Predicting Air Traffic Congested Areas with Long Short-Term Memory Networks

Loïc Shi-Garrier [1]   Daniel Delahaye [1]   Nidhal C. Bouaynaya [2]

[1]Ecole Nationale de l'Aviation Civile, Université de Toulouse, Toulouse, France

[2]Dept. of Electrical and Computer Engineering, Rowan University, New Jersey, USA

September 21, 2021

Content

Content

Context

- ATFCM is looking at occupancy problems: balance demand/capacity
- ATC is looking at proximity problems: ensure separation of aircraft
- ▶ Formation of "hot spots"

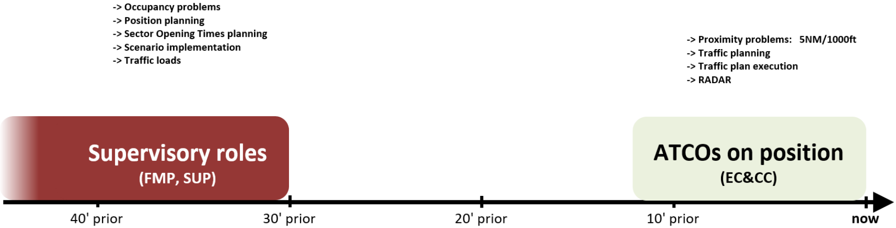


Figure 1: There is a **gap** between strategic and tactical operations

Extended ATC Planning

The EAP function main objective is to bridge the gap between ATFCM and ATC relying on:

- Automation tools
- Improved communication between local flow management position and controllers
- Information provision to help take early measures before traffic enters overloaded sectors

The EAP function is expected to reduce delays, reduce the number of ATFCM regulations, and enhance the safety.

Objectives

Two subproblems:

- Predicting congested areas tens of minutes before their formation
- Mitigating the congestion by appropriate early actions

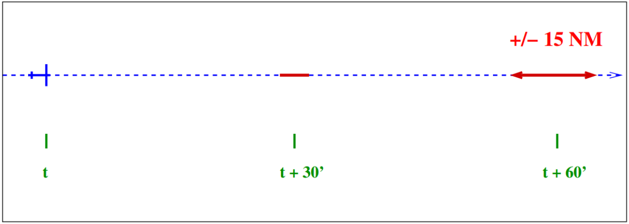
Objectives

Two subproblems:

- Predicting congested areas tens of minutes before their formation
- Mitigating the congestion by appropriate early actions

In this work, we focus on **predicting congested areas**

- Using machine learning techniques (Encoder-Decoder LSTM framework) to integrate ATFP with an intrinsic complexity metric
- In an extended time horizon of 40 minutes before crossing the sector
- Uncertainties are too high to predict which trajectories will be involved in conflicts



Methods for congestion prediction

Three existing approaches:

- Conflict detection tools
- Air traffic flow prediction: focus on aggregate models rather than individual trajectories. Recent implementations rely on machine learning to predict the aircraft count.
- Air complexity metrics

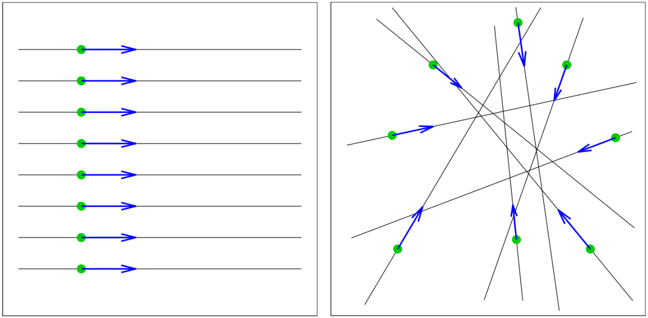


Figure 2: Different complexity for the same aircraft count

Content

Complexity metric

Congestion is defined as a situation where a set of trajectories strongly interact in a given area and time interval, leading to potential conflicts and hence requiring high monitoring from the controllers.

Air traffic **complexity** is a measure of the difficulty that a particular traffic situation will present to air traffic control.

Complexity metric

Congestion is defined as a situation where a set of trajectories strongly interact in a given area and time interval, leading to potential conflicts and hence requiring high monitoring from the controllers.
Air traffic **complexity** is a measure of the difficulty that a particular traffic situation will present to air traffic control.

Goal: to quantify the "complexity" of a traffic situation

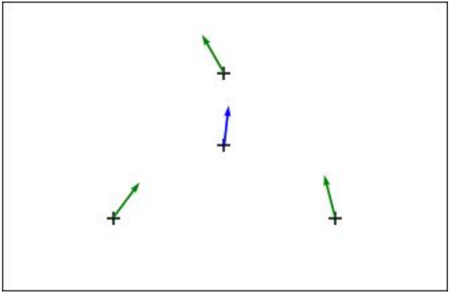
The complexity is linked with:

- The workload of the ATCO
- The conflicts probability
- The **geometry of the traffic** (convergence/divergence) and of the airspace

- We use a complexity metric based on **linear dynamical systems**

Complexity metric

Let $P = \begin{bmatrix} x_1 & x_2 & \cdots \\ y_1 & y_2 & \cdots \\ z_1 & z_2 & \cdots \end{bmatrix}$ and $V = \begin{bmatrix} v_{x_1} & v_{x_2} & \cdots \\ v_{y_1} & v_{y_2} & \cdots \\ v_{z_1} & v_{z_2} & \cdots \end{bmatrix}$

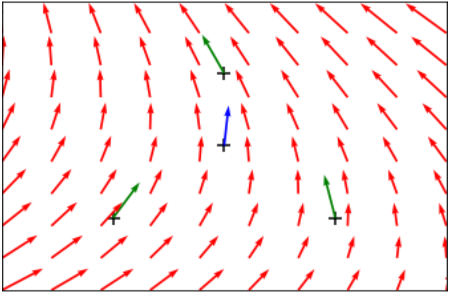


Complexity metric

Let $\mathrm{P} = \begin{bmatrix} x_1 & x_2 & \cdots \\ y_1 & y_2 & \cdots \\ z_1 & z_2 & \cdots \end{bmatrix}$ and $\mathrm{V} = \begin{bmatrix} v_{x_1} & v_{x_2} & \cdots \\ v_{y_1} & v_{y_2} & \cdots \\ v_{z_1} & v_{z_2} & \cdots \end{bmatrix}$

We consider the linear dynamical system $\mathrm{V} \approx \mathrm{AP} + \mathrm{b}$

A and b are obtained with least squares



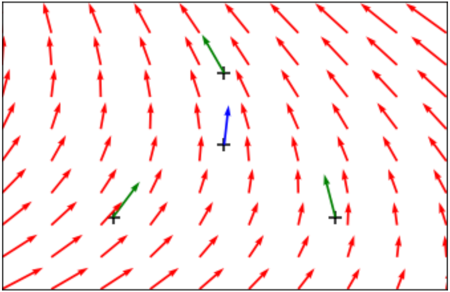
Complexity metric

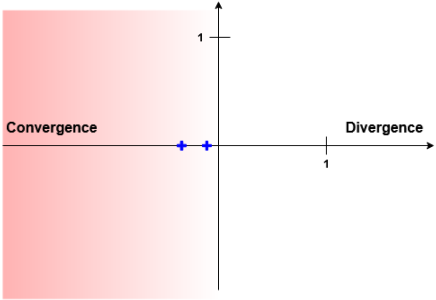
Let $\mathrm{P} = \begin{bmatrix} x_1 & x_2 & \cdots \\ y_1 & y_2 & \cdots \\ z_1 & z_2 & \cdots \end{bmatrix}$ and $\mathrm{V} = \begin{bmatrix} v_{x_1} & v_{x_2} & \cdots \\ v_{y_1} & v_{y_2} & \cdots \\ v_{z_1} & v_{z_2} & \cdots \end{bmatrix}$

We consider the linear dynamical system $\mathrm{V} \approx \mathrm{AP} + \mathrm{b}$

A and b are obtained with least squares

The **metric** is defined as $c(\mathrm{A}) = \sum_{\mathrm{Re}(\lambda(A))<0} |\mathrm{Re}(\lambda(A))|$





Content

Sequential Models

- Recurrent Neural Networks are well suited to process sequences of variable lengths, including trajectories
- Basic RNN cell: $y_t = \phi(W \cdot [x_t{}^T ; y_{t-1}^T]^T + b)$
- Many variants including LSTM

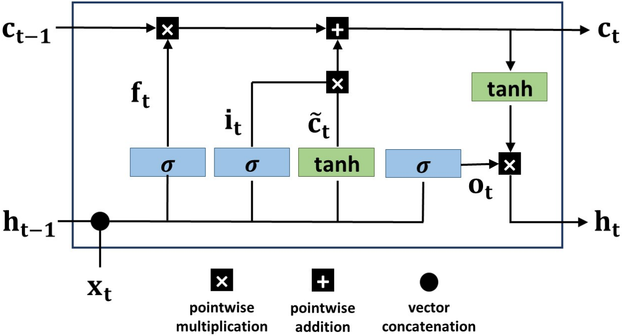


Figure 3: LSTM layer

Convolutional layer

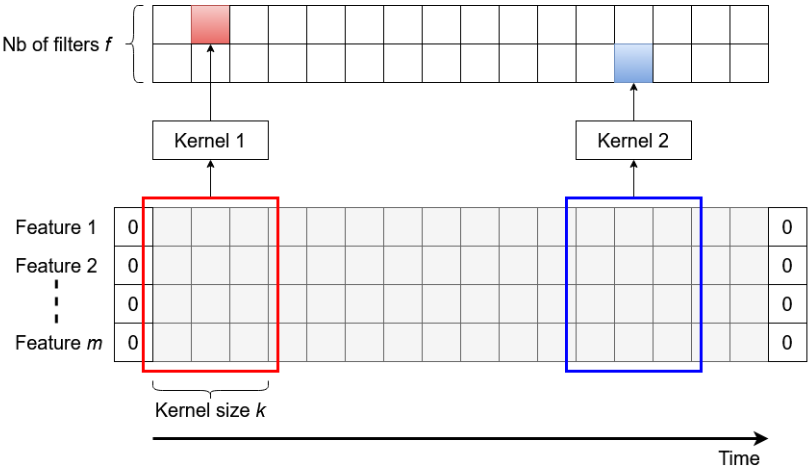


Figure 4: One-dimensional convolutional layer

Dataset

A trajectory is a sequence of aircraft states

$$state_i = [\text{timestamp, aircraft ID, latitude, longitude, altitude, ground speed, heading, rate of climb}].$$

- 8,011 simulated trajectories
- 3,025 time steps (1 time step = 15s)
- 2,434 sequences of length $t_{in} = 160$ for training
- 271 sequences of length $t_{pred} = 160$ for validation

We define a **supervised learning regression task**

Dataset

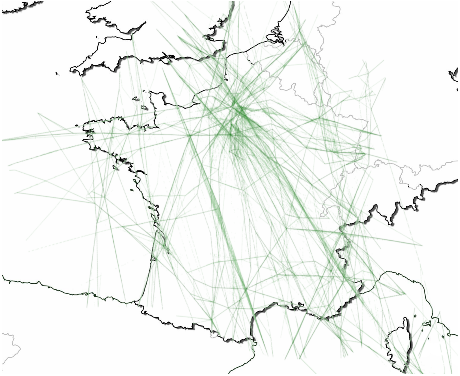
Definition of the inputs

For each time step t, there is a set of aircraft states

$state_1$; $state_2$; $\cdots$

These states are concatenated in $\mathrm{x_t} = \left[state_1 \quad state_2 \quad \cdots\right]^T$

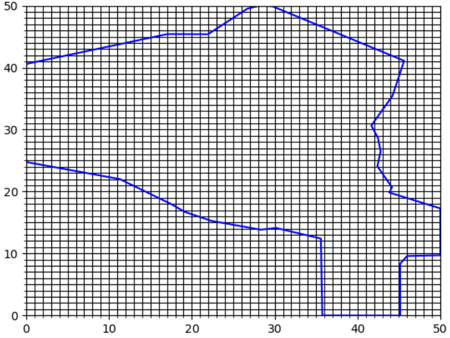
The input is defined as the sequence $X_{t_0} = \left(\mathrm{x}_{\mathrm{t_0}}, \quad \cdots, \quad \mathrm{x}_{\mathrm{t_0+t_{in}-1}}\right)$



Dataset

Definition of the target outputs

For each time step t, we define a $n \times n$ matrix $\mathrm{H_t}$
For each cell (i, j), $(\mathrm{H_t})_{i,j}$ is the maximum value of the complexity metric inside this cell (rescaled by a logarithmic function) i.e., **the worst case**



- The input $\mathrm{X_{t_0}}$ is associated with the output $\mathrm{H}_{t_0+t_{in}-1+t_{pred}}$

Content

Encoder-decoder network

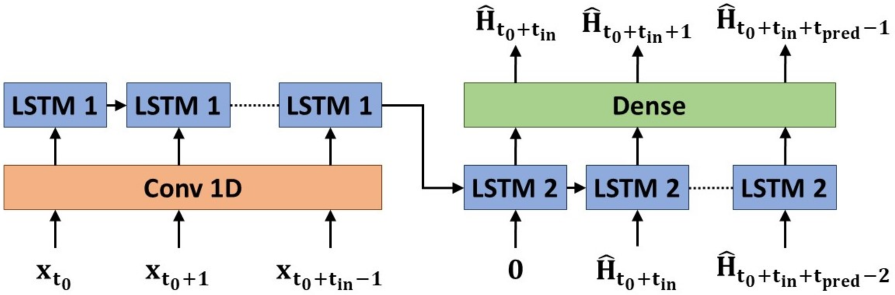


Figure 5: Encoder-Decoder model with "teacher forcing"

Full framework

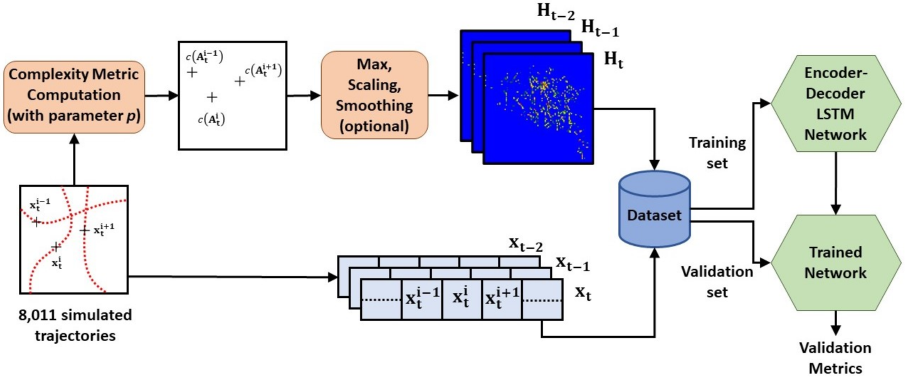


Figure 6: The dataset is composed of trajectories where x_t^i is the state vector of the i^{th} trajectory at time t. First, the complexity values are computed. Then, the dataset composed of the aircraft states and the complexity matrices is divided into training and validation sets. The encoder-decoder LSTM model is trained (resp. validated) with the training (resp. validation) set.

Content

Example of results

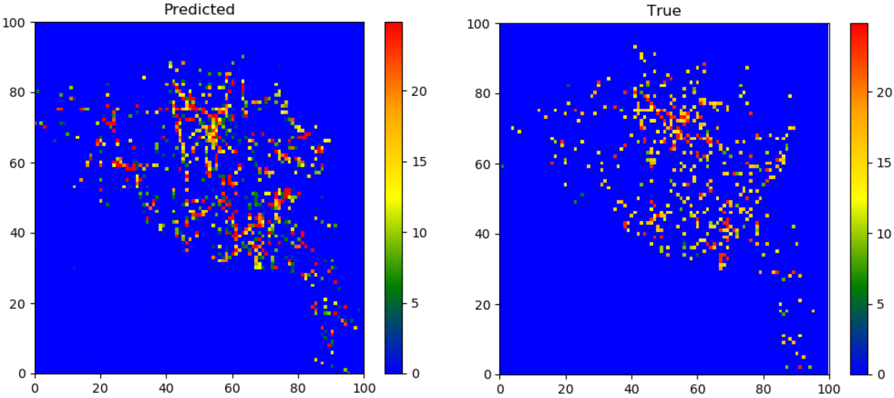


Figure 7: An example from the training set of the predicted matrix $\hat{H}_t$ (left) and the true matrix H_t (right)

Example of results

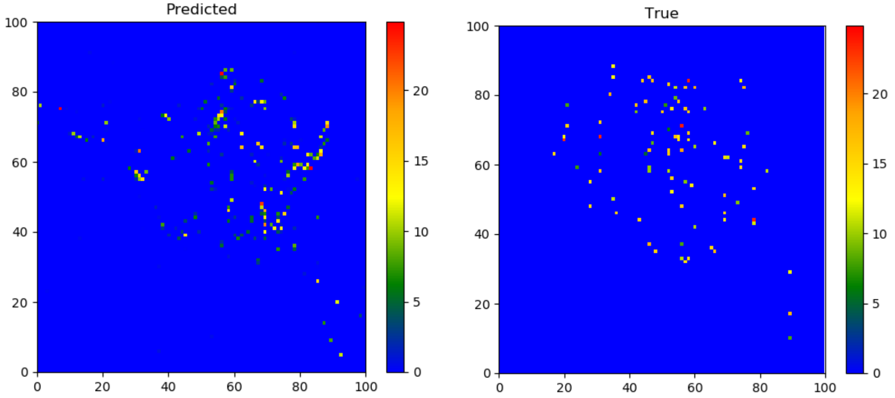


Figure 8: An example from the validation set of the predicted matrix $\hat{H}_t$ (left) and the true matrix H_t (right)

Preprocessing with Gaussian smoothing

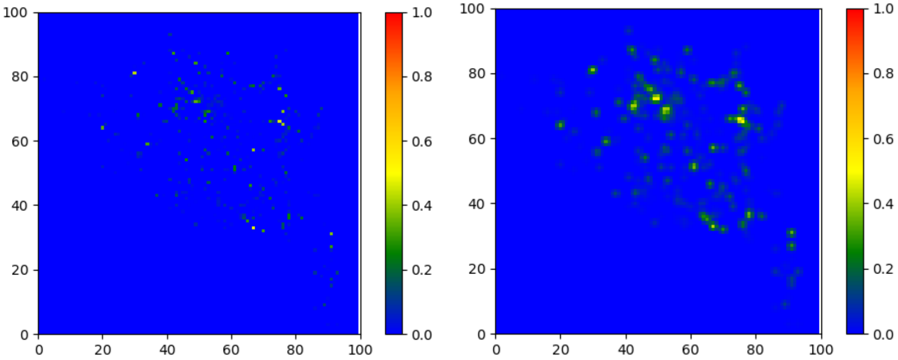


Figure 9: Original H_t (left) and smoothed H_t (right)

Results

Validation set: 271 examples (2,710,000 values)

182,982 non-zero values with mean = 0.4

678,240 non-zero values with mean = 0.08

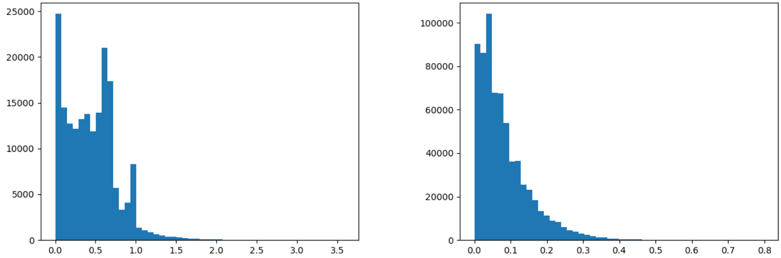


Figure 10: Distribution of non-zero absolute errors over the validation set with default model (left), and with Gaussian smoothing (right)

Example with Gaussian smoothing

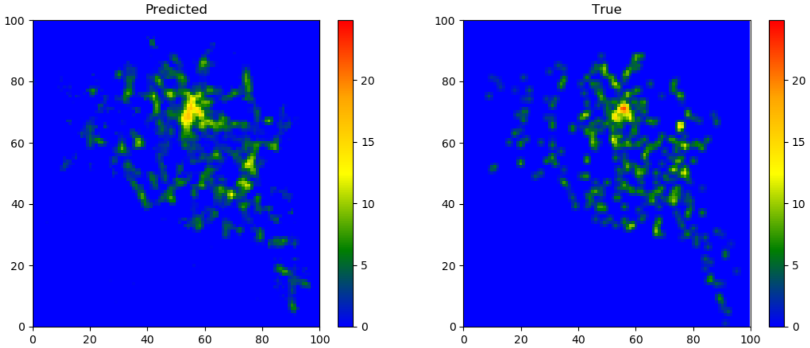


Figure 11: An example from the validation set of the predicted matrix $\hat{H}_t$ (left) and the true matrix H_t (right) with the smoothed data

Conclusion

In this work, we aim at predicting the air traffic congestion in a time horizon of 40 minutes by combining:

- Air Traffic Flow Prediction methods
- A definition of congestion relying on an intrinsic complexity metric based on a linear dynamical system
- An Encoder-Decoder LSTM network

Conclusion

In this work, we aim at predicting the air traffic congestion in a time horizon of 40 minutes by combining:

- Air Traffic Flow Prediction methods
- A definition of congestion relying on an intrinsic complexity metric based on a linear dynamical system
- An Encoder-Decoder LSTM network

Potential applications and future work

- Information provision to the flow management position to improve decision making
- Input to a mitigation system to bridge the gap between the FMP and ATC through the EAP concept
- Quantification of the uncertainty in the predictions
- Refinement of the model and training with larger dataset, historical data, and new features (e.g., weather data)
- Predict the estimated time of overflight to selected waypoints and use these predictions to compute a measure of complexity