ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

DATA-DRIVEN DELAY ESTIMATION AND ANOMALY DETECTION: A STUDY ON EUROPEAN AND TURKISH AIR TRAFFIC

M.Sc. THESIS

Muhammet AKSOY

Department of Aeronautical and Astronautical Engineering

Aeronautical and Astronautical Engineering Programme

MAY 2023

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

DATA-DRIVEN DELAY ESTIMATION AND ANOMALY DETECTION: A STUDY ON EUROPEAN AND TURKISH AIR TRAFFIC

M.Sc. THESIS

Muhammet AKSOY
(511201136)

Department of Aeronautical and Astronautical Engineering

Aeronautical and Astronautical Engineering Programme

Thesis Advisor: Dr. Emre KOYUNCU

MAY 2023

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

VERİ GÜDÜMLÜ GECİKME TAHMİNİ VE ANOMALİ TESPİTİ: AVRUPA VE TÜRKİYE HAVA TRAFİĞİ ÜZERİNE BİR ÇALIŞMA

YÜKSEK LİSANS TEZİ

Muhammet AKSOY
(511201136)

Uçak ve Uzay Mühendisliği Anabilim Dalı

Uçak ve Uzay Mühendisliği Programı

Tez Danışmanı: Dr. Emre KOYUNCU

MAYIS 2023

Muhammet AKSOY, a M.Sc. student of ITU Graduate School student ID 511201136 successfully defended the thesis entitled "DATA-DRIVEN DELAY ESTIMATION AND ANOMALY DETECTION: A STUDY ON EUROPEAN AND TURKISH AIR TRAFFIC", which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Dr. Emre KOYUNCU** Istanbul Technical University	.....
Jury Members :	**Prof. Dr. İbrahim ÖZKOL** Istanbul Technical University	.....
	Dr. Cengiz PAŞAOĞLU General Directorate Of State Airports Authority	.....

Date of Submission : **20 April 2023**
Date of Defense : **18 May 2023**

To my family and to the one whom I grew up with,

FOREWORD

This study is important in the context of showing the effectiveness of deep learning and reinforcement learning algorithms on commercial and civil aviation applications. The fact that commercial aviation produces a vast amount of data per year means there is too much value waiting to be harnessed and certainly there will be more and more AI applications in this field each year. It is a fascinating field where the pinnacle of physical and digital worlds meet and I feel happy and fulfilled to be contributing (even the tiniest) to this area. I would like to thank my advisor, Dr. Emre Koyuncu, for giving me the opportunity to study this multi-disciplinary field and for his continuous mentorship. I would like to thank my friends and my family for their good wishes and their unconditional support. Finally I thank Kübra, for being a faithful and colorful companion throughout my university life.

May 2023

Muhammet AKSOY
Aeronautical Engineer

TABLE OF CONTENTS

ABBREVIATIONS

SIS	**:** Susceptible - Infected - Susceptible
SIR	**:** Susceptible - Infected - Recovered
OD	**:** Origin - Destination
RL	**:** Reinforcement Learning
DL	**:** Deep Learning
A2C	**:** Advantage Actor Critic
FAA	**:** Federal Aviation Administration

SYMBOLS

δ	: Recovery Rate
Δ	: Matrix with Recovery Rates on Main Diagonal
β	: Infection Rate
p	: Probability of Infection
$\dot{p}$	: Change Rate of Probability of Infection
d	: Amount of Delay In Minutes
G	: Directed Graph
A_{ij}	: Flow Matrix Among Airports
π	: Reinforcement Learning Policy
λ	: Eigenvalue of a Matrix

LIST OF TABLES

LIST OF FIGURES

Page

DATA-DRIVEN DELAY ESTIMATION AND ANOMALY DETECTION: A STUDY ON EUROPEAN AND TURKISH AIR TRAFFIC

SUMMARY

Air traffic networks represent highly complex and interconnected physical systems. Unlike other transportation networks, air traffic is very heavily regulated and physically constrained. Although the airways and airspaces are somehow more flexible compared to land based transportation systems, the fact that aircrafts can only positioned on and operated by airports make them quite dependent on the operations of the airports. Air traffic is regulated to ensure safety, while also maintaining the throughput of travel from one location to another. While these regulations does a decent job on keeping the air travel safe and systematical, they fall short when there are disruptions among the network that hinders the air traffic. There are numerous reasons for disruptions in air transportation; weather conditions, accidents, capacity constraints, personnel strikes etc. Yet their negative effect to the air traffic is mostly the same: introducing delays. Due to the connected nature of the air traffic and airports, when a delay generating event occurs at one place, the other members of the network could experience the similar effects, if not at a larger degree. This delay propagation means there is a ripple effect through the network which can snowball the delay generations and cause very large congestions. To relieve the effects of delay generating events, air traffic federators regulate the air traffic in a reactionary way. This may include reducing the capacity on certain airports or airways, giving NOTAMs, holding aircrafts on the ground or in the air (with hold patterns). Since all these actions are *reactionary*, they are set in place after the delays already propagates through the network since it is trivial to asses and quantify the propagations in a large and complex network system. This study hypothesis that if the air traffic network can be modeled so that the propagations can be accurately calculated, it becomes possible to take proactive actions instead of reactive ones. Proactive actions are significantly more important when there is a risk of snowballing and propagation. It allows to take action when the ill effects are still contained on fewer members with smaller intensities. This paves the way for a more effective and less costly approach. Hence, the study proposes a method with 3 main parts; first one is to model the air traffic network so that propagations can be quantified, second one is to estimate the parameters of this model to keep a short-sighted vision into the upcoming network state and third one is to come up with a comprehensive action generating model to find optimal proactive actions that can keep the delay spreading at minimum and improve system resiliency. The air traffic modeling part is done via adopting compartmental model from epidemiology. This model explains the tranmission of disease within a population. When it is applied to the physical network system, instead of disease and humans, the delay amount and aircrafts is used. Additionally with the meta population model, instead of considering aircrafts one by one, airports can be used as they are focused points of aircraft populations.

By linking transmit rate to the flight frequency between airports and the recovery rate to the delay handling characteristics of the airport, The parameter estimation part is done by calculating the historic recovery rates of the airports and then using deep learning inference to predict the next time step's recovery rates. The other parameters of the air traffic model, such as the traffic flow, is already known before hand (flight plans). Therefore through the estimation of recovery rate the network state of the upcoming states can be accurately predicted. This prediction can then be fed to the action generating algorithm to make the most informed decision. The action generating algorithm therefore must fundamentally be a deterministic state to action mapper. Reinforcement learning approach is utilized to train this state to action mapper to make it capable of generating optimal decisions under a sufficiently large spectrum of conditions.

The final part of this study concerns with anomalous flight detection in air traffic as these types of flights are one of the sources of disruptions in an air traffic network. Although flight paths naturally diverge from one another, they still adhere to a set of patterns that have been tested in various environments and are optimized for them. These patterns may or may not be simple, depending on a number of factors, such as airspace use, the cognitive complexity of controllers, the weather, and NOTAMs. It is a challenging task to accurately classify flights just by their trajectories into a desired set of categories based solely on its statistical properties because of the high variance. For this purpose, the study incorporates a statistical approach that takes into account the time-based characteristics of the flight trajectories to determine whether they are abnormal or not. This statistical method with LSTM autoencoders makes it possible to train the model with historical data and quickly predict the flight class, taking into account the time-based characteristics of a flight trajectory. LSTM autoencoders can capture the class of a flight with relatively shorter time windows (16 second intervals). Therefore the air space can be periodically sweeped for anomalies while the network model and action algorithm runs in parallel. The obtained results demonstrate that the suggested architecture is quite capable of classifying abnormal flight trajectories as it successfully detects simulated fighter aircraft trajectories in airspaces with high commercial flight density. With the applications of deep learning and reinforcement learning, this whole methodology ensembles is largely data-driven, however the introduction of the compartmental model from epidemiology lays out a strong and accurate mathematical formula to support these data-centric approach. As the results suggests, The whole network's resiliency, i.e. its ability to keep delays from spreading and absorbing them, significantly increases when the optimal actions are reflected on the parameters. Additionally with the help of unsupervised learning, anomalous flights are also detected and represented as a disruption source to the network. Possible biases and shortcomings due to the data-driven approach is recognized throughout the study yet the overall method is deemed to be of significant importance in terms of managing resiliency through air traffic networks.

VERİ GÜDÜMLÜ GECİKME TAHMİNİ VE ANOMALİ TESPİTİ: AVRUPA VE TÜRKİYE HAVA TRAFİĞİ ÜZERİNE BİR ÇALIŞMA

ÖZET

Hava trafiği ağları oldukça karmaşık ve birbirine bağlı fiziksel sistemleri temsil eden ağlardır. Diğer ulaşım ağlarının aksine, hava trafiği çok sıkı bir şekilde regüle edilir ve fiziksel olarak daha fazla kısıtlamaya sahiptir. Uçuş yolları (airway) ve hava sahaları, karayolu taşımacılığı sistemlerine kıyasla daha fazla esneklik getirse de, uçakların yalnızca havaalanlarında konumlanabilmesi ve yalnızca buralarda uçuş operasyonlarına katılabilmesi aslında fiziksel olarak çok limitli bir ağ oluşmasına neden olur. Hava trafiği, güvenliği sağlamak ve güvenlik sağlanırken de seyahat hacmini mümkün olan en üst düzeyde tutma amacıyla regüle edilir. Bu düzenlemeler hava seyahatinin güvenli ve sistematik bir şekilde yapılmasını sağlarken, hava trafik ağında oluşan herhangi bir bozuntuyu engellemede bu düzenlemelerin pek de etkili olduğu söylenemez. Bu sebeple hava trafiğinde hava muhalefeti, kapasite sınırları gibi gecikmelere sebep olan bozuntular oldukça büyük çaplı tıkanmalara ve aksamalara sebep olabilir. Hava trafik ağlarındaki bu bozuntu ve aksaklıkların nedenleri arasında hava koşulları, kazalar, kapasite kısıtlamaları, personel grevleri vb. faktörler yer alır. Ancak, olumsuz etkilerin ortaya çıkardığı sonuç çoğunlukla aynıdır: uçakların kalkış veya inişlerinde gecikmelere sebep olurlar. Hava trafiğinin ve havaalanlarının birbirine çok yakından bağlı doğası nedeniyle, bir yerde gecikmeye sebep olan bir olay meydana geldiğinde, ağın diğer noktalarındaki üyeler, yani başka havalimanları, bu gecikmeyi aynı şekilde hatta daha büyük ölçüde hissedebilir. Bu olaylar sebebiyle, meydana gelen gecikme bir dalga gibi ağda yayılıp, ek olarak kartopu etkisiyle de ilk oluştuğu ana kıyasla çok daha fazla büyüyebilir. Hava trafiğini düzenleyen kuruluşlar, bu tip gecikme oluşturan olayların etkilerini hafifletmek için olaylar meydana geldikten sonra bazı düzeltici aksiyonlar alırlar. Örneğin belirli havaalanlarının veya hava yollarının kapasitesini azaltmayı, NOTAM'lar vermeyi, uçakları yerde veya havada bekletmeyi (holding patterns) içerebilir. Tüm bu işlemler "tepkisel" olduğu için, gecikme yaratan olaylar meydana geldikten sonra işleme koyulurlar, çünkü bu geniş ve kompleks sistemde gecikmelerin ağ içerisinde nasıl yayılabileceğini isabetli şekilde hesaplamak matematiksel ve işlem yükü açısından maliyetli bir hesap gerektirmektedir. Bu çalışmada kurgulanan hipoteze göre, hava trafiği epidemiyolojiden adapte edilen bölmeli hastalık yayılım modeli ile modellenebilir, parametreleri gelecek zaman adımları için tahmin edilebilir, daha sonra da bu tahmin edilen parametreler kullanılarak optimal aksiyon üretebilecek bir model oluşturulabilir.

Hava trafiğinin bölümlü model ile modellenmesinde, meta popülasyon modeli denilen yaklaşım benimsenmiştir. Normal şartlarda bölümlü model yaklaşımında bölümler popülasyonu oluşturan bireyleri temsil etmektedir. Hava trafik ağını oluşturan bireyler

de bu yaklaşımda uçakların kendisine tekabül etmektedir. Ancak meta popülasyon yaklaşımı ile, tüm uçaklar münferit olarak değerlendirmek yerine, uçuşların yoğunluğu ve frekansları olasılıksal ağırlık olarak temsil edilip, "popülasyon üstü popülasyon" oluşturularak havalimanları hava trafiği modeli popülasyonun en ufak elemanı olarak temsil edilebilir. Bölümlü modelde hastalığın, ya da bu çalışmadaki uygulamasıyla gecikmelerin yayılımını açıklayan bir dizi diferansiyel denklem bulunmaktadır. Bu denklemlerde kullanılacak parametreler; enfeksiyon oranı, iyileşme oranı ve meta popülasyon yaklaşımıyla elde edilen mevcut hastalık oranıdır. Bu modeli hava trafik ağına uyarlarken, enfeksiyon oranı gecikme etkisinin bir havalimanından diğer havalimanına aktarılma olasılığını temsil eder. Fiziksel karşılığında bu olasılık, bu iki havalimanı arasındaki uçuş sayısı ile ifade edilir. Örneğin toplamda 100 uçuş veren bir A havalimanının, belli bir periyotta 20 adet uçuşu B havalimanına oluyorsa, A havalimanın B havalimanına olan enfeksiyon oranı %20 olarak ifade edilir. Buna göre bir havalimanında, spesifik bir OD çifti sabit tutularak uçuş sayısı arttırılırsa, o OD çifti özelindeki enfeksiyon oranı düşecektir. Havalimanı toplamda ne kadar uçuş veriyorsa, OD çiftlerinden birine düşen enfeksiyon oranı o kadar azalcaktır. Fiziksel olarak yorumlandığında, büyük ve yoğun trafik alan havalimanları operasyonları geniş olduğu için herhangi bir bozuntuyu ve gecikme durumunu yüksek bir oranda bir başka havalimanına aktaramaz denilebilir. Ancak kısıtlı sayıda OD çiftine hizmet veren bir havalimanında yaşanan bir gecikme, yüksek bir oranla diğer havalimanlarına aktarılabilir. Mantıksal olarak, hedef havalimanı başlangıç havalimanının operasyonlarında ne kadar fazla yer tutuyorsa, enfeksiyon riskinin yani gecikmeyi hedef havalimanına aktarma riskinin o kadar büyük olduğu söylenebilir. Dolayısıyla enfeksiyon oranı her havalimanı için ayrı ayrı hesaplanan, o havalimanında operasyon gösteren OD çiftlerine ve toplam uçuş sayısına bağlı matris formunda bir parametre olmaktadır. Enfeksiyon oranı 0 ile 1 arasında bir olasılığı temsil ettiğinden dolayı, havalimanının hizmet verdiği her OD çiftindeki frekans sayısı toplam uçuş sayısına bölünerek normalize edilir ve bu sayede uçuş sayılarıyla enfeksiyon oranı birbirine bağlanmış olur. Tabii ki uçuş planları ve dolayısıyla enfeksiyon oranı gibi parametrelerin ifade edilebilmesi için zamansal olarak bir ayrıklaştırma yapılması gerekmektedir. Havacılıktaki yetkin kuruluşlarca iniş veya kalkış zamanlarındaki 15 dakikalık artı veya eksi kayma bir gecikme olarak kabul edilmemektedir. Dolayısıyla bu çalışmada da ayrık zaman pencerelerinin 15 dakika olarak seçilmesi uygun görülmüştür. Buna göre hava trafik ağındaki bir t zamanında bir havalimanının enfeksiyon oranı parametresine bakılmak istendiğinde, 15 dakikalık süre içerisinde kalkacak veya inecek olan uçakların bilgisiyle oluşturulmuş bir enfeksiyon oranı matrisi ya da trafik akışı matrisi görülmüş olur. Bölümlü modelde diğer bir önemli parametre olan iyileşme oranı, enfeksiyon oranının aksine 0 ile 1 arasında olmak zorunda değildir. Negatif veya pozitif herhangi bir skalar değere sahip olabilecek bu parametre aslında havalimanlarının gecikmeleri ne kadar iyi idare edebilecekleriyle belirli seviyede orantılı olan bir metriktir. Bu metrik havalimanlarının gecikmeleri idare etme performansını temsil etmesine rağmen, herhangi bir operasyon verisine bağlı şekilde değil tamamen bölümlü modelin diferansiyel denkleminde iyileşme oranının yalnız bırakılmasıyla hesaplanmaktadır. Daha önce bahsedildiği gibi bu denklemdeki 3 parametreden, mevcut gecikme oranı ve enfeksiyon oranları bilinmektedir, bu sayede iyileşme oranı hesaplanabilmektedir. Mevcut gecikme oranı

ve enfeksiyon oranı, çalışmada kullanılan verisetinde mevcuttur. Bu sayede tüm parametreleri bilinen, hava trafik ağına uyarlanmış bölümlü model elde edilir.

Metodolojinin ikinci kısmında ağın gelecek zaman aralığındaki durumu tahmin edilmek istenmektedir. Yani her havalimanının gelecek zaman adımındaki gecikme oranı hesaplanmalıdır. Bunun için enfeksiyon oranı ve iyileşme oranı parametreleri kullanılır. Enfeksiyon oranı zaten uçuş trafiği planıyla önceden bilindiği için, yalnızca iyileşme oranı gelecek adım için tahmin edilmelidir. Bunun için derin öğrenme metodu kullanılmaktadır. Buna göre bir derin öğrenme modeli, verisetindeki veri kategorilerini kullanarak bunlarla iyileşme oranı arasında bir ilişki kurmaya çalışır. Bu sayede modele veri beslendiğinde o verilerin ait olduğu havalimanıyla ilgili bir iyileşme oranı tahmin eder. Bu model kullanılarak ağdaki tüm havalimanları için gelecek zaman adımındaki iyileşme oranları tahmin edilir. Bu sayede gelecek zaman adımındaki gecikme oranlarını epidemik model ile hesaplamak için gereken iki parametre bilinmiş olur ve hava trafik ağının bir sonraki zaman adımındaki durumu tahmin edilmiş olur. Yapılan bu tahminle geleceğe dair bilgi edinilir. Bu bilgi aksiyon üreten modele beslenir ve modelden optimal bir aksiyon elde edilir. Bu model pekiştirmeli öğrenme metoduyla eğitilmiş bir modeldir. Onbinlerce yapay senaryo ile eğitilen bu model temelde epidemik modelin diferansiyel denklemindeki gecikme oranı değişimini olabildiğince minimuma çekmeye çalışır. Bu amaç tüm sistem için geçerli olduğu için, toplamdaki gecikme oranı düşecekse bazı havalimanların gecikmelerinin artmasına müsaade edilebilir. Pekiştirmeli öğrenme ile eğitim sırasında, mevcut verisetindenden referans alınarak, rastgele aksiyonlar denenir ve bu aksiyonların ağda iyileşmeye sebeb olup olmadığı kontrol edilir. Bu iyileşme epidemik modeldeki diferansiyel denklemlerin lineerleştirilip, özdeğerlerinin hesaplanmasıyla bulunur. Gecikme oranıyla çarpım halinde olan matrisin en büyük özdeğerinin reel kısmı olabildiğince minimize edilmeye çalışılır. Aksiyon sonrasında bu değer artıyorsa sistem stabilitesi kötüye gitmiş, azalırsa da iyiye gitmiş demektir. Bu geri beslemeyle pekiştirmeli model eğitilir ve bu hedef değeri düşüren aksiyonlara yüksek ödül atanır. Bu sayede pekiştirmeli öğrenme modelinin iç parametrelerine çeşitli senaryolarda en optimal aksiyonu üretebilecek şekilde ince ayar yapılmış olur. Çalışmanın son kısmı, hava trafik ağında bozuntulara sebebiyet verdiği için kuraldışı uçuşlar ve bunların otomatik tespitini sağlayan bir model ile ilgilenmektedir. Uçuş rotaları birbirinden çok farklı olabilse de, bazı belli başlı kurallar ve örüntüler çerçevesinde, uçuş verimliliği için optimize şekilde oluşturulmaktadırlar. Bu kurallar ve örüntüler havasahası kullanımı, kontrolörlerin bilinç düzeyi, hava muhalefetleri NOTAMlar gibi birçok faktöre bağlı olarak basit ya da kompleks olabilir. Tüm bu ortam içerisinde uçuşların parametreleriyle, yani sadece istatistiki bilgilerle, sınıflandırma ve anomali tespiti yapılması oldukça zorlu bir problemdir. Probleme çözüm olarak bu çalışmada, uçuşların zaman temelli karakteristiklerini dikkate alıp, herhangi bir kuraldışılık olup olmadığını tespit edebilen bir model önerilmiştir. LSTM Otokodlayıcı mimarisi ile oluşturulan bu model nispeten kısa zaman aralıklarıyla (16 saniyelik veri akışı) bir uçuşun hangi uçak tipine ait olduğunu ve kuraldışı bir durum olup olmadığını tanıyabilmektedir. Bu sayede hava trafik ağı üzerinde çalışan diğer modellerin paralelinde, anomali barındıran uçuşları yakalayabilmek için havasahasındaki tüm uçuşları dinleyebilen bir model elde edilmiş olur. Elde edilen sonuçlara göre, önerilen algoritma mimarisi ticari uçuşların yoğun olduğu

havasahalarında askeri uçuşları, dolayısıyla havasahasında bozuntu oluşturan uçuşları başarıyla tespit edebilmektedir. Bu tez çalışması neticesinde derin öğrenme ve pekiştirmeli öğrenme sayesinde veri güdümlü olarak bir modeller bütünü elde edilmiş olur. Çalışma sonucunda epidemik modele oturtulmuş hava trafik ağı modelindeki iyileşme oranları tahmin eden derin öğrenme modeli, gelecek zaman adımındaki ağ durumu tahminine göre de optimal bir aksiyon oluşturan pekiştirmeli öğrenme modeli ve havasahasını tarayarak anomali barındıran uçuşların tespitini yapabilen bir derin öğrenme modeli olmak üzere 3 ana model elde edilmiştir. Elde edilen sonuçlara göre bu yaklaşım, çeşitli senaryolarda hava trafik ağı sisteminin gecikmelere olan direncini ve yayılıma olan stabilitesini artırmaktadır ve havasahasındaki kuraldışı uçuşları tespit edebilmektedir. Metodolojinin veri ağırlıklı olması, veri kalitesiyle ve veride bulunan eğilimlerden etkilenebilse de, epidemik modelin kullanılmasıyla en azından trafik ağının modellenmesi yükü matematiksel temele oturtulmuş olup, genel performans açısından tatmin edici performans sağlayabilen bir yaklaşım elde edilmiştir.

1. INTRODUCTION

Air traffic networks are highly infrastructure constrained just like most of the other transportation networks, road travel being the obvious exception. Due to the lack of freedom of where to depart and arrive physically, all commercial aircrafts must be subject to a set of rigorous physical constraints and operational regulations. The number of aircrafts flying over an airspace at a specific time can be more flexible in terms of capacity, but there can only be so many flights arriving or departing across a region at a specific time due to runway capacities. It is, therefore, of utmost importance to regulate the traffic across the whole network, where each flight affects one another. The fact that there are a limited number of aircrafts and airports means that in case of large enough disruption that causes a primary delay, this dependency creates a ripple effect that propagates the effect to other flights and airports which generate reactionary delays. Average departure delay for all causes is reported as 9.2 minutes by [1] on European air traffic network, of which 40% is caused by reactionary delays. It is therefore a justified necessity to control or somehow quantify and measure the delay propagation so that appropriate actions can be set in place to manage delays. In fact, a 10% drop in delays would net over 17 billion of dollars of cost mitigation, as suggested by [2] on a study on US air traffic network, this paints a clear picture on expected returns from actions on delay reduction. Controllers and airspace federators are usually doing a fine work controlling the throughput over a region, but there are so many parties at work in this vast transportation network and it is unreasonable to expect it to run without any disturbances. Some events are foretelling, such as most weather conditions, high traffic days of the year etc. While accidents, for instance, are quite sudden and unexpected. Due to this almost infinite sources of network disruptions, it is not possible to eliminate the disrupting event altogether to prevent delays from occurring. The issue is also amplified by the strongly connected nature of airport and traffic network, which causes delays to propagate in a snowballing effect.

1.1 Purpose of Thesis

The air traffic network's ability to absorb delays and keep them from propagating and snowballing is represented by the system's resiliency. It describes the behaviour of the whole network under disruptive events, therefore poses as a single but very important metric that must be managed to give the air traffic network a set of mechanism that can diminish the results of disruptions and delay propagations. Naturally, the purpose of this thesis is to come up with a sound and generalized approach that can achieve the desired levels of resiliency over an air traffic network while also paying attention to the employability, flexibility and cost-effectiveness (in terms of computational load) of the proposed methods. The de facto solution to managing resiliency lies in regulating the air traffic to account for such delay generating events, with the aim of reducing or preventing the spreading and accumulation effects. Corrective regulations usually materialize as taking an action to reduce the capacity of an airspace or an airport, while it is quite effective putting a stop to the delay propagation, means that the economic and operational damage is already done due to reduced capacity. Both the disruption itself and the capacity reduction to stop it from spreading causes harm to the viability of the air traffic operations. It then becomes more apparent that preventive actions are more valuable than corrective actions, but it requires insight into future states of the network. To come up with a novel and effective solution with regards to the mentioned set of problems, presented thesis proposes an ensemble which consists of an epidemic network model to represent the propagation dynamics, a deep learning model for estimating future recovery rates in the epidemic model and a reinforcement learning algorithm to take preventive and cost-effective ground holding actions in a trade-off sensitive, complex solution space.

1.2 Literature Review

Quantifying propagation in a network usually involves a deterministic or a stochastic model, depending on the nature and variables across the network. Numerous studies have been carried out on a broad spectrum of network-related fields to compose a model to accurately describe the characteristics of this aforementioned ripple effect, as well as in air transportation. Study [3] uses a model called Approximate Network

Delay (AND) which consists of a stochastic and dynamic queueing engine (QE) and a delay propagation algorithm (DPA). Here, QE computes delays at each airport and feeds the DPA, then DPA updates flight schedules and demand rates at all airports according to computed delays. As its name suggests, DPA is an algorithm therefore it does not model the delay propagations, rather, it determines if there is a large enough primary delay then processes the flights before that delay and traces the following legs of the delay generating flight, which results in updated airport demand profiles. The model does not include information from airline/airport reactions and does not account for different delay sources, making it a frail model that is unsuitable for comprehensive network modeling. Study [4] uses Random Forest algorithms to predict departure delays on a 2-24 hour horizon in the future. It aims to predict the departure delay either at an airport or on an origin-destination pair for any flight on the route. Delay bracket classification errors are about 19% while delay regression error is about 21 minutes. Although promising, the study proposes a predictor algorithm for single nodes or links in the network, while making the attempt to capture the topology of the air traffic network. The paper [5] presents a study where a delay propagation network based on the Bayesian network (BN) model was built to investigate the interdependence relationship of flight delays among US commercial airports from May to December 2019. The network analysis theory was applied to explore the macroscopic appearance of delay propagation and simulate the extent of delay propagation when different airports occur delays. The study found that the delay propagation network was characterized by a small L, and the cumulative degree distribution followed an exponential function. The ratio between in-degree and out-degree of the network indicated that large airports were more affected by downstream airports while fewer affected by upstream airports, and small airports were more susceptible to upstream delays. The study also simulated the impact of delayed airports on other airports and found that flight delays in small airports had fewer impacts on other airports compared to large airports. The paper suggests further research in building delay propagation networks of different countries and considering the temporal component of delay propagation. The study's findings support the hypothesis that graph based network models can accurately describe the physical system. Additionally the finding

that smaller airports have smaller impact on other airports in the network is inline with the findings if this study. The paper [6] compares the performance of different approaches to predicting delays in air traffic networks using operational data from the National Airspace System of the United States. The study considers three classes of models: Markov Jump Linear System (MJLS), classical machine learning techniques such as Classification and Regression Trees (CART), and three candidate Artificial Neural Network (ANN) architectures. The study shows that prediction performance can vary significantly depending on the choice of model/algorithm and the type of prediction, such as classification or regression. The importance of selecting the right predictor variables or features is also discussed. The ANN is found to be a good algorithm for the classification problem, while the MJLS model is better at predicting actual delay levels on different links and outbound delays at major airports. The study also compares the effect of temporal factors and spatial distribution of current delays in predicting future delays and shows that the MJLS model outperforms the ANN in predicting the future spatial distribution of delays. A tradeoff between model simplicity and prediction accuracy is also revealed. This study's finding is important to base off the ethodology of choosing an epidemic model for network modeling. The study suggests that even though ANNs provide good performance in classifications, MJLS algorithm was better for predicting delay levels. This is one of the reasons why an epidemic model is implemented in this thesis methodology; it is to relieve the deep learning models from representation capacity needed to describe delay propagations, by using an mathematically informed spreading model, the deep learning model can only be used for parameter estimation and can be relatively kept simple.

The prediction of delays brought on by air traffic management in traffic streams en route to congested airports is covered in paper [7]. The main focus is on the air traffic stream coming from the west and moving towards Tokyo, which is under air traffic management in a particular air sector close to the terminal area. The viability of anticipating delays using pre-departure data, like as weather forecasts and expected trajectories, is examined in the study. In order to estimate average delays and sharp increases in delays near congested zones, the research compares the precision of queueing analysis and artificial neural network models. According to the study, an

autoregressive neural network can accurately forecast both the average delays and steep increases. The topic of on-time and effective air traffic flow management (ATFM) is covered in paper [8] considering the expected increase in air traffic volume as well as the growing need for general aviation and unmanned aerial vehicle (UAV) aircraft. The authors suggest establishing a more intelligent ATFM architecture by employing the enhanced automatic dependent surveillance-broadcast (ADS-B) approach to track and monitor aerial vehicles in real-time. Using scattered ADS-B ground stations and extracted ADS-B communications, they create an aviation big data platform that can be used to track and forecast the amount of air travel between various cities. Two machine learning approaches are utilized to forecast traffic flow and the experimental results based on real-world data show that the suggested traffic flow prediction model employing long short-term memory (LSTM) can achieve superior performance, especially when out-of-distribution contributing factors are accounted for. This study is important to emphasize the effectiveness of LSTM model in air traffic modeling. The methodology proposed in this thesis also utilizes LSTM to make predictions about the recovery rates of the airports. Temporal context plays quite a big role in air traffic and LSTMs are powerful tools that can make use of it. The research [9] suggests an effective modeling approach based on the epidemic model for delay propagation in airport networks. The authors employ complex network theory to examine the long-term dynamic of airport delay and take into account the normal release rate (NRR) and average flight delay (AFD) as measures of airport delay. They discover that the propagation factor (PF) dominates the long-term dynamics of airport delays and create an integrated airport-based Susceptible-Infected-Recovered-Susceptible (ASIRS) epidemic model for the spread of air traffic delays from a network-level perspective. This model can simulate the evolution of delay propagation in airport networks. Additionally, a simulator run using empirical data from China demonstrates that the model is substantially more accurate than the probabilistic prediction method in predicting the number of delayed airports in the near future. This research is a straightforward implementation of the epidemic model on air traffic network, however it focuses on long-term dynamics and solely used for modeling the network. Contrary

to this study, the thesis aims to expand the literature by also incorporating a model for parameter estimation and a model for action generation to manage resiliency.

1.3 Hypothesis

As explained in section 1.2, there are several methods to quantify and analyze the effects and impacts of delay-generating events and the resulting delays. All methods try to somehow fit the physical system to a mathematical model without losing the ability to account for physical constraints and properties of the system. Although possible, mathematically modeling the whole network is not computationally feasible. So, assumptions and simplifications are used on all methods to varying degrees. Still, there is a common component across all approaches: the propagation model. Deterministic or probabilistic, it is required to model the propagations to effectively account for the ripple effect through the network.

Considering the literature and the data-driven tools available at hand, this thesis hypotheses that the air traffic network can be mathematically modeled, its state parameters can be estimated and optimal preventive actions can be obtained by another state to action mapping model.

To go into detail, the air traffic network model will be based on compartmental model that is widely used in epidemiology, as it is mentioned in section 1.2 it has been successfully used in modeling the physical system of air traffic network and is very capable to accurately represent the interconnected nature of delay propagations. Once the network is accurately represented, its state parameters become the defining attributes of the system. This means the network can be wholly separated in discrete temporal space. This paves the way to estimating the future parameters (future states) based on current and/or past parameters. This way, a short-sighted trend of the network can always be calculated and preventive action model can generate optimal actions based on this available information.

2. AIR TRAFFIC MODELING

Air traffic modeling refers to the process of creating mathematical or computer-based simulations that represent the movement of aircraft in the airspace system. The models are designed to simulate various factors that can affect air traffic, such as weather, congestion, and other operational constraints. Some of these mentioned factors are intertwined with each other, such as arrival or departure delays. This type of inherent dependency multiplies the complexity of the air traffic model.

The primary purpose of air traffic modeling is to help planners and operators to better understand the behavior of the airspace system under different conditions, and to develop strategies for improving its performance. In the scope of this thesis, the aim is to quantify and assess the impacts of delay generating events and delay propagations. So, the role of air traffic modeling is to extract an analyzable representation of a physical system.

In an air traffic network, delays usually occur simultaneously on different flights/airports and from different sources. Aside from the delay generating effects, the accumulated delays also propagate from one airport to another. This means there is a topological mechanism that is also contributing to the total delay in the network. Therefore it is crucial to be able to model these ripple effects in a network, especially when the network is heavily interdependent and constrained, which basically means lack of flexibility. The following sections explain the delay propagation model and how it is integrated with the physical parameters of an air traffic network.

2.1 Adopting From Epidemiology: Compartmental Model

In epidemiology, a compartmental model is a mathematical framework used to understand how infectious diseases propagate through a population. A compartmental model's fundamental principle is to categorize or *compartmentalize* the population according to the presence of illness. Population is attributed a class of "states"

and assumed to be bound to these states, meaning a member of the population can only be described with these predefined states. The use of compartmental models is not limited to epidemiology and can be applied to many other fields, including physical networks such as transportation systems, power grids, and communication networks. For example in road transportation systems, the network can be divided into compartments representing vehicles, road segments, and intersections. The movement of vehicles between these compartments is governed by a set of differential equations that take into account factors such as traffic flow, travel time, and the road capacity. In the case of air transportation instead of vehicles and road segments, there are aircrafts, airports, airways and airspaces. Additionally, physical properties such as travel time, congestions, airport capacity are also present as the contributing elements of the model.

There are several different models within the compartmental modeling framework that are used to represent different disease transmission dynamics. Two of the most commonly used models are the SIR model and the SIS model. The SIR model, divides the population into three compartments: susceptible (S), infected (I), and recovered (R). The key assumption of the SIR model is that individuals who recover from the disease become immune and cannot be reinfected. This model is typically used for diseases where recovery results in long-term immunity, such as measles or chickenpox. The SIR model is useful for predicting the impact of vaccination and estimating the overall size of the epidemic. The SIS model, on the other hand, divides the population into two compartments: susceptible (S) and infected (I). Unlike the SIR model, individuals who recover from the disease return to the susceptible compartment and can be reinfected. This model is typically used for diseases where recovery does not result in long-term immunity, such as the common cold or a physical ill-effect like delays. The SIS model is useful for predicting the steady-state prevalence of the disease and estimating the effectiveness of actions that only provide temporary relief.

When it comes to physical networks, there are varying scenarios where one model might be more fitting to use than the other one. While SIR model assumption can be more useful when modeling the impact of interventions that permanently remove nodes or elements from the network, for instance if a power generator fails and is permanently taken offline, the SIR model can be used to predict the impact of the

failure on the overall network performance, the SIS model may be more appropriate when modeling the impact of interventions that provide temporary relief, such as load shedding, traffic management measures or in this case: diminishing and absorbing delays on airports.

2.1.1 SIS model

Originally proposed by [10], this compartmental epidemic spreading model is known as Susceptible-Infected-Recovered (SIR) model. The SIS model is a modified version of SIR model in which the "Recovered" state is non-existent, therefore the subject falls into "Susceptible" state once again after the infection period has passed. SIS model is logically suited to the study of air traffic networks, since there are no biological immunities present in physical systems. The model formulates the spreading dynamics in a population and is extensively used in complex non-biological networks, as surveyed in a study by [11]. The model describes two distinct states for the individual in the population and two governing parameters. As mentioned above, these self-explanatory states are susceptible and infected. The governing parameters are the infection rate, $\boldsymbol{\beta}$ and recovery rate, $\boldsymbol{\delta}$. The former represents the rate at which the disease (or any other condition) is transmitted from one individual to another while the latter tells about the rate of recovery for the individual. Obviously, in complex networks with vast populations and complicated interaction dynamics, it is rather troublesome to try to quantify these parameters. Statistical approaches are employed on historic data to accurately measure these parameters, but to be able to use the model for prediction purposes, appropriate parameters should be obtained or predicted for future states.

2.1.2 Meta population model

The epidemic model accounts for each individual separately in a population, however the model can also be framed so that the individuals in a network can represent other populations. This type of approach is called meta-population modeling and is used to make the semantic population groups in a network more abstract, which results in less complex modeling. In the context of this study, the airports are chosen as the individual

nodes of the epidemic model, even though the delays are in fact acting on flights. A representative diagram is given in Figure 2.1.

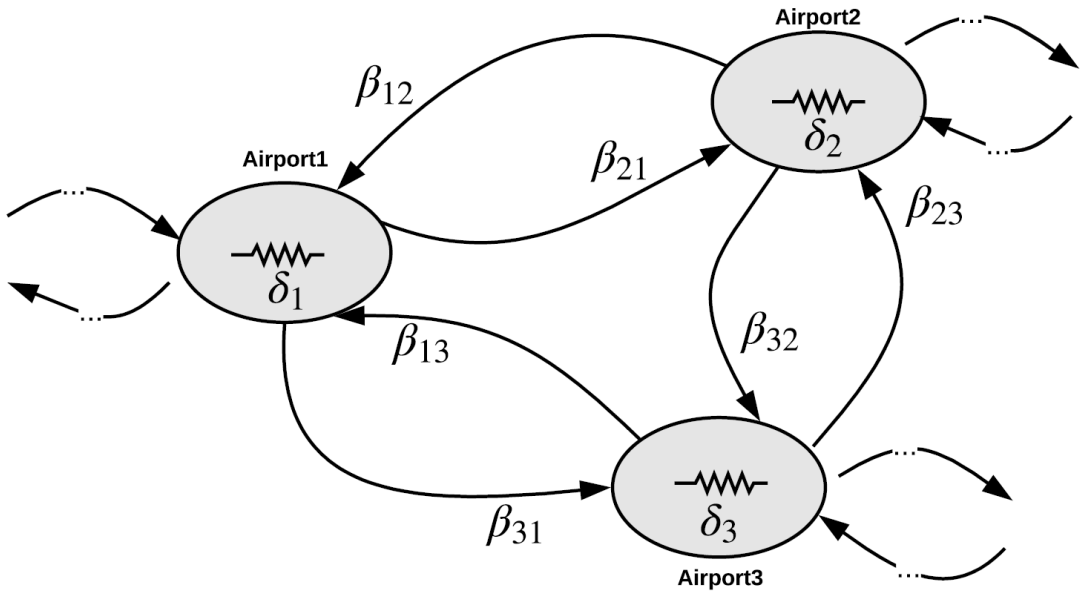

Figure 2.1 : Diagram of the meta population model.

As explained by [12] the SIS dynamics on an arbitrary network can be expressed as in equation 2.1

$$\dot{p} = -\delta_i p_i + \sum_{j=1}^{N} \beta_{ij} p_j (1 - p_i) \tag{2.1}$$

Here, $\boldsymbol{p_i}$ represents the probability of flight $\boldsymbol{i}$ being delayed. However, when meta-population approach is opted for, $\boldsymbol{p_i}$ instead represents the average flight delay per time window, such that

$$\mathrm{p_i} = \sum_{i}^{N} \frac{d_i}{N} \,/\, 15\ mins,\ d_i \leq 15\ mins,\ N \in \mathbb{Z}^+ \tag{2.2}$$

where $\boldsymbol{d_i}$ denotes the amount of delay of flight $\boldsymbol{i}$ and $\boldsymbol{N}$ denotes the number of flights. A probability value must be bounded in $\boldsymbol{p_i} \in [0,1]$, hence the normalization by 15 minutes. Both arrived and departed flights are included in the calculation of this average delay. If there were not any delay from the traffic within the time window

or the delay is negative (earlier than scheduled arrivals/departures), the probability is taken to be zero, which means no probability of infection in epidemic model terms.

2.2 State Parameters in Compartmental Model

A set of differential equations that take into consideration elements like the rate of transmission, the pace of recovery, and the size of the population control how people migrate between various compartments. As explained below, the rate of transmission will be linked with the frequency of flights among airports, the pace of recovery will be linked with a arbitrary performance metric (can also be called delay handling) while also accounting for other differences to adopt the compartmental model from epidemiology to the physical system of air traffic.

2.2.1 Recovery rate

In our meta-population approach, recovery rate refers to the rate at which the airport reduces average flight delay per hour, which is negatively proportional to $\dot{\boldsymbol{p}}_{\boldsymbol{i}}$. Considering the differential equation 2.1, the $\boldsymbol{p}_{\boldsymbol{i}}$ is bounded in $[0, 1]$ and it is clear that positive recovery rates are reducing the average delay whereas negative recovery rates mean the opposite. Therefore, accurately estimating recovery rates is of paramount importance to this study, as the preventive action framework will utilize and account for the recovery rates of the airports when interfering with the ongoing traffic.

2.2.2 Graph representation

To define the infection rate, it is very convenient to create a graph representation of the air traffic network. This way, the interconnections and the topology of the network can be clearly outlined which tremendously helps to formulate the governing parameters of the epidemic model. Let G_{nxn} be a directed graph of an air traffic network that consist of n *nodes* or airports. This means there are $n^2 - n$ possible origin-destination pairs when self loops are subtracted. These OD pairs are actually the sole physical interaction among these airports that can be considered as a connection. This means for a connection between G_i and G_j to be established, there must be an existing physical flight between the two. To represent this connectivity, an adjacency matrix A_{ij} can

also be formed, such that A_{ij} represents the link from i-th airport (departure) to j-th airport (arrival). The existence of such connection would then mean $A_{ij} \neq 0$ where as the actual value of A_{ij} would give information about the *strength* of that connection. It is now clear that a metric should be chosen to assess the strength for connections that exist. For instance, flight distance, flight duration, flight count, number of direct flights etc. could all be metrics to assess connection strength. It is therefore imperative to think about a key strength metric that describes the interaction and dependency between airports as well as possible. Although very straightforward, number of flights is a good indicator of connection strength between airports, especially for delay considerations. To give an example, if the connection strength were to be measured by passenger-miles, the type of aircraft would have a greater impact compared to a pure flight count based assessment. Aforementioned quantities are all relatively isolated and it is generally possible to combine a number of these quantities to create a strength measure, at the cost of formulation simplicity and data accessibility. Considering these factors, flight frequency or number of flights in an OD pair is chosen to be the main metric for representing connectivity strength in this study.

2.2.3 Infection rate

Infection rates are bound in $[0,1]$ and they measure how strongly the delay can be propagated between airports. To cleverly implement the flight traffic flow to infection rate representation, the adjacency matrix of the network is row normalized which means $\sum_{j=1}^{M} A_{ij} = 1$ for each row i. It essentially converts an adjacency matrix at time window t to an infection matrix, just like the example below

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 2 & 3 \\ 2 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 3 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 4 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & 0.14 & 0.14 & 0 & 0.28 & 0.43 \\ 0.50 & 0 & 0.25 & 0 & 0.25 & 0 \\ 0 & 0.50 & 0 & 0 & 0 & 0.50 \\ 0.14 & 0.14 & 0.43 & 0 & 0.28 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.00 \\ 0 & 0 & 0.20 & 0 & 0.80 & 0 \end{bmatrix} \tag{2.3}$$

Note that the rows represent the departures from a specific airport so the more spread the traffic from an airport the weaker the delay transmission will be whereas airports that send traffic to fewer destinations will have strong delay transmissions. This might

sound counter intuitive at first but the fact that the delays are always represented in averages means that when the OD pairs are few in numbers, the average delays become closer to actual delays but when the OD pairs are diversified the difference between the actual delay of a flight versus the airport average can be relatively high. Hence, it makes logical sense when the transmission strength weakens when more ambiguity is present.

3. DELAY RECOVERY ESTIMATION

State estimation is a process of inferring the unknown states of a system based on available measurements or observations. In the context compartmental model, this involves estimating the parameters of nodes (or *airports*) at the next discrete time step.

Once the delay propagation model is set in place, the parameters that are fed into the model are the main source of information about the state of the network that we are examining. This means for predicting delay propagations, one only needs to predict the relevant parameters. As explained in section 2, there are two governing parameters for the epidemic model. Infection rate, β can be obtained for future timewindows because the flights are scheduled according to plans that are readily available. Recovery rate, δ on the other hand can only be *calculated* historically, by isolating δ in equation 2.1. Other than historic calculation, it must be inferred either from statistical approaches or any other heuristics, which will be explained in following sections.

The raw data used in this study is comprised of delay information for flights flown over Europe from January through July in 2018. It has a temporal resolution of seconds both in timestamps and delay information. However, the tailored dataset has resolution of 15 minutes in arrival and departure timestamps and a minute in delay information. This means that a flight that took off at 13:33 is in the bracket of 13:30-13.45 and may have a delay of 4 minutes. Then the delays of these flights are averaged to get the infection value p of the airport for a 15 minutes bracket. This puts the infection value, infection rates (traffic flow) and recovery rates in the same resolution of 15 minutes.

Not all airports are individually included in the dataset, the degree centrality of the airports (sum of departures and arrivals) follows an exponential curve as seen in Figure 3.1, which means there are a considerable number of airports that operate with relatively fewer traffic. To account for this unproportional influence of traffic, only the top 20 airports in the degree centrality ranking list is included as a unique airport. The remaining airports (and their influence) are aggregated to an arbitrary airport with

the code "XXXX". This modification means the air traffic network now consist of 21 airports. In graph theory terms, the subgraph of these aggregated airports will now be represented by a single node. This means their internal links are omitted and their external links are now directed at the new arbitrary node.

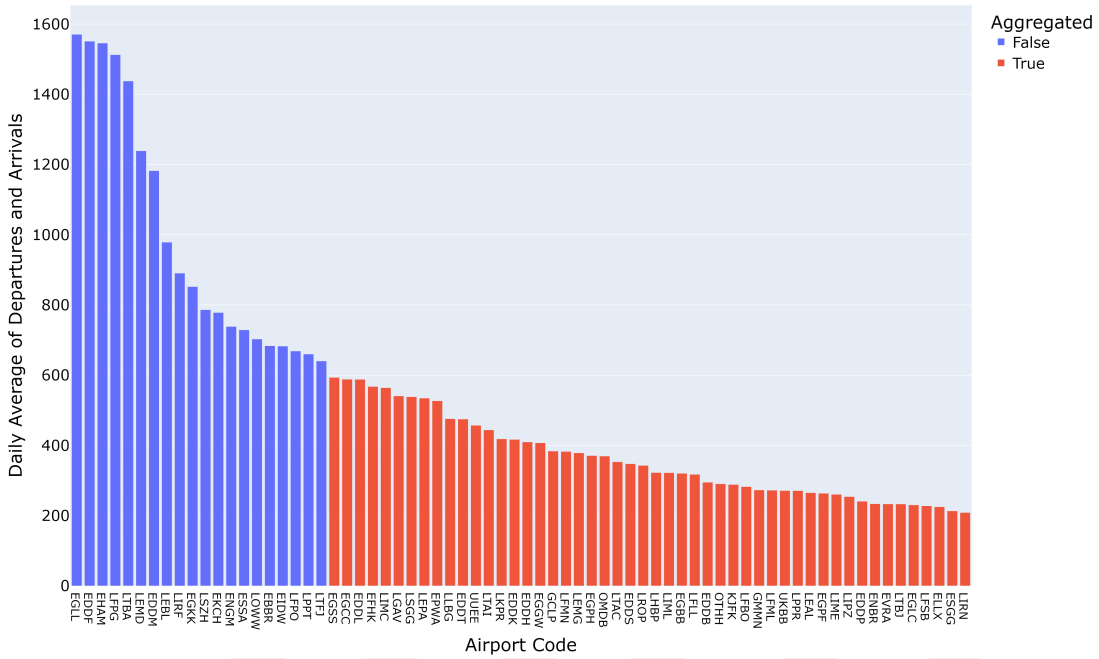

Figure 3.1 : Degree centrality of airports.

In summary, the dataset at hand has traffic plans, average delays of 21 airports and recovery rates obtained from equation 3.1. Its timeline starts from 1st of January 2018 and ends at 30th of July 2018 and all timesteps have a resolution of 15 minutes.

Table 3.1 represents an example snapshot for the aforementioned dataset. As it can be seen, temporal space is divided into 15 minute windows per day for 6 months and each airport has delay and recovery rate features. There are some instances where delay values are missing, therefore the recovery rate cannot be calculated. Missing values for *any* feature in the dataset is filled with the median value of that feature, if the value is required by another calculation flow.

3.1 Exploratory Data Analysis: Recovery Rate

Originally named in study [13], EDA has been the go to method for the beginning of any data-driven method to better understand the data at hand. As mentioned before

Table 3.1 : Example snapshot from the reference dataset.

Date	Time Window	Airport Code	Recovery Rate	Total Delay (min)	Delay Per Flight (min)
2018-01-01	0	EGLL	1.3342	20.1500	10.0750
2018-01-01	0	LFPG	2.0491	22.4833	11.2415
2018-01-01	0	EHAM	1.0747	18.3833	6.1277
2018-01-01	0	LTBA	1.1745	47.3833	6.7690
2018-01-01	0	EDDF	1.1013	15.0000	15.0000
⋮	⋮	⋮	⋮	⋮	⋮
2018-06-30	92	EHEH	1.1605	36.5333	5.2759
2018-06-30	92	LGTS	0.7941	33.6333	9.1333
2018-06-30	92	ZBAA	0.6915	35.4000	10.7214
2018-06-30	92	EGAC	1.3518	56.6333	10.7750
2018-06-30	92	UGTB	0.7109	40.0666	13.3555

recovery rates are calculated by isolating it in the equation 2.1. It is therefore expressed as in equation 3.1.

$$\delta_i = \frac{\sum_{j=1}^{N} \beta_{ij} p_j (1 - p_i) - \dot{p}}{p_i} \quad \text{where} \quad \dot{p} = p_{i_t} - p_{i_{t-1}} \tag{3.1}$$

The parameter $\dot{p}$ is known for only in historic data. Since it expresses the change of average delay for airports, it has to be extracted from a change that has already happened. For estimating purposes, however, it needs to predicted and the quality of this prediction is organically related with the accuracy of the epidemic model. In order to have a decent prediction, the data at hand should be thoroughly examined, hence the aim of the exploratory data analysis.

After the calculation of recovery rates, the first step is to look at the distribution to have a general sense on the data. In Figure 3.2, it is observed that recovery rate has a normal distribution for each of the airports, while some of the airports have multiple peaks and the median recovery rate for each airport lies in 0-0.5 range. Multiple peaks could be representing different operational behaviours, such as day time and night time operations. The airports are sorted from left to right and from top to bottom with decreasing average daily traffic. This can portray a picture on how the traffic affects the distribution of recovery rate. The peaks are higher in airports with high traffic, meaning their operations are more consistent. There is also a slight difference in the median recovery rates, where the median recovery rate of airports with smaller traffic is slightly greater. This makes logical sense as less traffic means better delay handling.

Negative recovery rates represent operations where the average delay has actually increased for a specific time window. To better understand the delay handling behaviours, it is imperative to examine the correlating factors of recovery rate.

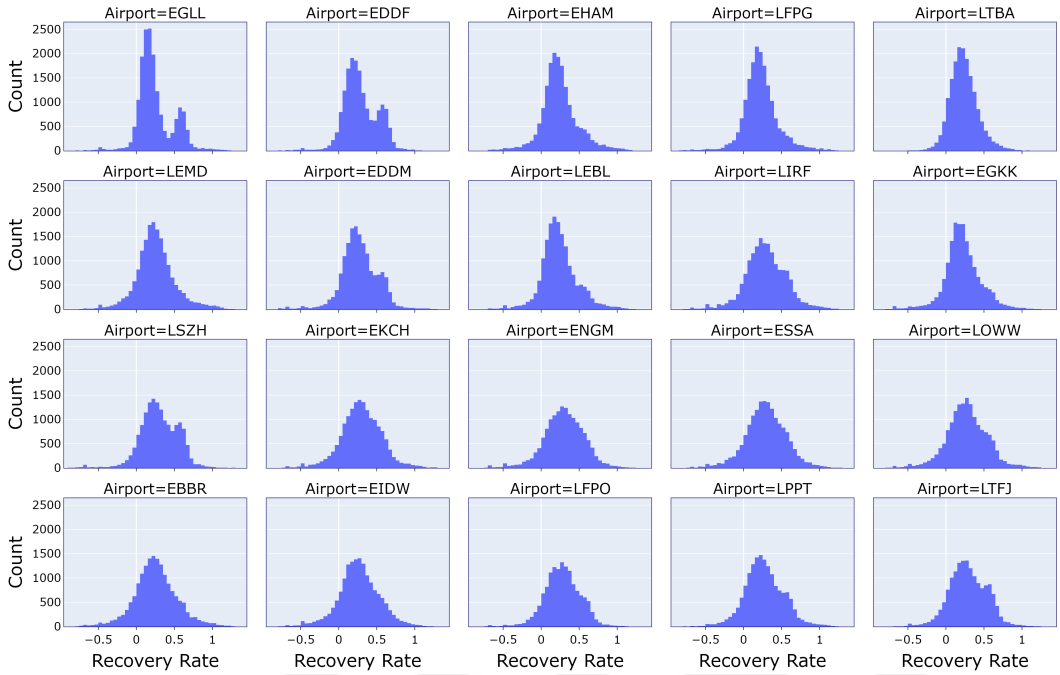

Figure 3.2 : Recovery rate distribution of airports.

3.1.1 Recovery rate clustering

Now that the recovery rate distributions are observed, further examinations can be done on contributing and influential factors on the recovery rate like a Pearson correlation [14]. Although there seems to be a hint of linear correlation for for same factors, further investigation is necessary to get a full grasp on how recovery rate is effected by external factors.

Airports and air traffic over airspaces are always monitored and appropriate measures are taken to control the throughput and the traffic. Since most of the airports are already working in high capacity, the only meaningful action is to decrease capacity on an airport to prevent delays from propagating further. As explained in the first section, depending on the root cause, certain regulations are put in place to aid with

the systematic action taking. The source events for these regulations can be rough weather conditions, operational deficiencies etc.

3.2 Deep Learning Approach

When statistical approaches fall short of a decent enough prediction performance, it may be beneficial to gravitate to more data-driven solutions. Deep learning allows computational models to represent raw data with multiple layers of abstraction to be used for prediction (or else), as explained in [15,16]. The ability to create abstractions i.e. extract latent hierarchical structures ([17]) allows for a more sophisticated way of predicting the recovery rate. Supervised learning can be summarized as learning from examples as expressed in [18], using it, a computational model can be trained on the study dataset to perform a more competent prediction on recovery rate. Conventional deep learning models are also called universal function approximators, as explained in [19]. In the scope of this study, there are two trivial functions that can serve as a hypothesis for recovery rate prediction, as expressed in equations 3.2, 3.3

$$f(\delta_t, \sigma) = \delta_{t+1} \quad \text{or} \tag{3.2}$$

$$f(\sigma) = \delta_{t+1} \tag{3.3}$$

Hypothesis function in equation 3.2 includes previous recovery rate values in the calculation of the next one. While the latter function is based only on vector σ, a set of external parameters that are either directly or indirectly linked to the recovery rate. Equation 3.2 seems to be the logical hypothesis for the reasons explained below.

As outlined in previous sections, recovery rate is calculated according to the equation 3.1. The idea is to find an appropriate vector of parameters, σ, that has reasonable influence over the recovery rate. Appropriate in this context means that the vector has just enough information without introducing unnecessary computational overhead. Recovery rate can also be expressed as in equation 3.4.

$$\delta = \delta(\dot{p}, p, \beta) \tag{3.4}$$

The expression means the recovery rate is a function of $\dot{p}, p$ and β, these parameters represent change of average delay, average delay and traffic matrix among OD pairs, respectively. A better approach, therefore, becomes to find what influences the parameters that the recovery rate depends on. While β parameter is a direct conversion of the network-wide flight plan, the delay values are influenced by a huge amount of external factors, some of which are not able to be quantified. This means that recovery rate is inherently driven by a series of external factors that cannot be completely funneled into a set of parameters to form a feature vector. Although the shortcoming of this approach is clear, using hypothesis function in equation 3.2 mitigates this to some degree, by incorporating past recovery rates into the prediction. To outline the idea, let $\sigma_t^* \in \mathbb{R}^n$ be a theoretical vector that consist of *all* external or internal parameters that have any effect on recovery rate at time step t. So that

$$\delta_{t+1} = L(\sigma_t^*) \tag{3.5}$$

where L is an unknown non-linear deep learning model under $\mathbb{R}^n \rightarrow \mathbb{R}^1$. Since it includes only quantifiable and available parameters, $\sigma \in \mathbb{R}^m$ vector accounts for less parameters than $\sigma^* \in \mathbb{R}^n$, where $n \gg m$ such that

$$\sigma_t \subset \sigma_t^* \tag{3.6}$$

There is no available data to quantify the remaining $n-m$ parameters within the access of this study, to account for this drawback, past recovery rates can be used. Instead of equation 3.5 as the hypothesis, below function can be used

$$f(\sigma_t, L(\sigma_{t-1}^*)) = \delta_{t+1} \quad \rightarrow \quad f(\sigma_t, \delta_t) = \delta_{t+1} \tag{3.7}$$

which is the same with equation 3.2. The aim then becomes finding the function f, that does a *decent* job of mapping the features (σ_t, δ_t) to the recovery rate (δ_{t+1}).

3.2.1 Supervised learning

Supervised learning is the process of deducing a mapping (f) from inputs (σ, δ) to outputs (δ) [20]. By calculating the recovery rates and putting together the relevant parameters as feature vectors, the required inputs and outputs i.e. the annotated *training* data is obtained. Table 3.2 shows a sample from the training data used for recovery rate prediction.

Table 3.2 : Snapshot from deep learning model training dataset.

Sample	Past Recovery Rates	Inflow	Outflow	Eigenvector Centrality	···	Weather Event	Recovery Rate Ground Truth
1	-0.8199 -0.3009 0.6547	0	2	1.836487	···	0	1.2358
2	1.1475 0.9521 1.1596	0	4	3.662118	···	0	1.6973
3	0.6509 0.6541 -0.2450	8	1	2.195037	···	0	2.4523
4	0.9340 0.4593 1.4523	4	9	3.241101	···	1	-1.9532
5	1.2451 1.2523 1.5436	2	11	6.546862	···	0	0.9568
6	.1.9723 1.2508 1.2469	5	4	2.132774	···	0	2.9682
7	0.6206 1.2460 1.1705	7	5	8.361506	···	0	1.1381
8	1.9734 2.3502 1.8601	11	9	1.137709	···	1	0.5723
9	0.9626 -0.6262 -1.1409	3	6	1.407725	···	1	0.8555
10	0.4569 1.2484 1.5961	16	6	1.850520	···	0	1.7811

Thanks to the ground truth data available at hand, the model is always adjusts itself to make closer estimations when fed with a set of specific features. Contrary to unsupervised learning, supervised learning has the advantage of tailoring the model to perform better on the real world data. Although, there are some pitfalls about this as the model is more prone to pick up biases from the dataset and is also heavily effected by the distribution of the dataset. In this study's particular dataset, recovery rates are scattered around scalar 1 with a normal distribution as seen in Figure 3.2. This would make the model favor the values around 1 more, since it is constantly gets updated with

samples with that value. There are relatively fewer samples that cover the scenarios where recovery rate is about -4 or +4. Still, the constantly informed training phase overweighs these disadvantages.

3.2.2 Temporal context

The training data at hand is in the form of tabular data. Which means each row represents a separate state and is a sample by its own. According to the hypothesis function in expression 3.2, only the samples from timestep t is required to get the recovery rate at next time step. Since the data samples are sequential i.e. they are ordered in time domain, temporal context can also be added. This means that using not only the previous timestep, but several past samples to predict the next one to provide context to the samples. The hypothesis function then becomes as in equation 3.8.

$$f(\sigma_{t-2}, \sigma_{t-1}, \sigma_t, \delta_{t-2}, \delta_{t-1}, \delta_t) = \delta_{t+1} \tag{3.8}$$

The performance of deep learning models for tabular data is somewhat on par with that of non-deep models such as Gradient Boosted Decision Trees [21], with no generally applicable superiority between the two [22]. Deep learning approach is chosen because of better flexibility with the type of data inputs. Note that sequential data points can also be represented as a single point representing 3 distinct timesteps but throughout the study abstract data manipulations are avoided when possible to keep the data complexity at minimum.

4. DECISION MAKING FOR RESILIENCY

Estimating the future states of the air traffic network is only one part of the act, the other valuable proposition lies in coming up with a robust decision making to make the network more resilient. Resiliency in this context means the ability to reduce the *overall* delay *change* i.e. boosting delay reduction or diminishing delay increase. Therefore the validation for the study's hypothesis does not require an absolute reduction in delay but rather a resilient behaviour induced by the suggested actions. These actions will be provided by a mathematical model called the agent which is obtained from reinforcement learning approach.

4.1 Reinforcement Learning

Reinforcement learning is a paradigm in machine learning that involves an agent learning some actions in an environment with a feedback of scalar reward signal. In this case the agent is the mathematical model or the policy that takes the environment parameters as the input and provides actions or action possibilities as the output. This means the decision making mechanism that is required for preventive actions lies within this mathematical model or agent. This policy can be deterministic or stochastic, as in equations 4.1 and 4.2. The former gives a mapping from a state to an action, while the latter gives a distribution of actions that sum up to 1 for all possible actions. Since the observable parameters in the environment (infection rate, recovery rate) are deterministic and discrete by nature, a deterministic policy will be a good match. This way, a state to action mapper will be obtained in a manner that is fully optimized for delay prevention.

$$\pi(s), \quad \pi_d : S \to A \tag{4.1}$$

$$\pi(a|s), \quad \pi_s : S \times A \to [0,1] \tag{4.2}$$

As it is the case with all learning frameworks, the model needs to learn, meaning it needs to be trained. The only way to herd this learning process is through the shaping of this scalar reward signal. The model creates a mapping from environment parameters to the actions and collects feedback depending on how good it is at deciding. This means to end up with a robust decision making algorithm, the environment parameters must be accurate and the reward signal must be guiding the model towards optimal behavior. The epidemic model in equation 2.1 is the differential relation between the $\dot{p}$ (delay change) values and the governing parameters (β, δ, p). Therefore, it is intuitive to formulate the reward signal in a way to minimize the value of this expression. The equation can be reorganized as in the following equation 4.3.

$$\dot{p}_i(t) = \sum_{j=1}^{N} \beta_{ij} p_j(t) - \delta_i p_i(t) - \sum_{j=1}^{N} \beta_{ij} p_j(t) p_i(t) \tag{4.3}$$

Since β and p values are bounded in $[0,1]$, the term $\sum_{j=1}^{N} \beta_{ij} p_j(t) p_i(t)$ is always positive, this means the expression is upper-bounded by the remaining terms, as shown in equation 4.4.

$$\sum_{j=1}^{N} \beta_{ij} p_j(t) - \delta_i p_i(t) \geq \sum_{j=1}^{N} \beta_{ij} p_j(t) - \delta_i p_i(t) - \sum_{j=1}^{N} \beta_{ij} p_j(t) p_i(t) \tag{4.4}$$

$$\text{where} \quad \sum_{j=1}^{N} \beta_{ij} p_j(t) p_i(t) \geq 0 \tag{4.5}$$

Now that the upper boundary is defined, the overall range of $\dot{p}$ can be moved towards lower values, by lowering the upper bound itself. The idea is minimizing the maximums of this range, so that the resiliency of the network can be increased with absolute certainty. Due to the complexity of the network and contributing parameters, this approach (boundary shifting) is the only one that guarantees minimization under any circumstance.

The upper bounding term $\sum_{j=1}^{N} \beta_{ij} p_j(t) - \delta_i p_i(t)$ can also be written in matrix notation rather than summation to represent the whole system, as shown in term 4.6

$$\sum_{j=1}^{N} \beta_{ij} p_j(t) - \delta_i p_i(t) = (B - \Delta)\hat{p} \tag{4.6}$$

where B is the infection matrix of the whole network as in example expression 2.3, Δ is diagonal matrix that is formed by placing the recovery rate (δ) values diagonally ($diag(\delta)$) and the $\hat{p}$ vector is the difference in average delay per flight (p) values for all airports. An example of this term would look like expression 4.7.

$$\overbrace{\begin{bmatrix} -1.50 & 0.14 & 0.14 & 0 & 0.28 & 0.43 \\ 0.50 & -1.72 & 0.25 & 0 & 0.25 & 0 \\ 0 & 0.50 & 0.54 & 0 & 0 & 0.50 \\ 0.14 & 0.14 & 0.43 & -0.49 & 0.28 & 0 \\ 0 & 0 & 0 & 0 & -0.79 & 1.00 \\ 0 & 0 & 0.20 & 0 & 0.80 & -1.13 \end{bmatrix}}^{B-\Delta} \overbrace{\begin{bmatrix} -0.13 \\ 0.30 \\ 0.23 \\ -0.41 \\ -0.77 \\ 0.46 \end{bmatrix}}^{\hat{p}} \tag{4.7}$$

The above expression represents the upper bound in the equation 4.4. The only free parameter in the expression is the infection matrix, B, which represents the traffic flow among the network. Therefore minimization can only be achieved through manipulating this matrix properly, which is also means coming up with best possible actions to reduce the overall delay over the network.

4.1.1 Reward signal

The expression in 4.6 can be minimized by lowering the real part of the largest eigenvalue of the matrix $(B - \Delta)$. Since the real part of the largest eigenvalue is a scalar value, its change can be directly linked to the reward signal with a gain to give positive rewards to its minimization. Such that

$$\text{reward} = -k * real(\lambda^*_{max_t} - \lambda_{max_t}) \tag{4.8}$$

One modification to this reward can be to make the difference of real parts quadratic. This would hint at the reinforcement model to not bother with small changes and emphasize actions that create large differences. Hence the expression becomes

$$\text{reward} = \left\{ \begin{array}{ll} k * real(\lambda^*_{max_t} - \lambda_{max_t})^2 & \text{if } real(\lambda^*_{max_t} - \lambda_{max_t}) \leq 0 \\ -k * real(\lambda^*_{max_t} - \lambda_{max_t})^2 & \text{if } real(\lambda^*_{max_t} - \lambda_{max_t}) > 0 \end{array} \right\} \tag{4.9}$$

The quadratic form quietens the differences around zero and amplifies the ones larger than one. Although the quadratic form has a weakened gradient around zero, it can compensate that with a strong gradient at large values. Which means the values of actions that cause large differences in system resiliency will be more clearly spaced, potentially making it easier for the model to settle at an optimized policy.

4.1.2 Action space

As mentioned before, actions can be taken through manipulating the B matrix only and this matrix depends on the flow matrix of the network (or adjacency matrix of the network graph). Therefore, the actions must act on this flow matrix which has integers that represent the relevant OD pair's flight count at that specific time window. Changing the flow matrix consequently changes the infection matrix, B. Let the flow matrix with positive integer entries be A, the action flow will then be as follows

$$\text{action} \quad \rightarrow \quad A \quad \rightarrow \quad B \quad \rightarrow \quad (B-\Delta)\hat{p}$$

The actions can be formulated as choosing from non-zero OD pairs and canceling a single flight. This would subtract 1 from the respective flow matrix entry. A sample action then would be as shown below

$$\text{action} = (i, j)$$

which just gives an index from the flow matrix whose entry is non-zero (meaning there is a flight on that OD pair) and decrements it by one i.e. canceling a single flight on OD pair at a particular time. Then the new infection matrix can be calculated from the manipulated flow matrix and the change in network stability can be measured directly from equation 4.9.

The scenario is less straightforward when we introduce *delays* into the action space. For instance, when a delay of 15 minutes is applied to an index at i, j, that particular traffic transfers to the next flow matrix, since flow matrices cover 15 minute windows. This means the new action would take the form

$$\text{action} = (i, j, \text{delay index})$$

where *delay index* represents encodings to express the delay amount as seen below.

$0 \rightarrow$ Cancel flight.

$1 \rightarrow$ Delay 15 mins.

$2 \rightarrow$ Delay 30 mins.

$3 \rightarrow$ Delay 45 mins.

$4 \rightarrow$ Delay 1 hour.

To give the reward signal to this action-state pair, equation 4.9 must be calculated before and after the action, to obtain the difference. Since the traffic is delayed, rather then being cancelled, it is shifted to the next flow matrix. The next flow matrix will change the next infection matrix, B_{t+1}. Recovery rates will also be different for the next time step (δ_{t+1}). This means the reward calculation at equation 4.9 must now account for both $B_t - \delta_t$ and $B_{t+1} - \delta_{t+1}$.

In fact, when considering that a flight usually takes more than an hour, a flight hold or cancellation affects more than a single time window. A 2-hour long flight would take up 8 time windows for 15 minutes each. This means for 8 consecutive flow matrices, the respective OD pair will have 1 count more. When the action cancels this flight, all 8 of the flow matrices lose a count from the respective index. This means the reward equation 4.9, which compares the before and after states of the network to assess an action's benefit, must compare the consecutive states as a whole. Let ε be equal to sum of consecutive flow matrices of 12.

$$\varepsilon = \sum_{i=1}^{12} \lambda_{max_t} \tag{4.10}$$

Since a flow matrix represents a 15 minute window, 12 consecutive matrices cover a duration of 3 hours. This duration should be more than enough, considering the average flight time of flights inside Europe was 101 minutes in 2022 [23]. Then, the modified ε^* becomes

$$\varepsilon^* = \sum_{i=1}^{12} \lambda^*_{max_t} \tag{4.11}$$

The reward function then takes the form in equation 4.12.

$$\text{reward} = \left\{ \begin{array}{ll} k * real(\varepsilon^* - \varepsilon)^2 & \text{if } real(\varepsilon^* - \varepsilon) \leq 0 \\ -k * real(\varepsilon^* - \varepsilon)^2 & \text{if } real(\varepsilon^* - \varepsilon) > 0 \end{array} \right\} \tag{4.12}$$

4.2 RL Training

Reinforcement learning models are trained by defining an agent, a set of possible actions for the agent to take, and a reward function that the agent seeks to maximize. The agent interacts with an environment by taking actions and observing the resulting state and reward. Over time, the agent learns the optimal policy, or sequence of actions, to take in each state in order to maximize the total reward it receives. In the scope of this study, the environment is actually the state of the network, which can be fully summarized by the matrix $B - \Delta$ in expression 4.7. The agent is actually the model itself with reward being the function in 4.12.

The training process typically involves a trial-and-error approach where the agent explores the environment by taking random actions at first, and then gradually adjusts its actions based on the feedback it receives from the reward function. This process is known as the exploration-exploitation trade-off, where the agent must balance between trying new actions to learn more about the environment and exploiting its current knowledge to maximize the reward. This is why most of the time, the beginning of the training is very unstable, with agent going for different kinds of actions in a variety of different scenarios, however as the training goes on, the agent stabilizes around a possible optimal action policy.

There are several algorithms used to train reinforcement learning models, including Q-learning, policy gradient methods, and actor-critic methods. These algorithms differ in the way they estimate the optimal policy and update the agent's actions based on the observed rewards. As explained in 4.1, a deterministic policy is a better fit for this specific use case. However, even stochastic policies can be made deterministic

after training if the action with the highest probability is always chosen over remaining actions. This is also the case with this study's approach to which helps the training by utilizing the exploratory side of the stochastic algorithms while keeping it deterministic at final simulations.

4.2.1 Advantage Actor - Critic algorithm

The implementation of the action generating model is based on an Advantage Actor Critic (A2C) algorithm. It is a model-free, online, policy-based reinforcement learning algorithm that simultaneously learns a policy and a value function. Actor-Critic and Advantage function are two parts combined that form the A2C algorithm.

The policy and value functions are learned simultaneously in Actor-Critic algorithms. The value function is represented by a critic network that calculates the expected reward for a given state, whereas the policy is represented by an actor network that links states to actions. The critic network is updated to reduce the discrepancy between the expected reward and the actual reward, while the actor network is modified to maximize the expected reward.

The advantage of taking a given action in relation to the anticipated reward of the existing policy is determined using the advantage function technique. The difference between the expected reward of the present policy and the expected reward of adopting a particular action in a particular state is used to calculate the advantage function.

The algorithm works as follows:

1. Initialize the actor and critic networks with random weights.
2. For each time step:
    - Observe the current state of the environment.
    - Use the actor network to select an action based on the current state.
    - Take the selected action and observe the reward and the next state.
    - Calculate the advantage function.
    - Update the actor network using the advantage function and the policy gradient.

- Update the critic network using the mean-squared error loss between the estimated value and the actual reward.
- Set the current state to the next state.

3. Repeat step 2 for a fixed number of episodes or until convergence.

With a mathematical notation the flow of the ideas can be expressed as the pseudo-code in algorithm 1.

Algorithm 1 Advantage Actor Critic Algorithm

Input: Environment with state space S and action space A, Actor network with parameters θ, Critic network with parameters w, Discount factor γ, Number of episodes N, Number of time steps per episode T, Learning rate α

Output: Trained actor and critic networks

Initialize the actor and critic network parameters θ and w with small random values.

for *each episode n from* 1 *to N* **do**

Initialize the starting state s_0.

for *each time step t from* 0 *to* $T-1$ **do**

Use the actor network to calculate the action probabilities $\pi(a|s_t,\theta)$ for each action a in state s_t.

Sample an action a_t from the action probabilities.

Take action a_t and observe the reward r_t and the next state s_{t+1}.

Calculate the advantage A_t using the critic network:

$$A_t = r_t + \gamma V(s_{t+1},w) - V(s_t,w)$$

Update the actor network using the policy gradient:

$$\nabla_\theta J(\theta) = \mathbb{E}s_t \sim \rho^\pi, a_t \sim \pi[\nabla\theta \log \pi(a_t|s_t,\theta)A_t]$$

Update the critic network using the mean squared error loss:

$$L(w) = \frac{1}{2}\sum_{t=0}^{T-1}(r_t + \gamma V(s_{t+1},w) - V(s_t,w))^2$$

Update the critic network parameters w using gradient descent:

$$w \leftarrow w - \alpha\nabla_w L(w)$$

Set s_t to s_{t+1}.

end

end

Return the trained actor and critic networks.

With this algorithm the an actor and a critic network is obtained and actor network generates the desired optimal actions when fed with the $B - \Delta$ matrix from equation 4.6.

5. AIRSPACE ANOMALY DETECTION

The topic of anomaly detection is intriguing especially for systems that are critical. Certain precautions are frequently needed for safety-critical systems in order to identify potential anomalous events that could damage or disrupt the system. This requirement results from the possibility that a systemic issue or "anomalous" event could result in damage of huge proportions. A person could be hurt or killed, money could be lost, or value could be damaged. It could also be a hoax or a danger to national security. A small anomaly can have an unlimited impact both immediately and over time. Therefore, it is essential to create tools for spotting or anticipating such abnormal events. Airspaces and air traffic networks are inherently linked, since airspaces are the mediums that the air traffic travels through. Hence, disruptions in airspaces can alter the air traffic in negative way. The remaining part of the study tries to figure out a way to estimate the disruptions and their effects. All these events have in common is that they hinder the flow of air traffic. However, there is another problem in airspaces. The problem is airspace and airways violations. Since airspaces and airways are a part of the air traffic regulations, events that stem from this issue play a large enough role that cannot be ignored. This chapter describes a methodology that can detect a flights type from flight parameters i.e. a model that can tell if a streaming flight data belongs to a commercial or military aircraft. This ability is important since it provides a data-driven method that can classify any aircraft into 4 main categories: commercial, civil, rotary-winged and military aircraft.

5.1 Unsupervised Learning

Unsupervised learning involve training algorithms on data without any direct supervision or direction. These type of algorithms work with unlabeled data, which means they are tasked with identifying patterns and relationships in the data without any prior knowledge of what the data represents, in contrast to supervised

learning, where the machine learning model is given labeled examples to learn from. Unsupervised learning's objective is to give the machine learning model the ability to identify underlying patterns or structures in the data that would be challenging or impossible to find through manual analysis. This is quite helpful in anomaly detection since anomalous events usually have no labels and are very rare by definition.

Due to the nature of the data and patterns, it might often be impossible for humans to recognize anomalous events. If a flying anomaly is minor and follows a clear pattern, a person might be able to spot it right away. However, more subtle anomalies necessitate the examination of other factors. Humans are capable of doing this as well, but the work and costs would be enormous. For a human to perform a lot of repetitive chores, it is not particularly advantageous or productive. An anomaly is defined in a fairly imprecise way. A helicopter's flight, for instance, would appear unusual if the monitoring system is set up for commercial planes. Therefore, discretization of some kind is essential. In the end, the definition and identification of non-anomalous occurrences must be exact and homogenous in order to find anomalies. Because of this, grouping the regular patterns would be a very beneficial strategy to use before looking for outliers.

5.2 Autoencoders

Autoencoders are made up of an encoder and a decoder as their main components. A lower-dimensional representation of the input data is created by the encoder, and a reconstruction of the original data using the lower-dimensional representation is created by the decoder. Autoencoders are designed to learn a compressed representation of the input data that captures its key features while omitting noise and unimportant details. The model is trained to reduce the reconstruction error between the input data and its reconstructed output during the training process.

5.3 Training Process and Data

Distribution graphs as seen in Figure 5.1 are significant because they give a preliminary projection of any biases the trained model may have. For instance, if an unsupervised learning model is trained on this data set's Altitude data, due to the higher frequency,

it will be biased towards the cruise altitudes. The model will be more influenced by the higher frequency of the cruise level altitudes simply because it will get updated for cruise data more frequently. Knowing this helps to forecast potential learning model behavior and set reasonable expectations. Because it is easy to distinguish and different levels of values have different frequencies, the normal distribution of the data is preferred. Despite the size of the data set, there are a fair number of grounded flights; removing them reduced the computational cost. The data that were used in this study also cover a period of about 120 days. It is unlikely that there isn't even one anomaly in today's world. Anomalies, as the name suggests, are extremely uncommon. Consequently, it won't have an impact on the model. However, it would be preferable if the data set contained no anomalies at all. However, attempting to remove anomalies from a data set can be challenging and not always easy. It is better to simply use as much data as you can in order to combat this issue because, for each anomaly added to the data set, multiples of regular events will also be added, limiting their influence on the model.

The training data has a time-series structure and is multivariate. Due to the reason LSTM layers use the data from earlier inputs, they can handle input data that is time-dependent. In this context, a window length of three samples is deemed sufficient. Be aware that the frequency of data collection is actually correlated with the window length choice. The training data used in this study were gathered at a frequency of 0.125 Hz, or one sample every eight seconds. As a result, a window length of 3 samples is actually 24 seconds, which is a sufficient amount of time for real-time detection. To illustrate, a single sample vector with six scalar features is denoted as $x_t \in \mathbb{R}^6$, and the two sample vectors that came before it are denoted as $x_t - 1 \in \mathbb{R}^6$ and $x_t - 2 \in \mathbb{R}^6$. They combine to create a 2D array that is fed into the model, which then tries to reconstruct the three input vectors.

The output of the model is then used to measure the mean squared error between the input and output sequences. The mean squared error is calculated as in equation 5.1, where x_i is the input vector and the $\hat{x}_i$ is the reconstructed vector.

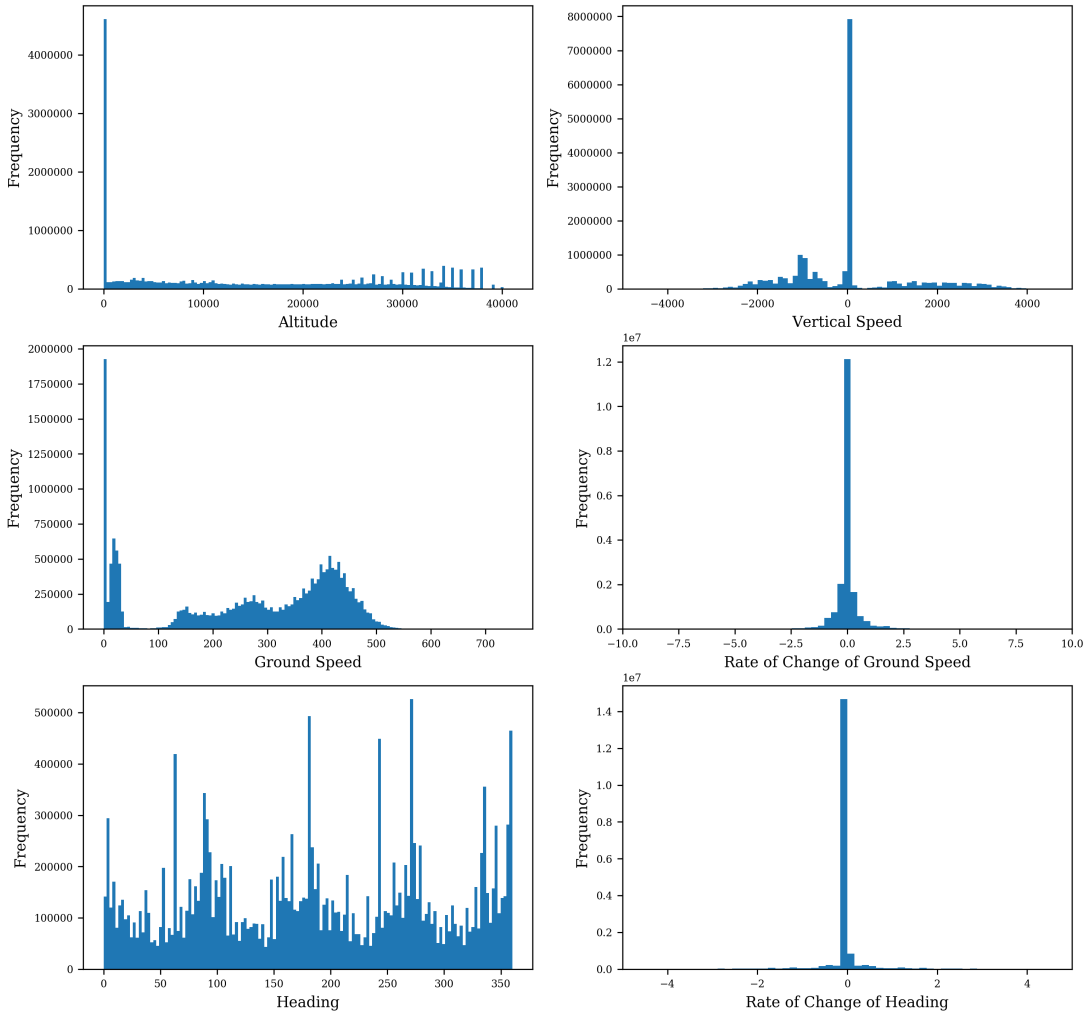

Figure 5.1 : Frequency distribution of features

$$\mathscr{E} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2 \tag{5.1}$$

The chain rule of derivative is then used to compute and propagate backwards the gradients of the weights that would reduce this error. The optimizer Adam, a popular optimizer that has been demonstrated to converge in any local minimum with a positive Hessian, is used to handle this process. Several experiments were conducted, and the end result is the model's final architecture. The model's performance in terms of model validation and generalization has been taken into account when tuning the network's size (the number of hidden units), layers, dropout layers, activation functions, initializers, etc. The experimental models have bottleneck dimensions between 1 and 5, hidden units ranging from 128 to 1024, layers ranging from 2 to

6, and only the encoder part (the decoder part is symmetric). The Adam optimizer, the *ReLU* activation function, a batch size of between 32 and 256, and other frequently used options have been used for the other parameters because they have been proven to produce good results. In general, growing the model's size and number of layers decreased the training loss, but doing so came at a cost in terms of computation and a faster rate of overfitting, where the model memorizes the data rather than learning from it.

There are 188 flight trajectories in the test set. 47 different days were used to create the trajectories, and each day was sampled from 4 separate categories. There is not a single grounded sample in the test set. In the test set, the trajectory (sample) length ranges from 20 to 4500. Each of the four autoencoders receives a sample from the test set, and four distinct reconstruction errors are generated. The sample is given the class of the lowest RE generating autoencoder, and a prediction is made. A design that is more effective at classifying flight patterns is likely to be better at spotting abnormalities. An RE threshold needs to be chosen in order to find anomalies. The RE threshold can vary from application to application and is somewhat dynamic. As a result, this study does not suggest a definite answer about the threshold; nonetheless, as a reasonable first effort, a threshold of two times the worth of training loss can be used. Thus, if the RE of a sample exceeds the defined threshold, an anomaly will be identified.

6. CONCLUSIONS

The aim of this thesis was to model the air traffic network on the airport level so that the propagation of delays and disruptive events can be measured and predicted and then be fed to a decision-making algorithm to output cost-conscious delay preventing actions. Our approach uses a mathematical epidemic spreading model to represent the propagations within the network, a deep learning model to estimate delay handling performance of each airport and a reinforcement learning model to produce the mentioned delay preventing actions. The epidemic model is used for airports in a similar fashion to infectious diseases. Through a set of parameters, the model accounts for the temporal and topological spreading of delays among airports. To effectively implement the model to an air traffic network, "infection rate" and "recovery rate" parameters got linked to statistical quantities about the network. Infection rates are obtained from traffic between the airports and recovery rates are obtained from implementing the model to historic data.

The recovery rate is used to accurately predict the propagations among the airports. It is directly linked to the resiliency of an airport and correlates with how well an airport handles the reactionary delays. So, to estimate the future propagation states of the network, these two parameters need to be predicted. Infection rates are linked to traffic therefore are already known from the flight plans, recovery rate, on the other hand, needs to be estimated for future states. With this information, delays and disruptive events (delay spikes) can be propagated across the whole network, resulting with the predictions of how the delay will change per flight. The DL model is trained on historic data so that it can estimate the delay handling of all airports, when fed with relevant information about the airport capacity, demand, weather events, existence of regulation etc. Through the recovery rate parameter, each airport in the network has a way to communicate its performance to the spreading model, it decreases when there are disruptive events on the airport, it increases when the airport is functioning properly.

The final flow of information can be summarized as in Figure 6.1.

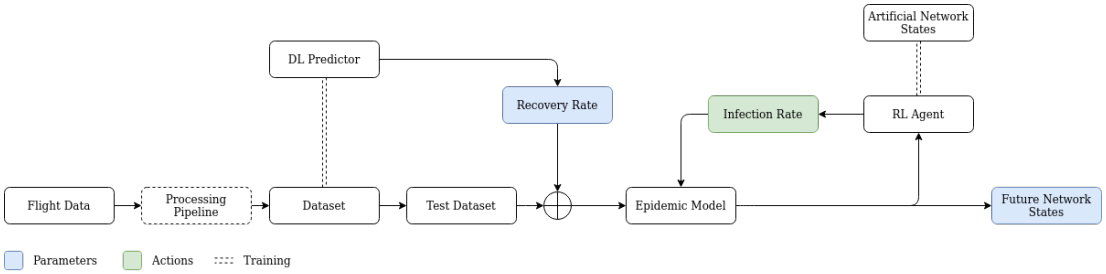

Figure 6.1 : Representative diagram for the overall algorithm flow.

Here, to make sure the methodology performs as intended a test dataset is formed. Then the recovery rates of this dataset is estimated through the deep learning model and then the matrix term in equation 4.6 is modified with the new recovery rates. This modified term is then fed into the trained actor network and an action is generated. The action is applied to modify this term again (with the change in infection rates) and equation 4.3 is recalculated to get the $\dot{p}$ values. This obtained value is then *compared* with the real $\dot{p}$ value (since we already know how the delays will change from historic data) to assess whether are approach gives a satisfactory performance or not.

In Figure 6.2 and Figure 6.3 it can be seen that epidemic model and DL prediction can accurately estimate the change of P value (directly related to delay per flight) for future states of the network. The accuracy of this estimation is of paramount importance for generating preventive measures. On these figures, blue line shows the actual change of delay per flight in 15 minute brackets, this means when P dot is positive, average delay per flight is increased. The orange line represents our prediction on P dot value. So it is desired that these two lines are close to each other as much as possible. Because if the prediction is accurate, correct preventive actions can be taken to mitigate the propagation of delays. Right axes show the correspondence of P dot value in in terms of minutes. This is obtained by simply multiplying P dot value with 15 minutes, since it is a value normalized by 15 minute windows.

The algorithm constantly tries to increase this network stability through the mathematical formula in a cost-effective manner and within a set of constraints. The algorithm is trained for millions of artificial scenarios of air traffic flow controlling, so that it can learn to prevent delay propagation and accumulation. Through an intelligent

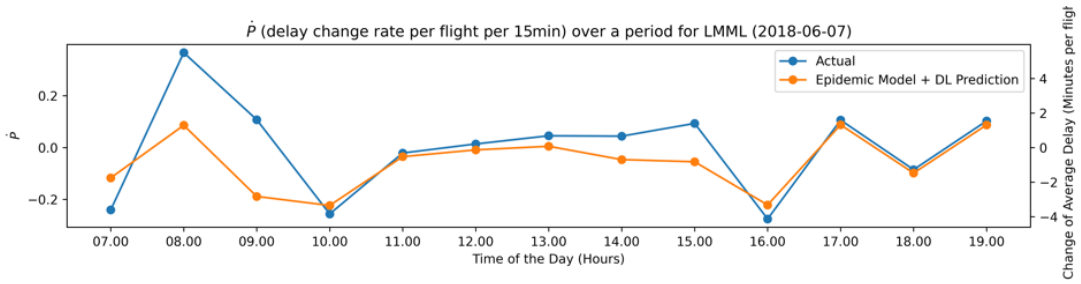

Figure 6.2 : Delay change rate of a randomly selected airport.

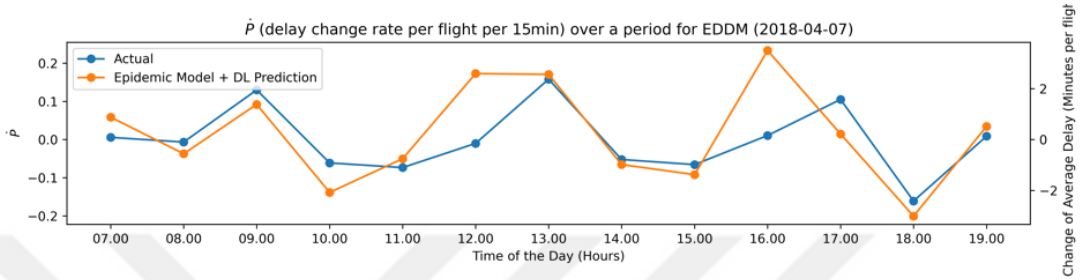

Figure 6.3 : Delay change rate of a randomly selected airport.

reward shaping, the algorithm knows when it does a good job on preventing the delays or when it fails. This way, it is constantly improved to be used at testing and simulations. The actions are grounding, 15-, 30-, 45- and 60-minute hold-offs.

In Figure 6.4 and Figure 6.5 a plot of the validation from one of the simulations can be seen. As stated in the figure, the change in P value (directly related to delay per flight) drops when the actions are taken, it should be noted that the actions do not guarantee the elimination of delays, rather, it improves the resiliency, helping the network stabilize delay accumulations and reach to optimal levels quicker. The magnitude of these changes is also related to the definition of cost, in these simulations a conservative (high costs for actions) approach is taken for computational efficiency. That is why only 25 actions are taken from the decision-making algorithm. The right axes show what the P dot value corresponds in minutes. The P dot value is normalized by number of flights and by 15 minutes to get a value bounded in 0 and 1. To convert it back to minutes, it can be multiplied by number of flights and 15 minutes again. The left side plots show almost all positive values of P dot, this means the average delay increases in this time period for these airports. However, after the actions are taken, the rates of these increases are reduced. Right hand side plots show how much reduction

is done on average in time windows. The improvements can range from half a second to 9 minutes, depending on the state of network at that time.

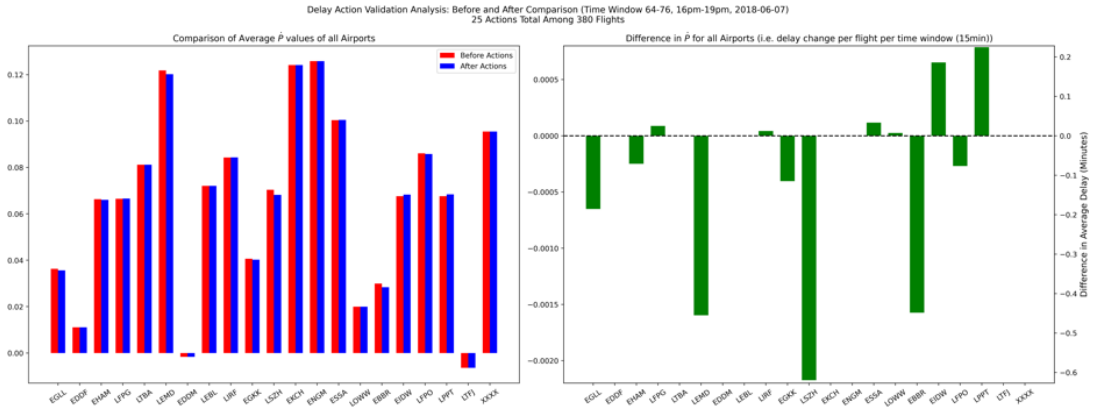

Figure 6.4 : Simulation results (plot on right magnifies the differences of plot on left for better observability).

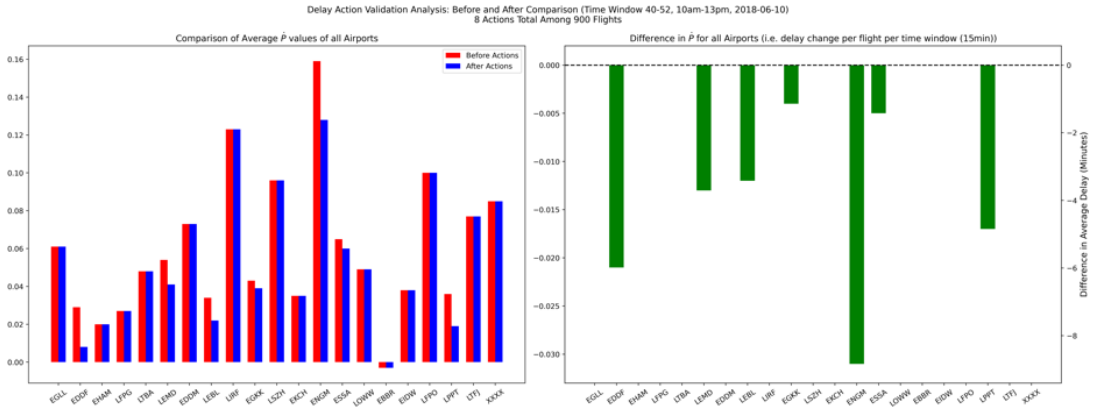

Figure 6.5 : Simulation results (plot on right magnifies the differences of plot on left for better observability).

There has been another simulation for the autoencoding model to test the model's ability to classify the flight trajectories into different types and to detect anomalies from time-based inputs. Aligned with this purpose, the model has been put to test with the data of different airspaces including different types of flights. Classification report from a simulation is shown in Table 6.1 and Table 6.2.

Ultimately, it is observed that regarding the problems of modelling network propagations, a combination of mathematical model and data-driven approach provides more flexibility and robustness compared to a fully switch over from mathematical

Table 6.1 : Confusion matrix for LSTM Autoencoder type classification.

	Prediction			
	Commercial	Civil	Rotary	Military
Commercial	35	8	1	3
Civil	18	21	6	2
Rotary	3	6	37	1
Military	10	4	1	32

Table 6.2 : Binary classification report for LSTM Autoencoder type classification.

	Precision	Recall	F1-Score
Civil & Commercial	0.78	0.87	0.82
Military & Rotary	0.86	0.76	0.80
Accuracy		0.81	

modelling to purely data-driven approach. This way it becomes possible to fully harness both the stability of mathematical models and the insights of data-driven models. Additionally, thanks to the nature of the problem, artificial scenarios can be easily manufactured to train an RL model, which then can be utilized as a generic optimal action generator. Overall, the methodology practiced in the study proves to be a good option to the problem of resiliency management on air traffic networks.

REFERENCES

[1] **Walker, C.** (2022). CODA DIGEST ANNUAL 2021, **Technical ReportCDA_2021_04**, EUROCONTROL, https://www.eurocontrol.int/sites/default/files/2022-04/eurocontrol-coda-digest-annual-report-2021.pdf.

[2] **Peterson, E.B.**, **Neels, K.**, **Barczi, N. and Graham, T.** (2013). The Economic Cost of Airline Flight Delay, *Journal of Transport Economics and Policy*, *47*(1), 107–121, http://www.jstor.org/stable/24396355.

[3] **Pyrgiotis, N.**, **Malone, K.M. and Odoni, A.** (2013). Modelling delay propagation within an airport network, *Transportation Research Part C: Emerging Technologies*, *27*, 60–75.

[4] **Rebollo, J.J. and Balakrishnan, H.** (2014). Characterization and prediction of air traffic delays, *Transportation Research Part C: Emerging Technologies*, *44*, 231–241, https://www.sciencedirect.com/science/article/pii/S0968090X14001041.

[5] **Li, Q. and Jing, R.** (2021). Characterization of delay propagation in the air traffic network, *Journal of Air Transport Management*, *94*, 102075, https://www.sciencedirect.com/science/article/pii/S0969699721000582.

[6] **Gopalakrishnan, K. and Balakrishnan, H.** (2017). A comparative analysis of models for predicting delays in air traffic networks, *Air Traffic Management Research and Development Seminar*, http://hdl.handle.net/1721.1/114750.

[7] **Takeichi, N.**, **Kaida, R.**, **Shimomura, A. and Yamauchi, T.** Prediction of Delay due to Air Traffic Control by Machine Learning, https://arc.aiaa.org/doi/abs/10.2514/6.2017-1323, https://arc.aiaa.org/doi/pdf/10.2514/6.2017-1323.

[8] **Gui, G.**, **Zhou, Z.**, **Wang, J.**, **Liu, F. and Sun, J.** (2020). Machine Learning Aided Air Traffic Flow Analysis Based on Aviation Big Data, *IEEE Transactions on Vehicular Technology*, *69*(5), 4817–4826.

[9] **Li, S.**, **Xie, D.**, **Zhang, X.**, **Zhang, Z. and Bai, W.** (2020). Data-Driven Modeling of Systemic Air Traffic Delay Propagation: An Epidemic Model Approach, *Journal of Advanced Transportation*, *2020*.

[10] **Kermack, W.O. and McKendrick, A.G.** (1927). A contribution to the mathematical theory of epidemics, *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, *115*(772), 700–721.

[11] **Nowzari, C.**, **Preciado, V.M. and Pappas, G.J.** (2016). Analysis and Control of Epidemics: A Survey of Spreading Processes on Complex Networks, *IEEE Control Systems Magazine*, *36*(1), 26–46.

[12] **Lajmanovich, A. and Yorke, J.A.** (1976). A deterministic model for gonorrhea in a nonhomogeneous population, *Mathematical Biosciences*, *28*(3), 221–236, https://www.sciencedirect.com/science/article/pii/0025556476901255.

[13] **Tukey, J.W.** *et al.* (1977). *Exploratory data analysis*, volume 2, Reading, MA.

[14] **Goodwin, L.D. and Leech, N.L.** (2006). Understanding Correlation: Factors That Affect the Size of r, *The Journal of Experimental Education*, *74*(3), 249–266, https://doi.org/10.3200/JEXE.74.3.249-266, https://doi.org/10.3200/JEXE.74.3.249-266.

[15] **LeCun, Y.**, **Bengio, Y. and Hinton, G.** (2015). Deep learning, *nature*, *521*(7553), 436.

[16] **Goodfellow, I.J.**, **Bengio, Y. and Courville, A.** (2016). *Deep Learning*, MIT Press, Cambridge, MA, USA, http://www.deeplearningbook.org.

[17] **Ash, R.B.** (1998). *A Primer of Abstract Mathematics*, Mathematical Association of America, USA.

[18] **Learned-Miller, E.G.** (2014). Introduction to supervised learning, *I: Department of Computer Science, University of Massachusetts*, 3.

[19] **Hornik, K.**, **Stinchcombe, M. and White, H.** (1989). Multilayer feedforward networks are universal approximators, *Neural networks*, *2*(5), 359–366.

[20] **Cunningham, P.**, **Cord, M. and Delany, S.J.**, (2008). Supervised learning, Machine learning techniques for multimedia, Springer, pp.21–49.

[21] **Friedman, J.H.** (2001). Greedy Function Approximation: A Gradient Boosting Machine, *The Annals of Statistics*, *29*(5), 1189–1232, http://www.jstor.org/stable/2699986.

[22] **Gorishniy, Y.**, **Rubachev, I.**, **Khrulkov, V. and Babenko, A.** (2021). Revisiting Deep Learning Models for Tabular Data, ***M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang and J.W. Vaughan***, *editors, Advances in Neural Information Processing Systems*, volume 34, Curran Associates, Inc., pp.18932–18943, https://proceedings.neurips.cc/paper/2021/file/9d86d83f925f2149e9edb0ac3b49229c-Paper.pdf.

[23] (2023). EUROCONTROL Data Snapshot, **Technical Report**, EUROCONTROL, https://www.eurocontrol.int/sites/default/files/2023-03/eurocontrol-data-snapshot-40.pdf, date retrieved 16-04-2023.

CURRICULUM VITAE

Name SURNAME: Muhammet Aksoy

EDUCATION:

- **B.Sc.:** 2020, Istanbul Technical University, Faculty of Aeronautics and Astronautics, Aeronautical Engineering
- **M.Sc.:** 2023, Istanbul Technical University, Faculty of Aeronautics and Astronautics, Aeronautical and Astronautical Engineering

PROFESSIONAL EXPERIENCE AND REWARDS:

- **2019-Present** Research Engineer in Istanbul Technical University Aerospace Research Center.
- **2021-Present** Research Assistant in Istanbul Technical University Aviation Institute.

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:

- Koyuncu E., **Aksoy M.**, Munoz A., Pons J., Delahaye D., Zopp R., Kuenz A., Soler M. (2022). ATM network modelling, uncertainty propagation with thunderstorm disruptions, *12th EASN International Conference on Innovation in Aviation and Space for opening New Horizons* October 19, 2020, Barcelona, Spain. (Presentation Instance)
- Kuenz A. , Knabe F., Alex Y., Koyuncu E., **Aksoy M.**, Munoz A., Polaina M., Guemes A., Pons J., Prats X., Lavandier J., Delahaye D., Zopp R., Frias A., Garcia J., Soler M. (2022). Simulation Exercises for robust Flight dispatching solution under thunderstorm disruptions, *12th EASN International Conference on Innovation in Aviation and Space for opening New Horizons* October 19, 2020, Barcelona, Spain. (Presentation Instance)
- Lavandier J., Delahaye D., Gonzalez-Arribas D., Garcia J., Soler M., Koyuncu E., **Aksoy M.**, Munoz A., Pons J., Prats X., Zopp R., Kuenz A. (2022). Network-wide robust and resilient metaheuristic trajectory optimization under thunderstorm disruptions, *12th EASN International Conference on Innovation in Aviation and Space for opening New Horizons* October 19, 2020, Barcelona, Spain. (Presentation Instance)

OTHER PUBLICATIONS, PRESENTATIONS AND PATENTS:

- **Aksoy M.**, Ozdemir O., Guner G., Baspinar B., Koyuncu, E. (2021). Flight Trajectory Pattern Generalization and Abnormal Flight Detection with Generative Adversarial Network, *AIAA Scitech 2021 Forum*, 11–15 & 19–21 January 2021 Virtual Event, https://doi.org/10.2514/6.2021-0775. (Conference Paper Instance)