heathrow Airport.

5. Figure 3 provides intuition to how airborne holding is detected in the trajectory data. I am surprised that the authors simply used the position of the aircraft in relations to the bounding boxes drawn / visualized to detect holding patterns. Since the authors have full access to the entire trajectory, it seems to me that a much more robust and generalizable approach would have been to detect holding as a function of, e.g., turn rates, heading, etc., i.e., physical parameters of the trajectories, and not just the positions? It is also unclear if the authors consider aspects such as path stretching, additional vectoring, as airborne holding as well. While, strictly speaking, these are more speed control tactics, it would increase the in-air time. This point also needs to be clarified.

Response

Holding aircraft are not simply detected on the basis of their position in a bounding box i.e. an aircraft is not considered to be holding simply if it is inside one of the bounding boxes. The bounding box was used as a first stage filtering technique to limit the search region. Lines 142-150 explain how a holding flight is detected and how its holding time is calculated in greater detail.

The authors agree that a more generalizable approach would be that suggested by the reviewer. However, since this study focuses on airborne holding time at London Heathrow only and given that its

holding stacks are well-defined by the AIP, a simpler – yet still effective – approach was implemented to focus effort on the holding prediction models themselves. A more generalizable approach may be considered in future work.

This study primarily focuses on airborne holding specifically in vertical holding stacks. Other tactics are considered to be out of scope of this work.

6. A minor editing point in Figure 4 – it is relatively clear that the authors modified using, e.g., a paint tool or drawing tool, to add the entry angles into a holding stack. It looks a bit unpolished, especially the uneven/asymmetric arcs representing an angle. I would strongly suggest the authors re-do this figure with more polished modifications.

Response

This figure has been updated.

7. This is perhaps just a phrasing / wording comment, but it is quite strong to say "ML algorithms are only capable of handling numerical values." Of course, the authors indicate that the ML algorithms utilized in this paper require some kind of numerical encoding / embedding scheme to be performed prior to ingesting the trajectory data. The authors should clarify that ML algorithms *can* handle non-numerical values, but some kind of transformation must be done first.

Response

'ML algorithms are only capable of handling numerical values;' has been replaced with 'Non-numeric variables can only be handled by ML algorithms following some kind of transformation;' as seen in Lines 170-171

8. In the paragraph starting on line 150, the authors mention the wake turbulence categories which are used for aircraft separation and sequencing. It would be great if there could be additional discussion on the impact of wake category re-categorization, since I believe RECAT-EU would indeed apply to flights coming into LHR. If the authors could comment on how these additional categories might impact their results, that would be great.

Response

The models were re-trained using RECAT-EU WTC instead of ICAO WTC. The results show that the RECAT-EU WTC has a negligible effect on ML error metrics and ranks low in terms of feature importance for the regression problem. With regard to the time series problem, LightGBM found that RECAT-EU category D had a high feature importance, as can be observed in Table 11 of the manuscript. Lines 183-188 show the implementation of RECAT-EU.

9. Table 9 is a bit confusing to me. Why are different window lengths used for the same algorithm when applied to different holding stacks? This seems like it may overfit to a specific holding stack at LHR, which would throw the ability of the ML prediction algorithm to generalize in doubt. Also, I do not see any sensitivity analyses performed on the window lengths – how do we know that for, e.g., the BIG holding stack, that 15 is the optimal window length to use?

Response

In this study, it was decided to create a separate model for each holding stack; thus, each model is optimised for the specific holding stack. The optimal window length for each model was determined by using a grid search as mentioned in Line 248. This is the reason why the optimal window length varies from one model to another.

3.2 Response to reviewer 2

1. It might be beneficial to elaborate on how the proposed models could integrate into existing air traffic control systems. For instance, would real-time data feeds like Delay Index be necessary? Is the Delay Index available in real time or only post-operations?

Response

The Conclusion section has been updated to address this comment as seen from Lines 359-363. The proposed models could be integrated with the AMAN (or E-AMAN) to optimise aircraft sequencing and reduce delays in the TMA.

Yes, a real-time data feed of delay index would be necessary.

Also note that the Delay Index can be determined from AeroDataBox in real time as seen in Figure 1. This index is produced once every 15 minutes.

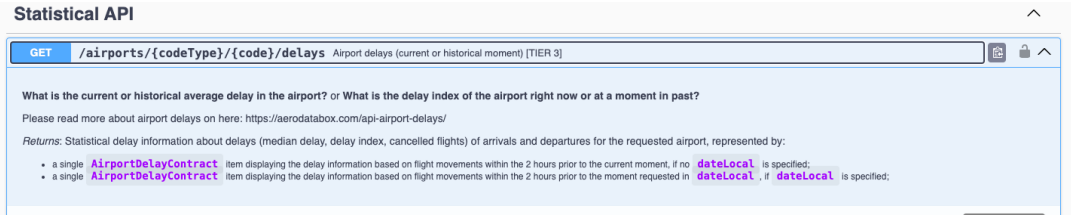

Figure 1. AeroDataBox API

2. A clearer explanation of the interaction between air traffic controllers and these predictions would enhance the reader's understanding. Additionally, discussing the timeframe of predictions would add depth. For example, are predictions limited to an hour in advance, or could they feasibly extend to several hours? Clarifying whether the current look-ahead time is operationally sufficient would provide valuable context.

Response

Models 1-4 predict the holding time of a specific aircraft that is situated at a particular time from the TMA boundary. Model 1 at the TMA, Model 2, 30 minutes before TMA and Model 3, 1 hour before the TMA i.e 1 hour in advance. This can be extended to several hours, even to the take-off time of a particular flight. This would be more operationally useful from the pilots' perspective but would come at the expense of increased prediction uncertainty. It would also require the creation of new regression models e.g. a model to predict holding time when an aircraft is 2 hours from the TMA, etc...

In the case of Models 5-8, the models can be used 'as is' to forecast holding time at any point in the future in 15-minute intervals. Thus, they could be used, for instance, to forecast average holding time in a particular holding stack over the next 24 hours.

Refer to Lines 364-371

2. Related Works

3. The literature review could benefit from additional references to contextualize the novelty of this work. For instance, the study presented at last year's OpenSky symposium—On the Causes and Environmental Impact of Airborne Holdings at Major European Airports (Dalmau, R.; Very, P.; and Jarry, G., 2023).

Response

Another 2 references have been added to the Related Works section to show the novelty and importance of this study, as seen in Lines 87-98.

3. Methodology

4. The methodology section would benefit from an introductory paragraph summarizing its structure and objectives, placed immediately after the "3. Methodology" heading.

Response

A paragraph has been added right after the heading, to explain the structure and purpose of the methodology section.

5. Further clarification on the extraction of information from textual METARs is recommended. Did the authors utilize existing tools or develop custom regular expressions for this purpose?

Response

The raw METAR was converted to a numerical score using the ATMAP algorithm which was developed by EUROCONTROL as explained in Lines 173-177. This link provides further information on how this algorithm works and how it calculates a score depending on the METAR information.

6. There appears to be some ambiguity regarding data access. AeroDataBox is cited as a source, but their website indicates the data is not freely available. It would be helpful to clarify the nature of the data access.

Response

The authors had a paid subscription to access data from AeroDataBox. A footnote has been added at the end of Page 5 of the manuscript to clarify that this data is not freely available.

7. Regarding the detection of holdings, why did the authors choose not to utilize the method available in the Traffic library? A brief comparison of this approach with the one adopted in the study, particularly in terms of predictive performance, would be insightful—even if only discussed qualitatively.

Response

The authors are aware of the holding method available in the Traffic Library, however, very little documentation is available on this method regarding its performance and reliability. A presentation on this holding method was given at the most recent Open Sky Symposium. Also, this method only states whether an aircraft is in a holding pattern or not (classification problem); it does not provide the duration of holding whereas the method proposed in this study calculates the duration of airborne holding. In the future the performance of the Traffic holding method could be compared to the approach used

in our study.

8. The use of one-hot encoding (OHE) in LightGBM might deserve reconsideration as it is not recommended by the authors of that algorithm. While the (assumed) rationale of maintaining consistency with neural networks is understandable, exploring alternative encoding methods, such as embeddings, could enhance generalizability.

Response

This will be investigated further in future work to analyse the impact of alternative encoding methods. This is mentioned in the future works section as seen in Lines 374-376

9. A minor but important clarification: using the landing runway to compute wind components during real-time inference might pose challenges, as the runway assignment is only known retrospectively. How is this addressed in the implementation?

Response

The authors acknowledge that this would pose challenges in practice. To overcome this, a seperate model could be trained to predict the runway(s) in use. The output of this model could then be input to our models to predict the holding time.

Dataset 1 has been updated for models 2-4 where the actual landing runway at the time of prediction was used to train the models. In the case of Model 1 in normal operations, the pilots would be given the runway in use by the time they reach the TMA.

The implementation has been updated, so the landing runways will be those at the time of the prediction. Refer to Lines 178-181

10. Referring to the task as a "time series problem" is somewhat broad. A more precise term, such as "time series regression" or "regression with time series features," might better describe the methodology. All in all, both regression and time series problems are regression problems.

Response

'time series forecasting' has been updated to 'time series regression' throughout the paper.

11. The choice of LSTM over simpler alternatives like GRU could be better justified. Is the complexity of the prediction task high enough to necessitate the added sophistication of LSTMs?

Response

Simpler alternatives like GRU have not been investigated, but during the literature review the authors found that LSTM produces better error metrics when compared to GRU. For instance, a study entitled 'A deep learning-based approach for predicting in-flight estimated time of arrival ' has shown that LSTM is a viable approach to ETA prediction in ATM and can surpass other techniques that are the state of the art at this task, such as ensemble and boosting machine learning methods. This study tested using GRUs instead of LSTM units, but found no significant differences in model performance, so they excluded them from the study. Also, LSTM have shown better performance when having a large dataset as we have in our study.

Given the nature of the problem – with sequential data and long-term dependencies – it was decided to use LSTM given their success in related studies/applications. Other approaches will be considered

in future work. The selection of LSTM for our study is explained in Section 3.4, Lines 213-218.

12. Table 3 introduces separate models for tasks with a single varying parameter. Instead, conditioning the model on this parameter (e.g., look-ahead time, or the holding stack identifier) could allow for improved generalization as the model would be trained with more data.

Response

Point taken. However, in this study, we wanted to develop separate models that are tuned for specific look-ahead times and specific holding stacks. This is something that will be considered in future work as mentioned in the conclusion Lines 372-378

13. LightGBM is used effectively in the study, but it would be helpful to discuss why it was chosen over other GBDT implementations, such as CatBoost or XGBoost. Were these alternatives evaluated?

Response

Various studies found that LightGBM outperformed other GBDT implementations. In a study entitled 'LightGBM: A Highly Efficient Gradient Boosting Decision Tree' , LightGBM was trained on multiple datasets even on flight delays, and it was found that LightGBM speeds up the training process of conventional GBDT by up to over 20 times while achieving almost the same accuracy.

In addition, a paper entitled 'Flight delay prediction based on LightGBM' has shown that the LightGBM algorithm outperforms comparative algorithms in terms of R-squared, MAE and training time metrics, showing higher prediction accuracy and shorter training time compared to the XGBoost and GBDT algorithms.

For these reasons, LightGBM was selected on the basis of performance achieved in similar applications. To highlight these choices, references have been added in Lines 208-212

14. Regarding numerical feature scaling, while min-max scaling is effective, ensuring features approximate a Gaussian distribution might enhance neural network performance. Exploring a power transformation n followed by standardization could be a worthwhile consideration.

Response

PowerTransformer from sklearn API has been applied as a preprocessing technique for LSTM (regression), also coupled with standardization (mean=0, std=1). The RMSE and MAE of the models has not been affected. On the other hand, the number of epochs for the model to converge (using the Early Stopping) has increased from 37 to 46 (Model 1). A plot of validation and training loss before and after power transformation can be seen on the next page. From Figures 2 and 3, one can see that the validation and training loss graph are much closer. This change in the pre-processing has been documented in Lines 255-257. With regard to the time-series forecasting, the MinMax Scaler gave superior predictions when compared to the PowerTransformer.

15. Dropout and early stopping are used to mitigate overfitting. Was overfitting observed in preliminary experiments? Providing a brief explanation would be helpful.

Response

In the preliminary models (before performing hyperparameter tuning) the number of epochs was set at a high value and overfitting was observed in some of the models. Overfitting was reduced by decreasing

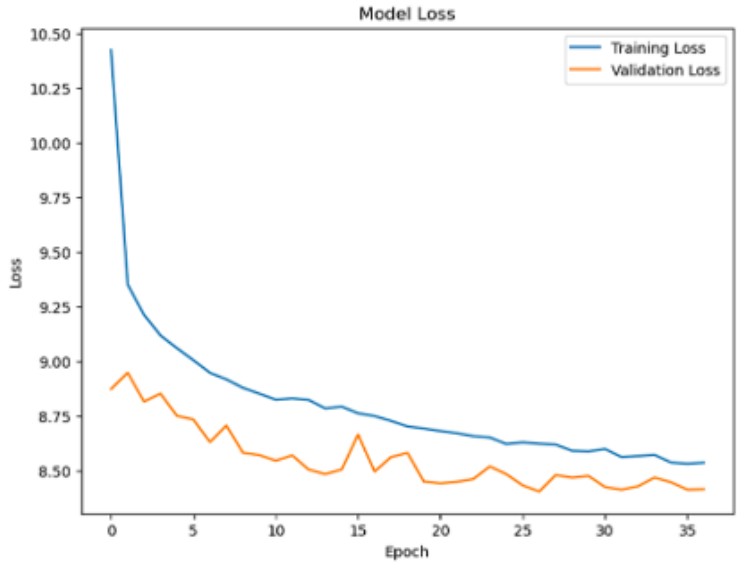

Figure 2. Traning and Validation Loss graphs before PowerTransformer and Standardization was applied.

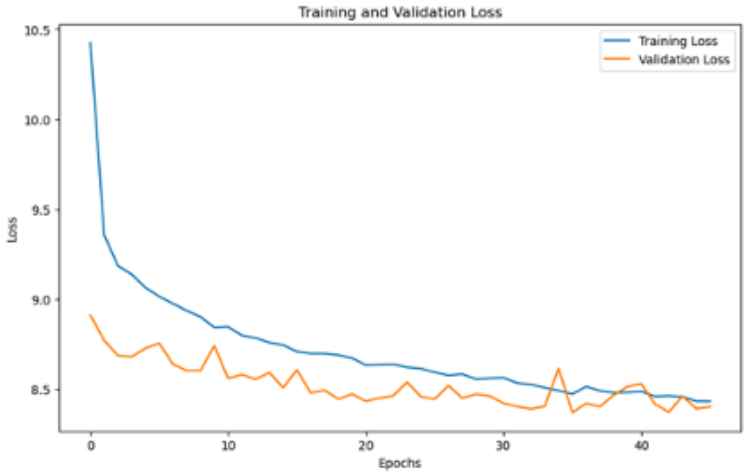

Figure 3. Traning and Validation Loss graphs after PowerTransformer and Standardization was applied.

the number of epochs. With the help of hyperparameter grid search and dropout methods, the error metrics improved and the training and validation graphs (Figure 3) converged better and the best fit was found.

4. Results and Discussion

16. The explanation of feature importance could be more detailed. For instance, which specific method was used? The reference to "permutation importance" in Section 3.4.2.1 suggests this might refer to features rather than parameters. Clarifying this terminology would improve readability. Additionally, exploring Shapley values might offer more insights into marginal feature contributions.

Response

In the previous version of the paper, various feature importance methods were used. In this (revised) version, the SHAP values were obtained and compared with each other and used as the main feature importance method to facilitate comparisons between models. Table 8 has been updated with SHAP summary plots for all models. Figures 7 and 8 (refer to manuscript) show the Shap values for Models 4 and 5 using each algorithm.

17. The conclusion section could place greater emphasis on operational aspects. For instance, what changes, if any, would be required to integrate this model into current operations? Who would use the predictions, and at what point in the decision-making process? Addressing whether the current model is ready for operational deployment—or highlighting areas for improvement—would underscore the practical novelty of the work.

Response

This comment relates to the first comment and is now addressed in Lines 359-371.

3.3 Response to reviewer 3

1. L61: Does this approach work well? What are the strengths and limitations of [4]?

Response

The models can predict ETA with an MAE of less than 6 minutes and 3 minutes, immediately after departure and immediately after TMA entrance, respectively. One strength is that the models can be applied to other airports (with similar runway configurations) without modifications. A limitation is that departure traffic, airport and airspace congestions are not considered. This comment has been added in the related work section Lines 68-70

2. L65: What are the inputs of the model? Are they aircraft trajectories, flight parameters, or something else?

Response

The inputs to the model are flight information, flight parameters, flight paths, weather, surrounding traffic, and airport performance metrics. This has been rectified in Lines 74-75 . Table 1 of the related study provides a more detailed illustration of the parameters.

3. L70: Does this result require improvement?

Response

While the result is good, there is always room for improvement e.g. to obtain the same or better prediction accuracy at even longer distances (than 500NM) from the airport. Furthermore, the models were only tested at Singapore Changi Airport, so there is no guarantee that they would perform well at other airports. This was addressed in Lines 79-80

4. L72: Is the input of the model an image of a trajectory, including the trajectory and the background map?

Response

The input of the model is an image with the target aircraft trajectory labelled red and all background aircraft trajectories labelled as blue. The background map is removed from the image. This can be found on the first page of the Introduction section of the related study.

5. Related Work Section: This section could be significantly improved. While it conveys that predicting holding patterns using machine learning is an interesting topic, the methods presented lack detailed explanations. It is unclear how these methods were used in developing your model, what their strengths and limitations are, and which gap in the literature your work addresses. Furthermore, predicting holding patterns with machine learning likely involves substantial data collection, processing, and feature extraction, but this is not clearly articulated. Specifically, it is unclear what types of inputs the methods employed, which would provide a more seamless transition to your methodology. Currently, the reader learns about the data you collected, but the rationale behind selecting this specific data remains unclear.

Response

The related work section has been extended, improved, and more references have been added. The research gap has been more clearly identified. Lines 87-101, elaborates on this point.

6. L82: Does ADS-B data refer to traffic surveillance data?

Response

Yes, it refers to the raw data taken from the transponder data sent from each aircraft and received by the Open Sky Network feeders.

7. L85: How did you determine the bounding box?

Response

The bounding box was determined by making sure that all data points were captured for all flights that are 1 hour away from the TMA boundary. This has been rectified in Lines 112-115.

8. L90: Is presenting a traffic object here essential? Consider whether this detail is necessary.

Response

The sentence 'The data was downloaded in a traffic format structure, which consists of a group of flight attributes,' has been removed.

9. L111: It would be helpful to present the general idea of this algorithm. Is its sole purpose to detect flights performing holding patterns and classify them according to specific holding stacks?

Response

The purpose of the algorithm is:

1. To detect holding flights in each holding stack and

2. To measure the duration of airborne holding of each holding flight

The algorithm is described in Lines 142-150. Building on the bounding box approach applied to each

holding stack, the next step involves examining changes in the heading of an aircraft over time to precisely pinpoint when it enters and exits a holding pattern. A sentence has been added at Lines 142-144 to link the following paragraphs together and help the reader understand what steps are going to be discussed.

10. L114: Is this the description of the algorithm introduced in L111? This is not entirely clear.

Response

Yes, it is.

11. L116: Are there no false positives caused by sharp turns or tromboning maneuvers? Is this criterion applied exclusively within the vicinity of a given holding stack?

Response

The algorithm is only applied to data points that fall within the bounds of the holding stacks (as shown in Figure 3 of the manuscript). This is mentioned in Lines 140-142.

12. L123: Why are two separate algorithms needed for right- and left-turn holding patterns? Is the only difference a +20° or -20° heading change? If so, is it necessary to define them as two distinct algorithms? Additionally, since the direction of holding is typically fixed for a specific holding stack, what happens if a stack has a right-turn holding pattern (e.g., Lambourne)? If the holding direction is indeed fixed, could the algorithm simply use the absolute heading difference instead of separate algorithms?

Response

In reality, there is 1 algorithm which takes into account the direction of the holding stack. This has been made clearer in Lines 154-155

13. L127: Providing validation results for this algorithm would be helpful. How confident are you in its accuracy for detecting holding patterns?

Response

The algorithm was validated by manually looking at the trajectory and data points of selected flights as stated in Lines 155-157. Using the bounding box helped to filter the required data points. In future work, the results of this algorithm could be compared to the Traffic API Holding classification method. The only limitation of the study is that since the data has been downloaded every 30 seconds, the holding time computed has an error of +/- 30 seconds.

14. L130: How would you explain this observation? Why is it relevant to your study?

Response

Figure 5 of the manuscript shows more flights hold in BNN and LAM than in the other holding stacks. The number of holding flights in each holding stack is relevant as it corresponds to the size of the training dataset available for each of the models developed in this study.

15. L133: Are the manipulations mentioned here those explained later in the chapter?

Response

Yes (e.g. parameters with string values are converted to numerical scores; METAR data is converted to a numerical score; etc.). To help the reader at Lines 163-164 'The various manipulations are described in the rest of this section' sentence has been added.

16. L136: Do you use the term "dependent variable" in subsequent sections?

Response

No. This was only added for readers who may be more familiar with this term than 'target' or 'output' variable.

17. L140: This is an awkward phrasing. In machine learning, "numerical values" typically refer to "continuous variables."

Response

It is common knowledge that numerical values can be either discrete (integer) or continuous (having a decimal place). For clarity the wording in Line 171 has been changed to 'Non-numeric variables can only be handled by ML algorithms following some kind of transformation;' to address a comment with one of the other reviewers.

18. L145: Is a score of 30 considered good or bad? Is the score a continuous or integer value? If it's an integer, do you treat it as continuous or categorical?

Response

Lines 176-177 have been updated to show that the score is an integer value and the higher the value the more severe weather conditions are. Also, Tables 1-2 of the manuscript show that the ATMAP weather has an integer value.

19. L140–149: This paragraph is poorly structured. It's unclear which variables were processed, which ones underwent one-hot encoding, and how ATMAP processing relates to the calculation of runway headwind/crosswind. Clarify these points and improve the logical flow.

Response

The variables that were one-hot encoded are listed in Tables 1,2 of the manuscript. The ATMAP processing is different from the headwind /crosswind component calculation. While ATMAP gives a score the headwind /crosswind component is calculated using knowledge of the METAR wind and landing runway. At line 177, the paragraph has been divided to separate the ATMAP algorithm from the runway headwind/crosswind calculation.

20. L159: To clarify, you retrieved the aircraft type via the Traffic library, then determined each flight's wake category and engine type? Aircraft type information is often missing in Traffic—how did you address this issue?

Response

Yes, that is how I determined the flight's wake category and engine type. As shown in Figure 4 Traffic API has an inbuilt method named aircraft_data() which returns aircraft type from the OSN aircraft database. This method works by matching the ICAO24 identifier from the raw data to the database.

Lines 191-195 have been added to clarify how the Type Code has been obtained.

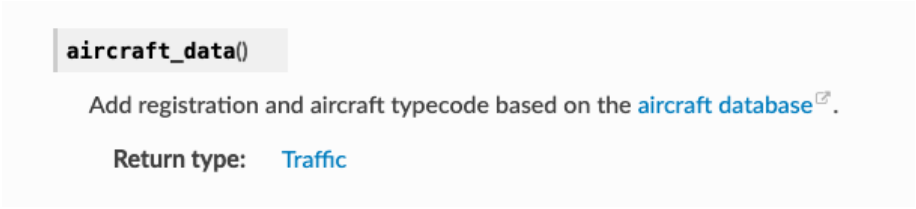

Figure 4. aircraft data() method from Traffic API.

21. L165: Are these models applied to both datasets, or is each model dedicated to a specific dataset?

Response

Separate models were created for each dataset as shown in Table 3 presented in the manuscript.

22. L180: First, what is the time length of your input? Are you predicting the average holding time for the next 15 minutes based solely on the previous 15-minute average? Are you using a longer observation window?

Response

With regard to time series regression of LSTM, the time length is 7.5 hours which corresponds to a window length of 30; this means that the previous 30 15-minute averages are used to predict the average holding in the next 15 minutes. This is stated in Lines 274-275. With regards LightGBM time series regression, the window length varies with the model as stated in Table 9 of the manuscript.

23. Second, are you predicting the average holding time for the entire airspace or for specific holding stacks?

Response

No, the average holding time is predicted for specific holding stacks, each having an individual model. Models 5-8 are the models trained for each stack as shown in Table 3 of the manuscript.

24. Lastly, regarding your train/test split, if input-output pairs are correctly aligned (e.g., each input corresponds to the subsequent time period's output), wouldn't shuffling the data be possible? If the observation distribution in the last month differs significantly from prior months due to weather, construction, or other factors, how does your approach handle this? While time series sometimes don't allow random splits, it's unclear if your work falls into this category.

Response

LightGBM and LSTM can capture long-term dependencies and seasonal patterns/trends, so they should cope with large changes from one month to another. Data should not be shuffled for the forecasting (time series regression) problem since the data points should be organised sequentially in time to capture patterns. The training dataset covers a large enough period to train the models accordingly.

25. L190: Is it necessary to detail the grid search procedure and hyperparameter space if the final hyperparameter values are not shared?

Response

The final hyperparameters are presented in Tables 6,9 and 10 of the manuscript. A reference to the optimal hyperparameters has been made in the methodology section as seen in Lines 243-244.

26. L197: Does a window length of 1 correspond to a 15-minute average? Additionally, the inability to shuffle your training dataset seems related to using grid search for determining window length. Consider mentioning this earlier.

Response

Yes, a window length of 1 corresponds to a 15-minute average. Lines 248-249 have been updated to clarify this. This is also stated in a footnote in Table 9 of the manuscript.

27. L201: Are you predicting multiple future timestamps? If so, this should be clarified earlier.

Response

In our analysis we are just predicting the next 15-minute average holding time, Line 221 has been updated to clarify this point. Also, Table 3 of the manuscript has been updated to show that the next 15 minutes average holding time is predicted.

28. L208: Are five timesteps used? Does the regression model also has sequential data? If not, what are those 5 time steps?

Response

This is a typo, the LSTM regression model has a window length of 1 since the data is not sequential. This has been corrected in Line 258.

29. L209: What is the architecture of your model? How many layers does it have, and what is the size of the hidden state? What do the 50 neurons represent? The current description lacks sufficient detail for replication.

Response

The requested details have been added in Lines 261-265

30. L217: Is the exact same architecture used for both array-like and sequential inputs? Unlike for the LGBM model, did you not search for the optimal observation window? Additionally, learning rate is a fundamental parameter—what value did you use?

Response

Yes, the same architecture was used but for the regression, the inputs were not sequential to have consistency with the LightGBM Model. The observation window was not made part of the grid search since a higher window yielded to better error metrics, up to a point where the difference was negligible. By running the algorithm on different windows, the optimal window was found to be 30. The learning rate was incorporated into the grid search, and this has been reflected in Tables 5 and 6 of the manuscript.

31. L241: It seems the LSTM setup contributes to the poor results. Additionally, LSTMs are primarily used for sequential input data, which does not appear to align with this scenario.

Response

With regards to the regression analysis (Dataset 1), the window length of the LSTM model was set as 1 so as to take in account that the data is not sequential.

32. L246: What feature importance analysis was performed?

Response

Initially for LightGBM the inbuilt feature importance method was applied, while permutation importance was applied for LSTM. For consistency and comments by the other reviewers, the SHAP values feature importance method was used for a consistent comparison for all models.

33. L255: For clarity, consider converting results into hours and minutes.

Response

All results have been converted to hours and minutes.

34. L257: For the LSTM model, what is the input window length?

Response

The input window length is 7 hours and 30 minutes as stated In Lines 313-315. Table 10 of the manuscript is also updated to show the window length.

35. L264: The base version of SMOTE may not be suitable for oversampling time series data.

Response

Instead of SMOTE a different augmentation method such as DTW-SMOTE or T-SMOTE can be applied. This has been rectified in Lines 323 -325.

36. Results and Discussion Section: This section primarily describes results but provides limited explanations. The discussion could be enhanced by elaborating on the factors contributing to good and poor results and providing perspectives on operational implications. Furthermore, limitations of the study are not adequately addressed.

Response

More analysis has been performed in the results and discussion section together with more information regarding the feature importance. The limitations of the study has been addressed in Lines 359-371

4. Review - round 2

4.1 Reviewer 1

I sincerely appreciate the efforts made by the authors to address every one of my comments. I am satisfied with the authors' edits and rebuttals. I have no additional comments, and am happy to recommend acceptance.

4.2 Reviewer 2

This paper attempts to apply machine learning (ML) techniques, specifically LightGBM and LSTM, to predict aircraft holding times at London Heathrow. While the research is relevant to air traffic management, I have identified several weaknesses and critical points that the authors must address before this paper can be considered for publication.

Introduction

The introduction briefly references the Point Merge System (PMS). The paper would benefit from mentioning other major airports using PMS for better context.

Related Works

The authors state that ML techniques have not been used to predict holding times. They should reframe this to better highlight the gap their work fills, focusing on the lack of predictive tools for holding times, not just the novelty of applying ML.

Methodology

- The 1380 NM by 1320 NM bounding box is excessive. Why such a large box was used to filter the traffic is unclear. The authors must justify this size or provide a more reasonable alternative.

- Figure 2: The inclusion of this figure is questionable. It's just a heatmap of arrivals at Heathrow. It provides no clear conclusion or insight. Additionally, the term "one day" is vague — which day was used?

- There is no explanation of how METAR data was processed, especially regarding how the weather fields were extracted from textual reports. This must be clarified for reproducibility and transparency.

- The journal emphasizes open science, yet no code or data repository is linked in the paper. The use of AeroDataBox, which requires a paid subscription, is also problematic (but I understand nothing can be done at this point). It would have been nice to create a synthetic dataset just for researchers to reproduce the methodology.

- The method for defining a bounding box around each individual holding stack lacks clarity. It is essential that the authors provide precise details on how these bounding boxes were determined.

- No (scientific) performance metrics provided for the method to detect holdings. Given the reliance on this detection, the authors should validate this method against ground truth data and provide metrics for its performance. Comparing this approach with the Traffic library's methods is also recommended.

- One-hot encoding (OHE) is not recommended for LightGBM, as per the creators of the model. The authors should acknowledge this and either justify their decision or explore better encoding options for categorical data — or at least warn the readers that OHE is not the right approach when using LightGBM (and other GBDT models like CatBoost).

- The approach to assign a landing runway to each individual flight works because Heathrow has a single runway in use, but this is not generalizable to other airports. The authors must clarify how the model can be adapted to airports with multiple landing runways. Specifically, how would the authors identify the landing runway of each individual flight? One could use the methods provided by the Traffic library (which work very well). The authors could give some guidance to the readers on this matter.

- The authors use RECAT-EU and Aircraft Type features in the regression problem but do not clearly explain how they are integrated into the time series model. Table 2 makes this clearer, but the explanation should be presented earlier in the text.

- In Table 1, the time of day is treated as a continuous variable, but this leads to issues at midnight. It should be treated as a categorical or cyclic variable to account for discontinuities at midnight.

- The authors specify units (kt, ...) in Table 1, but GBDT models like LightGBM are not sensitive to scaling. This section could be revised to remove unnecessary details.

- The authors justify using LightGBM based on its performance in another paper. However, CatBoost is known to handle categorical features better, and a comparison between LightGBM and CatBoost would be a valuable addition to this paper.

- While the authors justify using LSTM based on its success in another work, the performance improvements over simpler models like GRU are not sufficiently discussed. A comparison between LSTM and GRU models would be valuable, as GRUs can sometimes offer a more efficient alternative to LSTMs.

- The authors use different data splits for regression and time series problems. It would have been more logical to use consistent time frames across both problems to ensure a fair comparison of results in the test set between the regression and time series models.

- Why PowerTransformer was used in the regression and MaxMin scaling in the time series. This is not clear and must be clarified.

- Why not use a 9th Model with 4 heads, each one predicting the average delay of each holding stack in a multi-output setting? This could be included to the "future work" section.

- The authors use an LSTM with 50 neurons and 4 layers, and dropout of 0.2, but they do not explain why these particular choices were made. These hyperparameters should be optimized, and the authors should explain why such architecture was chosen in the first place.

Results and Discussion

- The inclusion of table 8 is redundant. The SHAP values could be better visualized in figures. Tables here are unnecessary and distract from the discussion.

- The explanation of SHAP values is too simplistic. The authors should provide more context for readers who may not be familiar with these techniques, explaining how they contribute to the model interpretation.

- The readability of Figure 9 is poor. The authors should try using a line plot instead to improve the clarity of their results.

Conclusion and Future Work

- Rather than building separate models for each look-ahead time, the authors could integrate this feature into a single model. This would improve generalization and make the implementation easier.

- The authors mention future comparisons between encoding techniques, but they fail to do this in the current paper. This is a relatively easy experiment to conduct.

I understand that many of the comments above, such as *"The authors mention future comparisons between encoding techniques, but they fail to do this in the current paper. This is a relatively easy experiment to conduct and would provide valuable insights."* may imply the need to carry out additional experiments. However, I would like to clarify that performing these experiments is not strictly necessary at this stage. It would be entirely acceptable if the authors explicitly outline in the manuscript

what they propose to explore in future work or suggest these points as recommendations for the research community. This would adequately address the identified gaps without overburdening the current scope of the paper.

4.3 Reviewer 3

The manuscript greatly improved and is much clearer. The authors carefully addressed all the comments. The reader can clearly grasp the stakes of the study as well as the different model implemented. Here are additional minor comments:

- I find the literature review better explained and we get a clear overview about previous work concerning holding patterns. However, the clear identification of the gaps still needs minor improvements:
  - The gap you identified is the lack of ML methods to predict holding times?
  - Why ML methods might be more adapted to the task than previous methodologies?
  - Is your paper purely exploratory: the main goal is to check if ML methods have the potential to overcome previous work?
  - Why LSTM and LGBM seem to be the obvious choice? Previous work in other field already demonstrated the reliability of those architectures for similar problems?
- When detecting a holding pattern, you compare the heading for two consecutive observations separated by 30s. If the heading is more than 20 degrees, then the holding count begins. However, the proper holding pattern could have been initiated anywhere between those two observations, which might lead to an error up to 30s between the real holding pattern start, and the detected one. Thus, from one sample to another in your training dataset, the detected start can be either the very beginning of the holding, or up to 30s in the holding. The same consideration holds for the end, which can lead to discrepancies up to one minute overall. Thus, your model performance depends on your ML model as much as on your detection algorithm. It might be worth to explain further the validation of your model (e.g. comparing the calculated times with the one manually labeled), and also mention it in the limitations if necessary.
- It might be beneficial to remind the size of each constructed training datasets.
- According to figure 9, the model seems to constantly underestimate the holding time. Do you have an idea about the reason?

5. Response - round 2

5.1 Response to reviewer 1

I sincerely appreciate the efforts made by the authors to address every one of my comments. I am satisfied with the authors' edits and rebuttals. I have no additional comments, and am happy to recommend acceptance.

Response

Thank you for your considerate review and for taking the time to go through our revisions so thoroughly. We're very pleased to hear that our responses and edits have addressed your concerns, and we appreciate your recommendation for acceptance.

5.2 Response to reviewer 2

1. The introduction briefly references the Point Merge System (PMS). The paper would benefit from mentioning other major airports using PMS for better context.

Response

PMS is now described on lines 27-31 of the manuscript. Airports using this system are also mentioned.

2. The authors state that ML techniques have not been used to predict holding times. They should reframe this to better highlight the gap their work fills, focusing on the lack of predictive tools for holding times, not just the novelty of applying ML.

Response

Lines 104-107 of the manuscript have been rephrased accordingly.

3. The 1380 NM by 1320 NM bounding box is excessive. Why such a large box was used to filter the traffic is unclear. The authors must justify this size or provide a more reasonable alternative.

Response

The size of the bounding box is derived from the fact that, in the case of Models 1-4, we are predicting the holding time of individual aircraft when they are at a certain time from the TMA boundary (30 minutes from the TMA in the case of Model 2 and 60 minutes in the case of Model 3). And, for these models, various parameters (speed, altitude, etc.) are required corresponding to the moment that each aircraft is at each of these times (0, 30 or 60 minutes) from the TMA boundary. The edges of the bounding box correspond to the outermost data points obtained from the OpenSky Network. This is also visible in Figure 2, where the aircraft closest to the edges of the bounding box are 1 hour away from the TMA boundary.

4. Figure 2: The inclusion of this figure is questionable. It's just a heatmap of arrivals at Heathrow. It provides no clear conclusion or insight. Additionally, the term "one day" is vague — which day was used?

Response

The purpose of this figure is to show a typical spatial distribution of flights into Heathrow airport and to identify major traffic flows. The exact date of the captured data has been added to the figure caption. (16 th March 2024)

5. There is no explanation of how METAR data was processed, especially regarding how the weather fields were extracted from textual reports. This must be clarified for reproducibility and transparency.

Response

Lines 196-201 of the manuscript have been updated to help the reader understand how the ATMAP algorithm processes the raw METAR. More details can be found in the ATMAP reference document provided.

6. The journal emphasizes open science, yet no code or data repository is linked in the paper. The use of AeroDataBox, which requires a paid subscription, is also problematic (but I understand nothing

can be done at this point). It would have been nice to create a synthetic dataset just for researchers to reproduce the methodology.

Response

Unfortunately, no other API could be found that provides the delay index freely. However, as listed in the Open Data Statement, the train/test datasets and the code have been uploaded to Github and can be accessed by the readers.

7. The method for defining a bounding box around each individual holding stack lacks clarity. It is essential that the authors provide precise details on how these bounding boxes were determined.

Response

The bounding box was defined manually for each holding stack in such a way that it captures the extremities of the corresponding holding stack as defined in the UK NATS AIP, Enroute charts. Below, find an example of the OCKHAM stack as found in the AIP. Refer to Lines 148-150 in the updated manuscript.

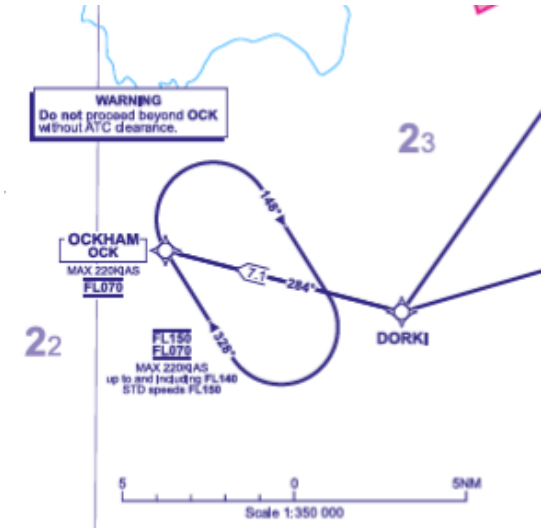

Figure 5. OCKHAM holding pattern stack taken from UK NATS AIP.

8. No (scientific) performance metrics provided for the method to detect holdings. Given the reliance on this detection, the authors should validate this method against ground truth data and provide metrics for its performance. Comparing this approach with the Traffic library's methods is also recommended.

Response

Manual validation, as well as a comparison between the proposed algorithm and the corresponding method in the Traffic API, has been performed. Refer to Lines 166-175 of the updated manuscript for details.

9. One-hot encoding (OHE) is not recommended for LightGBM, as per the creators of the model. The authors should acknowledge this and either justify their decision or explore better encoding options

for categorical data — or at least warn the readers that OHE is not the right approach when using LightGBM (and other GBDT models like CatBoost).

Response

The Fisher integer encoding technique has been applied to all LightGBM models. This is also reflected in Tables 1 and 2. Lines 193-195 have been added to describe this change. LightGBM categorical encoding has had a negligible impact on the error metrics.

10. The approach to assign a landing runway to each individual flight works because Heathrow has a single runway in use, but this is not generalizable to other airports. The authors must clarify how the model can be adapted to airports with multiple landing runways. Specifically, how would the authors identify the landing runway of each individual flight? One could use the methods provided by the Traffic library (which work very well). The authors could give some guidance to the readers on this matter.

Response

Normally, pilots do not know in advance which runway they will land on until they are inside the TMA, especially in the case of airports with multiple landing runways. Around this time, the pilots would listen to the arrival ATIS (or D-ATIS) and would know which landing runway(s) is active. Even then, the landing runway may change at short notice (late runway change).

In this study, it is simply assumed that the landing runway for an aircraft is the runway in use at the time of the prediction, which may be different from the runway that the aircraft eventually lands on (even at London Heathrow, the landing runway may change). Therefore, to predict the landing runway for an aircraft (in advance), one possibility would be to train a separate model to predict the landing runway. This is mentioned in the Future Work section of the updated manuscript (Lines 416-417).

The Traffic API provides the 'ILS_max' method which can be used to determine the landing runway from historic data; however, this is only known a posteriori.

11. The authors use RECAT-EU and Aircraft Type features in the regression problem but do not clearly explain how they are integrated into the time series model. Table 2 makes this clearer, but the explanation should be presented earlier in the text.

Response

Lines 228-229 of the manuscript have been revised to clarify this point before presenting the tables.

12. In Table 1, the time of day is treated as a continuous variable, but this leads to issues at midnight. It should be treated as a categorical or cyclic variable to account for discontinuities at midnight.

Response

For all models the time of day is now being treated as a cyclic variable where both cosine and sine transformations were done. This resulted in a minimal decrease in error metrics. This change is described in Lines 201-202

13. The authors specify units (kt, ...) in Table 1, but GBDT models like LightGBM are not sensitive to scaling. This section could be revised to remove unnecessary details.

Response

The unit column has been removed from Tables 1 and 2.

14. The authors justify using LightGBM based on its performance in another paper. However, CatBoost is known to handle categorical features better, and a comparison between LightGBM and CatBoost would be a valuable addition to this paper.

Response

The investigation of CatBoost has been added to the Conclusion and Future Work section (refer to Lines 422-423 of the updated manuscript).

15. While the authors justify using LSTM based on its success in another work, the performance improvements over simpler models like GRU are not sufficiently discussed. A comparison between LSTM and GRU models would be valuable, as GRUs can sometimes offer a more efficient alternative to LSTMs.

Response

GRU models will be explored in future work (refer to Line 423 of the updated manuscript).

16. The authors use different data splits for regression and time series problems. It would have been more logical to use consistent time frames across both problems to ensure a fair comparison of results in the test set between the regression and time series models.

Response

The authors are of the opinion that the regression and time series models cannot be compared directly to each other because they serve different purposes i.e. the regression models to predict the holding time of individual flights at specific times from the TMA; and the time series models to predict the average holding time in different holding stacks in 15-minute time intervals. As such, no attempt is made to compare the two groups of models, and that is (also) why different data splits are used.

17. Why PowerTransformer was used in the regression and MaxMin scaling in the time series. This is not clear and must be clarified.

Response

PowerTransformer was initially applied to the regression and time series problems. However, in the case of the time series, MinMax scaling resulted in superior predictions compared to the PowerTransformer scaling technique. A footnote has been added at Line 300 of the manuscript.

18. Why not use a 9th Model with 4 heads, each one predicting the average delay of each holding stack in a multi-output setting? This could be included to the "future work" section.

Response

The design of such a model is described in the future work section (Lines 426-428).

19. The authors use an LSTM with 50 neurons and 4 layers, and dropout of 0.2, but they do not explain why these particular choices were made. These hyperparameters should be optimized, and the authors should explain why such architecture was chosen in the first place.

Response

The LSTM hyperparameters are now optimised using GridSearchCV as shown in Tables 5 and 8. This was also explained in Lines 290-291 and 320-326 of the manuscript. The MAE did not improve, but RMSE improved significantly.

20. The inclusion of Table 8 is redundant. The SHAP values could be better visualized in figures. Tables here are unnecessary and distract from the discussion.

Response

The tables have been removed, and instead the Beeswarm SHAP plot for each model is being shown for each algorithm side by side.

21. The explanation of SHAP values is too simplistic. The authors should provide more context for readers who may not be familiar with these techniques, explaining how they contribute to the model interpretation.

Response

SHAP values and plots explanations have been added at Lines 327-333 of the manuscript.

22. The readability of Figure 9 is poor. The authors should try using a line plot instead to improve the clarity of their results.

Response

Figure 15 (updated manuscript) is a line plot. To improve readability, the figure has been updated to plot the 6-hour rolling average of the Original and Predicted holding times of LAM.

Conclusion and Future Work

23. Rather than building separate models for each look-ahead time, the authors could integrate this feature into a single model. This would improve generalization and make the implementation easier.

Response

It would be ideal to have a model that is capable of predicting the holding time at any point in time during the flight. However, it was found that the farther the flight is from the TMA (e.g. Model 3 vs. Model 1), the larger the error metrics. Lines 426-428 of the 'Future Work' section have been updated.

24. The authors mention future comparisons between encoding techniques, but they fail to do this in the current paper. This is a relatively easy experiment to conduct.

Response

The authors experimented with integer-encoding and OHE at the beginning of the study and it was found that OHE gives superior results. As explained in Lines 423-424 of the manuscript, in the future alternative encoding methods, such as embeddings, could be implemented and compared to each other.

5.3 Response to reviewer 3

1. The gap you identified is the lack of ML methods to predict holding times?

Response

Yes, as stated in Lines 104-108, a lack of methods in general (not just ML) was found concerning holding time prediction.

2. Why ML methods might be more adapted to the task than previous methodologies?

Response

In general, ML methods are better at making predictions than traditional statistical methods, and they can detect complex trends/patterns in large datasets. Also, the studies mentioned in the 'Related Works' section are evidence of the suitability of ML to related ATM tasks.

3. Is your paper purely exploratory: the main goal is to check if ML methods have the potential to overcome previous work?

Response

The goal of this paper is purely exploratory: to investigate if ML methods (specifically LightGBM and LSTMs) are suitable for predicting holding time.

4. Why LSTM and LGBM seem to be the obvious choice?
5. Previous work in other field already demonstrated the reliability of those architectures for similar problems?

Response

They are considered to be the obvious choice for this study based on the results obtained when these methods were applied to similar problems. Lines 240-244 provide several papers supporting this choice. However, it is proposed that other methods will be investigated in the future, such as CatBoost and GRUs, as stated in the 'Conclusion and Future Work' section.

6. When detecting a holding pattern, you compare the heading for two consecutive observations separated by 30s. If the heading is more than 20 degrees, then the holding count begins. However, the proper holding pattern could have been initiated anywhere between those two observations, which might lead to an error up to 30s between the real holding pattern start, and the detected one. Thus, from one sample to another in your training dataset, the detected start can be either the very beginning of the holding, or up to 30s in the holding. The same consideration holds for the end, which can lead to discrepancies up to one minute overall. Thus, your model performance depends on your ML model as much as on your detection algorithm. It might be worth to explain further the validation of your model (e.g. comparing the calculated times with the one manually labeled), and also mention it in the limitations if necessary.

Response

Yes, the difference between the actual and calculated holding time could be up to 1 minute due to the sampling interval used. This error is noted in Line 167 and could be reduced by using a higher sampling rate (e.g. once every second).

The detection algorithm was validated by comparing the holding times of a small number of flights - as calculated by the algorithm - with the holding times as calculated manually. A total of 30 flights were compared with this method with an error of 1 minute.

In addition, we compared our detection algorithm with that of the Traffic API. More information about this is provided in Lines 170-175.

7. It might be beneficial to remind the size of each constructed training datasets.

Response

Lines 188-190 of the manuscript have been included to remind the reader about the size of each dataset.

8. According to figure 9, the model seems to constantly underestimate the holding time. Do you have an idea about the reason?

Response

The following are possible reasons:

1. Imbalance in the training dataset, where shorter holding times are more frequent, leading the model to generalise toward those lower values.

2. Bias–variance tradeoff. A model with high bias pays little attention to the training data and oversimplifies the problem, leading to systematic errors—like always underpredicting. On the other end a model with high variance learns the noise in the training data and may overfit, leading to poor generalization. Minimizing errors caused by oversimplification and excessive complication requires finding the right balance or tradeoff between the two.

3. It could be that certain parameters are not being captured by our ML model which is, in turn, leading to an underestimation of holding time.