Universidad Politécnica de Madrid
Escuela Técnica Superior de Ingenieros Informáticos

Complex Networks and Data Mining: Toward a new perspective for the understanding of Air Transportation.

Ph. D. Thesis

Seddik Belkoura
Engineer (École Nationale Supérieure des Mines de Nancy)

2017

Escuela Técnica Superior de Ingenieros Informáticos
Universidad Politécnica de Madrid

Complex Networks and Data Mining: Toward a new perspective for the understanding of Air Transportation.

Author:

Seddik Belkoura
Engineer
École Nationale Supérieure des Mines de Nancy

Directors:

Dr. Massimiliano Zanin
Innaxis Research Institute

Dr. Antonio LaTorre
DATSI - Universidad Politécnica de Madrid

2017

Abstract

Complex systems, *i.e.* systems composed of a large set of elements transporting and interchanging information in a non-linear way, are constantly found all around us. In the last decades, the approach toward their understanding has shifted progressively from a *transportation* to an *information processing* point of view. In other words, we are moving from a movement-based analysis (*i.e.* tracking the movement of items through time and space to reconstruct various metrics about their behaviour) to a higher-level approach, where individual movements are left aside to focus on the distribution, processing and flow of the information within the system. The *information processing* approach presents the main advantage of being data-based, that is, that no *a priori* knowledge about the interactions in the system is needed, hence the absence of costly simulations models. Such paradigm perfectly fits within the air transport system, where thematics as important as delay propagation (for its economical, environmental and safety related consequences) has been until now mainly analysed from a *transportation* micro-level perspective. Yet, the progressive rise in aviation of data analyses encourages a more data-centred path. We here present the first work that aims at fostering the combined use of the intuitive microscopic point of view with a higher-level *information processing* approach, yielding a more complete characterisation of the delay propagation process. Specifically, the here propose a three-fold approach. First, we highlight the degree of subjectivity associated with network-based representations of the air transport system, which conditions the intelligence extracted from any information processing study. Secondly, we manufactur a new data mining technique to extract non-linear causality relationships, therefore enabling the creation of a more complete delay propagation network representation. Finally, we complement our results by a micro-level analysis, therefore ending up with a 360° view of the delay propagation process. These analysis have been performed mainly on a European dataset, but expanded to other airspaces whenever data have been available.

Keywords: Complex systems, complex networks, data mining, delay propagation.

Resumen

Existen muchos sistemas en el mundo real que se consideran sistemas complejos, es decir, sistemas compuestos de numerosos y diversos elementos que transportan e intercambian información de una manera no lineal. El enfoque microscópico adoptado para mejorar el entendimiento de dichos sistemas está siendo reemplazado últimamente por un planteamiento más macroscópico, es decir, por el procesamiento de la información del sistema. En otras palabras, las métricas de comportamiento resultantes de un rastreo físico e individual de los elementos del sistema están siendo abandonadas progresivamente en beneficio del estudio de la distribución, procesamiento y flujo de la información. Este nuevo enfoque tiene la importante ventaja de basarse en datos reales, sin necesidad de conocimientos previos para la construcción de modelos y, por lo tanto, sin necesidad de costosas simulaciones. El estudio del transporte aéreo en general y de la propagación de retrasos en particular, se presta perfectamente al uso de tal enfoque. Este tema tiene una alta importancia en el sector por sus consecuencias económicas y ambientales y por su relación con la seguridad del sistema, pero hasta ahora ha sido analizada casi exclusivamente desde una perspectiva microscópica. El reciente crecimiento del acceso a datos relacionados con la aviación parece favorecer un planteamiento más macroscópico. Desde nuestro punto de vista, esta tesis doctoral aborda por primera vez el estudio de la propagación de retrasos combinando la tradicional visión individual con una perspectiva más panorámica del proceso, resultando en una caracterización más completa. En concreto, el trabajo consta de tres partes. En primer lugar, se analiza el grado de subjetividad resultante de las posibles representaciones del sistema aéreo basadas en redes y cómo éstas condicionan los resultados obtenidos respecto a la propagación de retrasos. Posteriormente, se presenta la herramienta de análisis de datos creada para la extracción de relaciones causales no lineales y, por tanto, más adecuadas al problema de estudio. Finalmente, se completan los resultados con un análisis microscópico tradicional para proporcionar una visión global de proceso de propagación. Los análisis de este trabajo se han efectuado sobre datos del trafico aéreo europeo y han sido extendidos a otras regiones de acuerdo con los datos disponibles.

Acknowledgements

Cette section est peut-être la plus difficile à écrire.

Une thèse de doctorat est souvent perçue comme un texte ardu, dénué d'âme, froid, qui va rester à tout jamais rangé au fond d'un placard. Cependant, derrière les formules et les résultats scientifiques, il y a un parcours humain. Un parcours qui a débuté il y a de cela 3 années, dans une nouvelle ville, avec une nouvelle langue et de nouvelles coutumes. Alors, pour ceux qui ne liront probablement pas ce document (et je les comprends!), j'aimerais leur dédier ces quelques lignes pour qu'ils sachent que sans eux, cette histoire aurait été moins belle.

En premier lieu, je pense à ma maman. Maman, ta force de caractère, ton abnégation, ta générosité sont une source d'inspiration pour toute personne; et encore plus pour tes deux fils qui t'ont vu pleurer, tomber, sourire, te relever, persévérer, rire, aimer inconditionnellement et toujours en vivant tes émotions intensément. Tu es un modèle qui ne cesse de nous surprendre par tes capacités à aller de l'avant et ta volonté à améliorer les choses pour ceux qui t'entourent. J'espère que cette thèse n'est que le début d'un chemin qui cherche une voie honnête et satisfaisante, comme tu nous l'a appris, peut-être inconsciemment. Sache qu'on t'aime et qu'on sera toujours là.

Je voudrais dédier cette thèse a mon frère. Les années passent et je t'ai vu passé par différentes phases, toutes aussi difficiles. J'aurais voulu être a tes côtés pour t'apporter le soutiens nécessaire dans ton aventure. Je te souhaite la tranquillité que tu recherches dans la vie. Tu es l'une des personnes les plus importante dans ma vie, et je te dédies cette thèse car je sais que malgré les vannes qui suivront, tu es probablement celui qui se sentira le plus fier de moi.

Papa, je sais que tu seras fier de moi. Comme toujours.

Enfin, un clin d'oeil à la bolosse team (et raph), pour ne jamais répondre sérieusement à une question, et rester fidèles à vous-mêmes. Vous me manquez.

Por otro lado, me gustaría mencionar a otras personas que hicieron que esta aventura sea aún mejor. En primero, todas las maravillosas personas de Innaxis. Con vosotros, he aprendido un idioma, he descubierto un país y me he reído todos, pero todos, los días. La vida es más divertida con vosotros como compañeros. El buen ambiente fomenta el profesionalismo, en esto sois un espejo en que mirarse, y Innaxis tiene mucho futuro con un equipo así. Así que

gracias, Carlos, David, Paula, Héctor, Cristina, Arantxa, Samuel, Jorge y Inés, por darme esta oportunidad, creer en mí y enseñarme cosas todos los días (aunque algunos días sean cosas del Hola).

Tengo palabras especiales por Massimiliano, quien ha sido un mentor en todo el sentido de la palabra. Me has enseñado tantas cosas, y sigues sorprendiéndome con tus capacidades cada vez. Carlos, el día de mi entrevista, me dijo que estaría bajo la tutela de un futuro Nobel; pues no exageraba. Te deseo lo mejor por el futuro, y levanto mi cerveza a más ideas y publicaciones a venir.

Gracias a mis tutores Antonio y Chema, quienes, a pesar de la distancia, han mostrado gran implicación en mi trabajo. Os lo agradezco. Me gustaría agradecerle a Ángel Rodríguez su infinita bondad y su ayuda a la hora de entender las procedimientos académicos de la universidad.

JonnyBoy, te dedico estas lineas para conmemorar los varios intentos a entender el mundo, que a menudo, a base de cervezas, han podido resultar en un camino transcendental maravilloso o en preguntas fundamentales potentes.

Finalmente, dedico este trabajo a una persona especial: Nerea (aquí sentada a mi lado leyendo un libro mientras escribo estas lineas) quien, sin darse cuenta, nos hizo, a mi y a este trabajo, mejores. Te quiero.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Many, if not all, complex systems that surround us are the result of some kind of transport process. One may for instance think of social networks, in which individuals interact by transporting and interchanging information [J+08, SFS+09, ASZ12]; power systems, clearly transporting energy in its different forms [ABCX09]; or the brain, in which neurons interact by means of chemical and electrical signals [KSJ+00]. All those systems can be studied following two approaches. The first involves considering them as pure transportation systems: the researcher directly analyses the movements taking place within it, *i.e.* tracking the movement of items through time and space to synthesise various metrics about their behaviour. On the other hand, a much more abstract and complementary approach can be envisioned: individual movements are discarded, to look instead at how information is processed at different places. In other words, the focus is on how information is distributed among, combined at, and modified by the different elements composing the system; thus shifting from a *transportation* to an *information processing* approach.

The former approach would appear to be *prima facie* more direct and intuitive, as the researcher only has to focus on the movements that can be detected in the system; on the contrary, the latter requires resorting to abstract concepts, like information and computation, which may

not be simple to visualise. The information processing approach has nevertheless historically yielded relevant results, representing the technique of choice when studying complex systems. There are several reasons behind this success. First, there is no need for *a priori* knowledge of the interactions between the elements of the system and of its transport processes, as relevant information is inferred from the system's dynamics. Second, missing data, as a result of structural, rather than technical limitations, can be dealt with by means of similar strategies. Finally, no large-scale simulations are required: while the computational cost associated with some analyses may be relevant, they are usually several orders of magnitude smaller than those needed to construct a full transport simulator or to follow all aircraft trajectories. In fact, it is by and large recognised that, *as systems get more complex, the paradigm shift from transportation to information processing is the only viable solution for understanding their dynamics.* Any attempt to characterise the dynamics of distributed complex systems must necessarily involve information theory, as the latter is the language of computation, and this defines their nature [GM95]. For instance, the study of electric charge transfer in semiconductors focuses on the logical operations executed on the signals, and not on the individual transfer of electrical charges [FRB83, TLW$^+$89]; in molecular diffusion, cellular automata, and pattern formation, in which macro-scale behaviours are described by means of information terms, like the existence of constraints, conservation laws, or the competing interactions between elementary units; and in general in any natural spatio-temporal pattern formation process [CH93].

1.1.1 Delay propagation and information processing in air transport

The air transport system has mostly been analysed from a transport perspective, consistently with the nature of its objectives. Such an approach has especially been applied in the analysis and characterisation of delay propagation, one of the most important research topics in air transport management. Delays have profound implications in the cost-efficiency [CT11] and safety of the system [Duy93], and contribute to the negative impact of air transport on the environment [CDLHJ07]. To illustrate, delays imposed over 1.3 billion euro w.r.t. European airlines in 2007 [CT11], and 1 minute of ground delay implies between 1 kg to 4 kg of fuel

consumption, one order of magnitude higher in the case of airborne delay [CDLHJ07]. Delays have thus been studied by constructing large-scale simulations, as for instance in [ASRA04, Jan05, Wu05, Jet09, FRE13].

Is such approach effective in providing insights about the mechanisms behind delay propagation? Simulations and models can only yield a limited view, at best, of the dynamics of the system, as most interactions are unknown. Elements as essential as crew assignment and airline policies, both of them defining how delays can propagate across different flights, are usually relatively unknown and difficult to model. Any artificial model is thus bound to some level of incompleteness, whose repercussions are hard to assess. Additionally, when one tries to overcome such gaps through expert knowledge (*e.g.* through stakeholder consultations), the results are biased by people's preconceptions: *what the system is supposed to do*, as opposed to *what it is really doing*. As a result, the mechanisms behind delay propagation are largely still poorly modelled, and policies designed to reduce delays have limited effectiveness.

A better understanding of air transport architectural interactions may come from the complementary study of how the system processes information. When aircraft travel between two airports, they do not only *transport* passengers and goods, but also transmit information about the status of the departure airport (and of the whole crossed airspace) to the destination. One airport receiving (possibly delayed) flights and dispatching them to other airports is not just managing the movement of the aircraft, but is also *receiving*, *processing* and *retransmitting* information about the system. To the best of our knowledge, this macroscopic approach has never been attempted in air transport, except for a few studies dealing with statistical and macroeconomic factors [MSF10, ZN10, MT13]. We trust that combining the intuitive microscopic point of view with a higher level approach yield a more exhaustive, astute and rigorous characterisation of the delay propagation process.

1.1.2 Objectives

This PhD Thesis aims at developing a comprehensive attempt at describing and modelling air transport as an information processing system, by importing and adapting concepts that are

commonly used in complex network theory and data mining. In other words:

> *By considering air transport as an information processing system, to lay down the foundations for a new methodological approach based on complex networks and data mining, for the understanding of its dynamics, with a special focus on the problem of delays propagation.*

This global objective is organised in three specific sub-objectives, which are described in details here below.

Obj.1: To describe Air Transport system as a network. Complex network theory aims at creating a graph representation of the real system, therefore enabling a quantitative and qualitative analysis. Yet, the representation inevitably dissociates from reality whenever complexity is high, whether the disjunction be purposive to make the analysis tractable; or embedded within the available data. This first objective aims at answering the following question: does the heterogeneity of the system's representation bias the ensuing analysis ?

This will involve assessing the stability of complex networks framework over Air Transport common representations. The analysis will not only focus on the variability of the static characteristics of the representations, but, in light of the main objective of this PhD Thesis, will also tackle its impact on the delay propagation dynamics.

Obj.2: Delay propagation from an information processing perspective.

What does it mean to consider delay propagation as an information processing system? Air transport delays will be represented by means of functional networks, a representation that is now standard in neuroscience and other fields, including physics [MS99]. Nodes, typically airports, but also Area Control Centres or other elements of interest, are pairwise connected whenever a delay propagation is detected between them. The result is then an abstract representation of the structure created by the propagation process, on which all the power of the complex network framework can be applied: for instance, one can detect central nodes, *i.e.*

nodes mostly responsible for the delay propagation, and in which resources should be preferentially deployed; communities, or groups of nodes that strongly share delays; and more generally one can compare the structures created by different situations, *e.g.* summer *vs.* winter, normal days *vs.* days with extremely adverse weather, *etc.*

One important issue will be the definition of appropriate metrics for the detection of the propagation of the information, *i.e.* delays between airports. Existing metrics are not suited for a complete characterisation of the delay propagation phenomenon; new, tailor-made ones have to be developed; firstly, by trying to adapt existing data mining techniques to the idiosyncrasies of air transport data as the non-linearity of some relationships.

Obj.3: A transportation point of view. While the macroscopic approach adopted in Obj. 1 and 2 has been proven more relevant in recent years, a microscopic perspective still yields insightful information that is not visible from a higher level. Furthermore, the recent and continuous increase in computer power and parallelisation strategies makes micro-level analysis tractable. The third objective therefore aims at completing previously extracted high-level intelligence by increasing the precision of the analysis. In other words, elementary components of the Air Transport system (namely, airports and aircraft) will be inspected, through data mining techniques, in order to draw a 360° vision of delay propagation process.

1.2 Main contributions and publications

Several publications have emerged from the elaboration of this PhD Thesis, whose main contributions are resumed here below.

We present an analysis of sampling strategies for the reduction of complex network dimensionality. All complex systems, when studied through the lenses of complex network and data mining theories, entail data-related problems. The (possibly *unknown*) bias within the data, the presence of superfluous information or the high computational cost are all examples of how data can somehow disrupt the assessment of both static and dynamic topologies of the network.

For the first time we here present a quantification of the perturbation introduced by a sampling strategy, assessed across several airspaces. The analysis of the sampling strategy (Chapter 4) has been published in Transportation Research, Part E (Impact Factor of 2.974) [BCPZ16] and further extended in the Chinese Journal of Aeronautics (Impact Factor of 0.977) [CBZ16].

Furthermore, we provide a complete methodology for the use of data mining techniques in the study of complex networks, covering all delay propagation characteristics: starting from delay data, we show how data mining techniques allow for the extraction of specific information flow within the network, targeting different but compatible propagation situations. Therefore, the study of the expected delay propagation network in Europe and China have been conducted and compared to (in the European case) to the disrupted propagation triggered by severe perturbations. The case of China can be found in the Chinese Journal of Aeronautics (Impact Factor of 0.977) [ZBY16] while the European case has been published and presented in the 7th USA & Europe International Conference on Research in Air Transportation [BZ16].

Finally, we completed these macroscopic concepts by applying data mining in a more micro-level, focusing particularly on two singular elements of the system responsible for delay propagation: airports and aircraft. This has led to the creation of a new metric for the non-linear characterisation of airports behaviour (delay-wise), published in the Journal of Advanced Transportation (Impact Factor of 1.292) [BPZ17]; and to a methodology for the extraction of the spatial and temporal distribution of delay-generating events from flight trajectories, published in Transportation Research, Part C (Impact Factor of 3.805) [BPZ16].

Beyond these six contributions, a new paper have been prepared and is now under consideration in the Decision Support System journal (Impact Factor of 3.222). Specifically, the paper proposes a methodology to improve the interpretability of black-box classification algorithms. Furthermore, another paper explaining how non-linear causalities has been developed in order to highlight complex propagation channels within the network, and resulting in the creation of a Python Library, will be submitted to consideration in the BioInformatics Journal (Impact Factor of 7.307).

1.3 Structure of the document

Beyond this introduction, the second Chapter of this PhD Thesis discusses the current state of the art. It is intended at introducing the background material of the two main science fields used in this work, *i.e.* complex network theory and data mining, along with relevant studies of the Air Transport system. Both frameworks used in this work are data-based, thus Chapter 3 presents the data used to perform our analysis. Chapter 4 presents the non-negligible effect of standard sampling selection techniques to different types of network representations, and their effectiveness in improving the representativeness of the system; Chapter 5 tackles the problem of delay propagation through the investigation of different functional networks representing distinct mode of propagations, *i.e.* average, abnormal and non-linear channels; finally, Chapter 6 extracts knowledge from a microscopic level, complementing the macroscopic information extracted in previous chapters. Each one of these Chapters introduces a problem, then presents a solution tested on one or various air transport datasets and discuss its benefits and shortcomings. All results and insights have been summarised in Chapter 7, followed by the future lines of research, built on extensions or shortcomings of our work. For the sake of completeness, Annex A exposes a detailed description of our non-linear causality metric. Finally, as a topic transversal to all the proposed work here, Annex B presents a novel instrument for deep learning analysis which have been developed to enhance the interpretability of complex data mining tools, as their lack of transparency limits greatly the use of their advanced forecasting powers in Air Transport studies.

Chapter 2

State of the Art

There are two theories, or frameworks, which are by and large used in the study of complex systems. The first is the field of complex networks, which has emerged from graph theory to enhance the characterisation and understanding of the structural foundation of highly interconnected systems. The second, data mining, is the logical extension of traditional pattern analysis in data that - thriving on the recent and still on-going computational advances - consists in moving from large data sets in their unusable and unrefined form to the production of intelligence and insights. Both theories share similar goals and are seldom used simultaneously. The work proposed in this PhD Thesis unites them both, in order to elevate our understanding of air transportation as a complex system. In this chapter, we review the Literature of these two fields of research.

2.1 Data Mining

Analysis of patterns in data is not new. The concepts of averaging and grouping can be dated back to the 6th century BC in Ancient China [Goo68] or Greece [GP96], when statistics were gathered to help heads of state governing their countries in fiscal and military matters or even in music. From that moment on, the science of extracting information from data has never ceased to grow, basing itself on statistics, algebra and other low-level (*i.e. pure*) sciences, to

only find its ceiling in the limited computational capacities of the human brain. At present, with the proliferation, ubiquity and increasing power of computers, our abilities in data collection, storage, and manipulation are only limited by the different technological implementations of today's hardware (basically, speed and memory capacity). These computational advances renewed the ideal of an automatic data analysis, classification and understanding, which has subsequently been attracting growing attention since late 80's. From this ideal has emerged a new science or concept, called Knowledge Discovery in Databases (KDD)[1]. Throughout this Thesis, several concepts and techniques drawn from KDD will be used, such as data mining. It should be noted that data mining is a particular step of KDD; nevertheless, for the sake of completeness, the former will be described in the light of the latter.

2.1.1 The Science of Data Mining

While originally data mining was distinguished from KDD as being its algorithmic part, presently both terms are used interchangeably to refer to the overall process of discovering useful knowledge from data. Modern data mining, like every other science, needs to be supported by a scientific reasoning and a scientific method.

The scientific reasoning behind data mining is known as 'inductive approach' (see Fig. 2.1), which emerged as a response to the ineffectiveness of the mathematically proven 'deductive approach' in real world situations. In the inductivist theory, observations play two roles: first, in the discovery of scientific theories or models, and second, in their justification. A model (or a theory) is supposed to be designed foremost through the extrapolation of observations. Then, if large number of additional and independent observations conform to the model, and none to few deviates from it, the theory is supposed to be justified - or at least made more reliable.

As for the scientific method in data mining, it has been created by a consortium of NCR, SPSS, and Daimler-Benz companies in 1997. The process is called Cross-Industry Standard Process for Data Mining (CRISP-DM, see Fig.2.2) and consists in six standard phases [SBG+11, NMEI09]:

[1] In the last decade, the new concept of *data science* has emerged, whose goals include extracting knowledge from data and creating data products. KDD and data mining are usually considered as tools inside data science - see, for instance, [SBG+11] or [NMEI09]

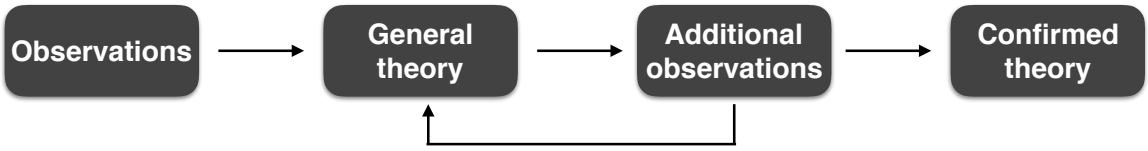

Figure 2.1: Inductive reasoning scheme.

Problem Understanding Solutions to a problem are inevitably framed partly in terms of things we do not observe directly, be they real (*e.g.* company units, complex interactions) or intangible (*e.g.* forces, laws of nature). As such, the building of a data mining model must be supported by a complete and sound understanding of the background problem, and thus by a clear and precise definition of the study objectives.

Data Understanding Stated generally, this phase consists in acquiring the data, describing them and performing an assessment of their quality. The acquisition and integration of data, that might come from different sources, is far from trivial, as information might be present in different format or might be expressed in different units. The definition of a data map guiding the preparation of each element in the corresponding format is followed by a thorough description of the variables to assess their 'cleanliness' (*e.g.* percentage of blank entries, etc.).

Data Preparation This phase is arguably the most important of the whole process, as the success of the final analysis strongly depends on it, and may consume up to the 90% of time and resources. The basic idea of data preparation is a succession of operations to transform and condition data to create a data set. For instance, data transformation is the operation that assess how to express data variables; data filtering the one that chooses what to do with outliers; and data reduction aims at reducing the amount of data to be analysed. [CMS99, ZZY03] present reviews of the existing techniques.

Modelling This is when the algorithms are actually trained, and information extracted. Mathematically, each technique has its own logic and its own specific requirements about the format of input and output data. This is the reason behind the importance of the previous step, *i.e.* shaping data to fit into the algorithm specifications.

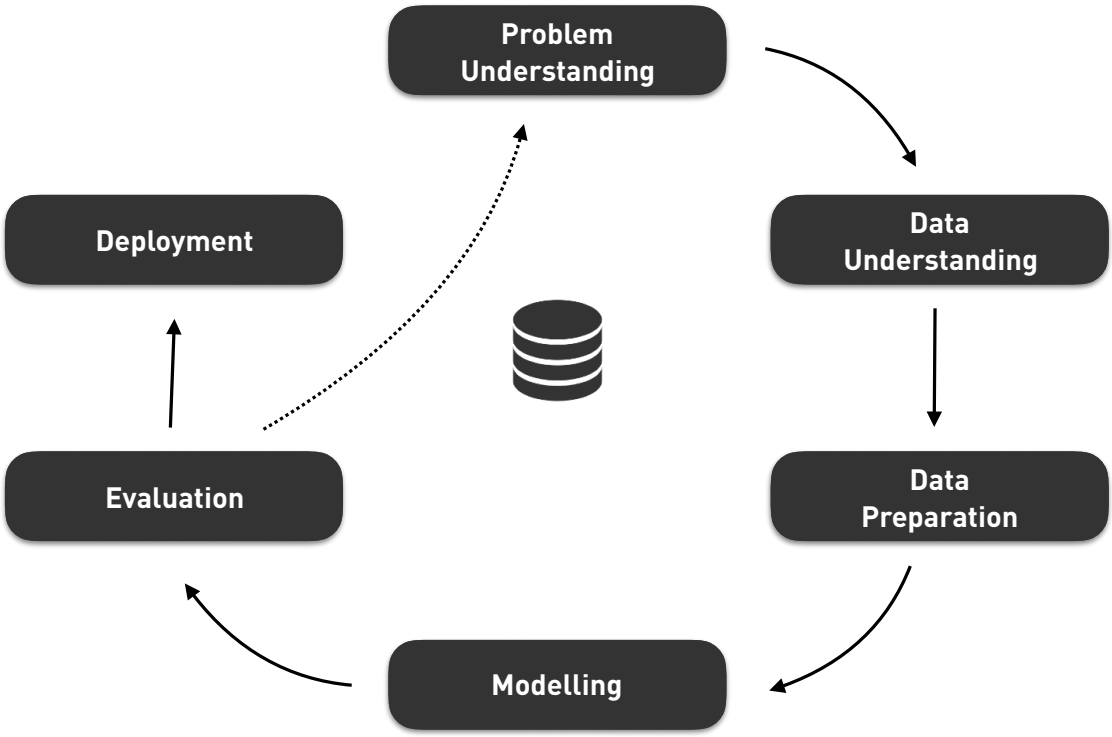

Figure 2.2: Knowledge Discovery in DataBase: Scientific Method.

Evaluation To specify which/if a solution describes the data, it is necessary to review the yielded output patterns in light of the background problem specified in the first phase. It is only when all relevant questions have been answered and the gained understanding of the problem have been judged satisfactory that one can move to the next phase. In all other cases, we reinitialise iteratively the process from step 1, including the new output patterns into the background problem.

Deployment When all information about the business problems has been gathered, this and the new knowledge have then to be organised and presented in an intelligible way, in order to be understood by professionals who may be unaware of the specific mathematics behind KDD.

2.1.2 Data Mining tasks and models

Data mining tasks, or the modelling phase of the KDD process, can be classified into two categories that correspond to two different levels of abstraction. Intuitively, one may expect

a data mining task to be predictive or descriptive; in other words, the patterns inferred from data can be used to make predictions or to cluster the data. On top of that, a task may be explanatory or not, inasmuch as it is aimed at increasing our understanding of the system. Note that the predictive and explanatory power of a model are independent aspects - even though they may be confused in some business applications. Let us delve deeper into those two categories and two levels of abstraction.

On one hand, the difference between predictive and descriptive algorithms is clear with respect to their objectives, as the former aims at forecasting the value of particular attributes based on the other variables (features), while the latter focuses on the description and discovery of hidden patterns encoded in the data, without implying any forecast. On the other hand, both build upon two shared principles. Firstly, the reliance on the calibration of a model over a portion of the information encoded in the data (called training set). Secondly, the implicit assumption that the data used for training the models are representative of the whole universe (*i.e* representative of all data generated by a system), thus ensuring the robustness of the model against any type of future instances.

Data mining models can thus be classified in two groups depending on whether they are predictively or descriptively aimed - the difference being partly bound to the type of model itself. An additional distinction must here be added, independently to the fact that an algorithm might be predictive or descriptive. The increasing complexity of some techniques challenges the ability of the human mind to extract or to access the rationale behind them - a concept synthesised in the term 'black-box algorithms'. This kind of algorithm are often compared with an imperfect oracle that can predict the outcome of an experiment or classify the entities of an environment, but provides no explanations whatsoever about its rationale. While black-box algorithms return high-level outputs, the user only has access to the low level mechanisms, thus making them very complex to be understood at a human level. This is an important limitation in some situations as financial regulations, fraud detection or automated systems, which require the models to be transparent, *i.e.* providing further explanation on the underlying dynamics. Note that this is independent from the predictive capabilities of the algorithms: even a perfect oracle, would it ever exist, would not help. The solution can only come from so called 'ex-

planatory' algorithms. Generally speaking, the only 'white-box' (*i.e.* transparent) models are shallow decision trees and sparse linear models (more details on this in Section 2.1.3). However, the use of such simple models comes with a reduction in the prediction accuracy, which is fostering the use of black-box models in many real-world applications, thus prioritising precision over explanation.

A tremendous variety of techniques and algorithms have been described in the literature, the most important of which are reviewed in the next subsection, focusing specifically on classification algorithms due to their relevance in problems involving complex networks.

2.1.3 Review of classification and prediction algorithms

As previously described, data mining algorithms are developed to learn from a set of training data, with the final aim of predicting a variable's next value or to classify a new unlabelled record. It is wieldy to transform any continuous prediction into a discrete classification - for that matter, any continuous entity could be discretised - while the opposite process is complex. Therefrom, all data mining algorithm that have been proposed in the last decades are either specifically dedicated to classification tasks or can perform both prediction and classification tasks - through a discretisation of the continuous prediction output. Each algorithm has different pros and cons and specific requirements on the format of the data. We list below the most successful and well-known types:

Regression models Regression analysis is a statistical process for estimating the relationships among variables. Specifically, it consists in tuning a function - that can be linear, non-linear or even multidimensional - to fit the data. Generally speaking, regression models belongs to the explanatory family, as they explicitly specify the influence of each variable on the output. Linear regressions, in particular, have extensively been used in practical applications [Fre91] because of their high interpretability. That said, regression models are hindered by a large number of explanatory variables - a problem known under the name of 'curse of dimensionality', see Section 2.1.5. To solve this problem, MLR

(Multiple Linear Regression) and later GAM (Generalised Additive Models, [HT90]) have been used. Basically, they consist in performing a fitting (linear for MLRs and non-linear for GAMs) for each explanatory variables - therefore decreasing the variance generated by the number of predictors. This category of algorithms is mainly used for prediction purposes, specifically for business intelligence where interpretability and control are important features.

LASSO The Least Absolute Shrinkage and Selection Operator is a regression method that penalises the absolute size of the regression coefficients, or in other words, that constrains their sum. The larger the penalty applied, the more regression parameters estimates are shrunk towards zero values, which is particularly efficient against highly correlated predictors [Tib96].

Artificial Neural Network (ANN) Neural Networks are inspired by the early understanding of the structure and function of the human brain. They are composed of artificial neurons, mimicking neural cells. performing a simple operation on aggregated data inputs and processing the result through a logistic function. This latter function plays the role of the activation process triggering the neurons' response - neurone's cells accumulates neighbouring cells' electrical impulses until reaching a threshold and 'firing' an impulse to an adjacent cell, see Fig. 2.3.

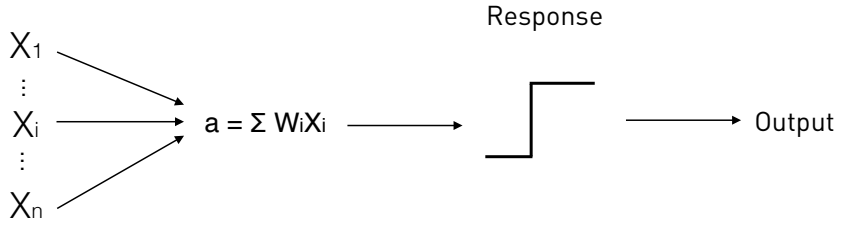

Figure 2.3: A Neuron's mechanism. The symbol X_i could stand for the input variables (in the case of an artificial neuron) as for the connected neurons (in the case of the human brain). W_i represent the numerical weights associated with each connection, which is analog to the strength of the synapses between brain neurons.

As in the brain, artificial neurones are connected together into an architecture or structure, called 'network', in which each input variable is connected to one or more output nodes. At the end of the network, one single output node is generally used when the network aims at predicting a continuous variable or performing a binary classification

(True/False or 1/0); on the other hand, various output nodes are present when the network is configured to do classification or estimation tasks. The most interesting property of a neural network arises when a middle layer of nodes (neurons) is intercalated between the input layer and output node(s), as shown in Fig. 2.4. The nodes in the additional layer provides the ability to model non-linear relationships between input nodes and output node(s). The greater the number of nodes in the middle layer, the greater the networks capacity to recognise non-linear behaviour. However, specific rules have yet to be established to define the optimal number of nodes to use.

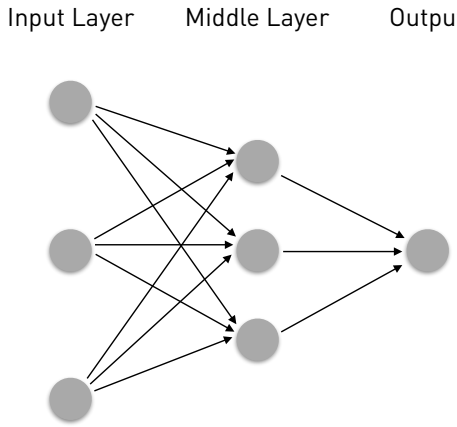

Figure 2.4: Artificial Neural Network basic structure.

The learning process of the neural network structure is here again loosely inspired by the human brain, as the connections' weights are adjusted iteratively through several existing adaptive processes. The most common is the 'back-propagation algorithm', which corrects the weights of misclassified points - or outlying points in case of a prediction process - as a function of the magnitude of the output error. The learning process is complemented by an optimisation process - the most common being the gradient descent - allowing a quicker convergence of the output. A loss function is defined, measuring the 'closeness' to the real output, and its gradient is measured at each step with respect to the weights in the network. This gradient is fed into the optimisation method, adjusting the weights of the links in order to minimise the loss function - or, in other words, in order to maximise the accuracy of the prediction or classification. All of the basics to build neural networks can be found in Refs. [HDB$^+$96, Zur92]. It must be highlighted that this algorithm is totally opaque, therefore classifying neural network as a non-explanatory algorithm.

Support Vector Machine (SVM) This method belongs to the category of the hyperplane classifiers, as it is based on drawing lines to distinguish categories between each other. The basic idea behind SVM is that a line separator is easier to compute than a complex non-linear one. Specifically, as shown in Fig. 2.5, original objects are rearranged (or *mapped*) using a set of functions called kernels - that can be linear, polynomial, radial or sigmoid - into a space where the two classes would be linearly separable. The best separation is achieved by the hyperplane that has the largest distance to the nearest training-data point of any class, as this minimises the error. More information on SVM can be found in [CV95].

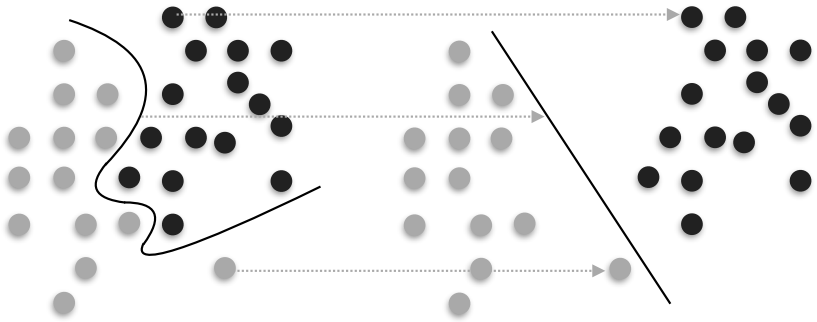

Figure 2.5: Support Vector Machine illustration.

Decision Trees They are hierarchical group of relationships organised in the form of a tree structure. The complete training data set is fed into the first node, called *root node*, for then being broken down into smaller and smaller subsets representing separate classes of the dataset - or continuous values when the algorithm is used as a regression tree. Each node in a decision tree represents an attribute, while each branch represents a value that the attribute can take - see Fig. 2.6. At each node containing non-uniform data, in the sense that they do not belong to the same category, a new attribute test is used to split them into smaller subsets. In other words, the aim is to divide the data subset into two (or more) smaller uniform pieces. This procedure is applied recursively on the resulting subsets, until all of them are composed of records of the same category.

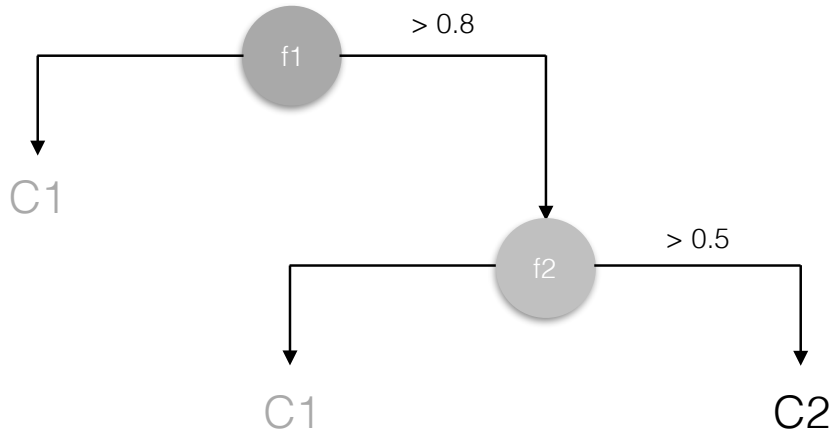

Figure 2.6: Decision tree illustration.

Decision trees present the main advantage of being simple to understand, provided the tree depth is limited, and are therefore considered as explanatory techniques. Besides, their ability to handle both numerical and categorical variables (hence little data preparation) and large datasets has made decision trees very popular. It must be noticed, though, that they suffer from the disadvantage of being based on heuristics in which locally-optimal decisions are made at each node.

Random Forests Random Forests were firstly proposed by Breiman [Bre01] and are based on tree predictors. Specifically, when the number of available variables of the dataset is large, the algorithm trains a number of trees on different subsets of the data, *i.e.* it grows an ensemble of trees, and the most popular class out of all the trees is then attributed to the record. Each tree in a random forest is grown as follows: *a*) a random subsample of the training dataset is selected as the tree's original node and *b*) a subsample of the attributes is selected, then *c*) the best split is computed for the specific tree with no pruning. The random forests algorithm has high accuracy among algorithms of classification and can handle high number of variables as they do not suffer from overfitting. However they are quite complex to understand - due to the high number of rationales - and are therefore considered as 'black-box' algorithms.

2.1.4 Validation

As we specified earlier, data mining - like inductivism - is observation based, in that models are both distilled from and validated by observations. If, in the previous section, we summarised some of the most important data mining algorithms for extracting models from observations, it is now necessary to validate them. This can be done through simple metrics, like the proportion of correct positive classifications or the distance of the forecast from the truth. However, such metrics are seldom sufficient to assess the real performance, as they are calculated on the training set: the question of how the algorithm generalises to the whole universe is still open. A model is defined as good not only when it fits the training set, but also when it generalises correctly to records never seen before. Note that the generalisation error can be quite large in comparison with the training error when the training set does not represent the characteristics of the whole universe - thus indicating that the model is overfitting the training data and losing its generalisation. The likelihood of this scenario increases with the complexity of the model, such that larger data sets are needed to complete the training process. In the case of two models with the same generalised error, the simplest is chosen according to the Occam's razor principle [RG01].

In order to evaluate the generalisation error of a model, it should suffice to consider a new data set, *i.e.* not used in the training phase, and measure the corresponding model error. This data set is commonly referred to as 'test set'. However, it is not that common to have so many data available to split it into a training and testing set, with the low availability of data forcing to resort to cross-validation techniques [ET95, K$^+$95, GHW79]. The main idea behind cross-validation is to cover all the possibilities offered by the data. K-fold cross-validation, for example, consists in partitioning the original data into k equally sized subsamples. Each part is then iteratively used as test set, while the others are used to train a model. This process is repeated k times with independent models, and the generalisation error of the model is estimated as the average of the errors obtained with the different testing sets. In the special case where k is equal to the number of observation, that is that the dataset is partitioned into subsamples of one singular observation, the method is called Leave-One-Out Cross Validation

(LOOCV).

2.1.5 Pitfalls

In order to ensure a correct and unbiased analysis of any real-world data set, one must be aware of several pitfalls that may compromise the outputs. For the sake of completeness, here we list the most important ones.

Sample size While having access to multiple features is commonly considered a positive aspect, one must be careful with the ratio between the amount of variables and the amount of instances (or observations). With more variables than observations, any pattern can emerge from random combinations, while it remains of little relevance for the problem. This effect is known under the name of 'curse of dimensionality', as firstly introduced by Richard E. Bellman [Bel57]. Some restrictions on the data may help avoiding this situation, as limiting the hypotheses being tested [GE03, JZ97], using cross-validation [K$^+$95, RPL10], or shuffling the data [Rus97].

Surgery Success	**Patient height**	**Patient number of siblings**	**Surgeon eye color**
No	1m78	3	Blue
No	1m50	0	Blue
Yes	1m80	0	Green
Yes	1m79	1	Green

Table 2.1: Importance of using a correct sample size.

Tab. 2.1 illustrates how easy it is to fall into this trap. Imagine an analysis aimed at defining the causes behind kidney surgery success. Due to the small sample size, irrelevant features may appear to be predictive - in this case, the surgeon eye color.

Grouping Due to the small amount of data describing certain types of events, one could be tempted to aggregate data from similar instances. Behind this practice hides the Simpson's paradox [Sim51] (or Yule-Simpson effect [Yul03]), and it is responsible for the appearance of trends in the aggregated data set that are different from the trends present in the non-aggregate data. Fig. 2.7 illustrate this concept, where three groups of data (respectively green squares, red circles and grey triangles) both display a common trend, *i.e.* a negative correlation between features 1 and 2 (black, grey and red solid lines). Nevertheless, when all groups are jointly analysed, a spurious positive correlation appears (dashed black line).

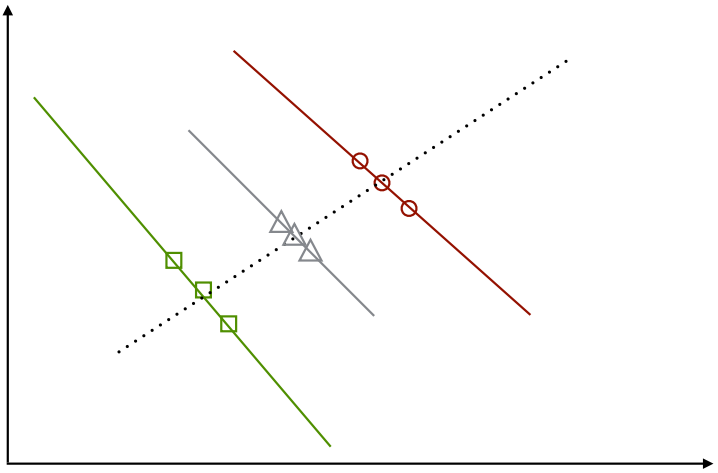

Figure 2.7: Example of the Simpson's paradox.

Stationarity The stationarity of the time series (*i.e.* of the data under study) is one of the most important requirement, as a large number of techniques implicitly require the data under study to be stationary for the conclusions to hold. In this context, an observable is defined as stationary if its statistical properties (*e.g.* variance, autocorrelation, etc.) are constant over time, which means that it is impossible to link the statistical properties of a sub-sample to a time scale. Two limiting situations may arise: that the underlying system it not stationary; or that, while being stationary in the present, this cannot be guaranteed in the future. They both can be tackled by either transforming the time series into stationary ones, or use techniques robust against non-stationarity. Why is this temporal independency fundamental? The reason lies in the foundations of data mining: the observations used to create a model must be representative of the whole universe. Thus, if one extracts a pattern from a system and the latter changes afterwards, the

validity of the former cannot be maintained.

Sample bias The need for stationarity and representativeness has important repercussions in the way data are sampled. If data are collected in a biased way, the knowledge extracted from them will manifest the same bias. In other words, the analyst must assure that both the training and test data come from the same distribution [AMMIL12].

Data snooping One of the most common trap is to compromise the analysis outcome by tuning the data set before the learning process. This usually happens when the data left for testing the model are, potentially in a subtle or hidden manner, affecting the learning process. A clear example is the normalisation of the data before slicing them into training and testing sets: the test data include information from the training set (through the normalisation) and, therefore, the predictive power of the model will artificially increase [AMMIL12, Whi00].

Causality Causality is an ephemeral and abstract concept, yet essential to understand the dynamics of a system. The common pitfall is to assume that a strong correlation implies causation, while many examples exist of spurious correlation that have nothing to do with causality. One way to avoid this error is to use the Granger Causality metric or Transfer Entropy [ZZRP12] when possible. See Section 2.2 for more details on causality.

2.2 Causality

The inductivist view to science is based upon the sheer error that a generalised prediction is tantamount to an explanation. The pattern of scientific reasoning behind inductivism has been heavily criticised, as the repetition of an observation cannot justify any subsequent theory. Such criticism is remarkably illustrated by Bertrand Russell [Rus01], with a chicken observing his farmer feeding him each and every day and thus extrapolating that the farmer will continue this generous behaviour the following day. The farmer ultimately wringing the chicken's neck proves that induction cannot justify any conclusion unless it has been placed into the adequate framework. Given some insight about the farmer's behaviour, the chicken might have been able

to understand why it was being fed, and predict his future death. This resumes one of the major complication of data mining: its relation with causal inference. What the chicken had in mind was a 'false' explanation of the farmer's behaviour, therefore expecting to be fed everyday. Had it guessed a different explanation - that the farmer's behaviour was driven by a more selfish motive - it would have extrapolated his behaviour differently. Through this illustration, Russell is trying to highlight the importance of the explanation in the inductive framework - knowing the causes driving a specific dynamic allow a better 'extrapolation' of it.

Due to the above considerations, one may expect explanations in data science to be supported by a causal model. That way, induction is supported by deductive thinking and the data mining model can be explained through causal logic. Nevertheless, this condition is seldom fulfilled.

One of the most well known instruments to find causal relationships is the Granger Causality test [Gra80] - an extremely powerful tool for assessing information exchange between different elements of a system, and understanding whether the dynamics of one of them is led by the other(s).

2.2.1 Granger Causality

The basic idea behind Granger Causality can be traced back to Wiener [Wie56] who conceived the notion that, if the prediction of one time series is improved by incorporating the knowledge of a second time series, then the latter is said to have a causal influence on the first.

The GC test is based on two very simple ideas, which take the form of two axioms:

1. Causes must precede their effects in time.
2. Information relating to a cause's past must improve the prediction of the effect above and beyond information contained in the collective past of all other measured variables (including the effect).

Granger [Gra88a, Gra88b] later formalised Wiener's idea in the context of linear regression models. Specifically, two auto-regressive models are fitted to the first time series – with and

without including the second time series – and the improvement of the prediction is measured by the ratio of the variance of the error terms. A ratio larger than one implies an improvement, hence a causal connection. At worst, this ratio is 1 and signifies causal independence between the two time series.

Therefore, a time serie X is considered to 'Granger-cause' another time serie Y if the inclusion of past values of X can improve the process of forecasting the values of Y. In mathematical terms, suppose X and Y to be two stochastic processes- Let us denote by $\sigma^2(Y_t \mid U_t^-)$ the variance of the residual of predicting time series Y using the accumulated information of the entire universe U from infinite past to present (the latter denoted by U_t^-; additionally, be $\sigma^2(Y_t \mid U_t^- \setminus X^-)$ the corresponding forecast error excluding X from the universe. Assuming the stationarity of the time series, the following definition of Granger Causality can be given:

Definition 2.1 *(Granger causality). If* $\sigma^2(Y_t \mid U_t^-) < \sigma^2(Y_t \mid U_t^- \setminus X^-)$*, then X granger-causes Y.*

Since its inception, Granger's and other derived causality metrics, as co-integration or transfer entropy to name a few [Sch00, SL08, Ver05], have been applied in economics [Hoo01], biomedicine/neuroscience [BDL+04, KDTB01, RFG05] and air transport [CC09, MSF10, BY13].

If Granger Causality has extensively been used in the analysis of real-world data, it has also been recognised that it presents several drawbacks [BS11]. From the point of view of a data analysis, two have to be highlighted. First, this metric is linear, in the sense that it assesses the presence of linear couplings between the time series - while there is a lot of real world cases where information propagates in a non-linear fashion [FVB+99, AMS04]. Second, time series must be stationary, requiring detrending and pre-processing the data [HMAS03], which is not always possible. In Section 5.2 we will show how to overcome these limitations and widen the possibilities for Granger's causality metric.

2.2.2 Extreme Event Causality

Granger causality and its related metrics share a common characteristic: the temporal dimension required to define the relationship. Yet the dynamic of some physical processes might not easily be observable, *e.g.* gene-gene interactions, as a single measurement is performed by subject and by gene, thus precluding the use of the temporal dimension. One can go further and state that some causalities might even be present with no need for a temporal dynamics. For instance, the offspring genetic material is caused by that of the parents, without any explicit dependence on time. Also, the correct location of a piece of a puzzle is defined by the global image, and not by the temporal sequence of the pieces' arrangement. It appears that some causalities must be discovered looking at the statistics of their realisations. Let us explain better this point.

This has been recently proposed in [Zan16] where the author resort to statistics of occurrences to define causality. Suppose a group of snapshots of a system, for which two properties X and Y are observed, each one naturally emerging as a consequence of the internal dynamics. If Y is also partly caused by X, then Y should be present whenever X is observed, as the former is a consequence of the latter. Nevertheless, due to the internal dynamics, Y can also appear spontaneously, *i.e.* in the absence of X. Note that there is nothing in the previous definition that logically requires causes to precede their effects, and that it is solely based on the statistics of appearance of both properties X and Y.

Mathematically, suppose p_1 the probability of the X-determined snapshots to have also the property Y; and p_2 that of the Y-determined snapshots to have the property X. In the case of a real causality, then $p_1 \approx 1$ and $p_2 \ll 1$. In the case of a confounding effect, *i.e.* when both X and Y are driven by another property Z, then $p_1 \approx p_2$. Thus, the test synthesises in testing the statistical significance of the hypothesis $p_1 > p_2$ through a binomial two-proportion z-test:

$$z = \frac{p_1 - p_2}{\sqrt{\hat{p}(1-\hat{p})(\frac{1}{n_1} + \frac{1}{n_2})}} \tag{2.1}$$

being n_1 and n_2 the number of snapshots associated with p_1 and p_2, and $\hat{p} = (n_1 p_1 + n_2 p_2)/(n_1 + n_2)$. The corresponding p-value is then obtained through a Gaussian cumulative distribution function, and the test is rejected or not depending on the chosen significance level.

Causality will play an important role in this PhD Thesis, as not only it allows a better understanding of the dynamics behind any system, but it is also the ultimate bridge between data science and complex networks, as it would allow to resume in a graph all the useful causal flows of information in a system. More on that in Chapter 5, but first, let us define more on details complex networks.

2.3 Complex Networks

During a long time, the nature of science was thought essentially reductionist. That is to say, science assuredly explains things reductively - by analysing them into smaller components. Every component is then eventually reduced to interacting molecules, and then to forces between atoms, etc. This vision of science is profoundly mistaken as it not only assumes that explanations always consists in analysing a system into smaller, simpler systems - how would this deal with fractals? - and, assuming that causes precedes events, eventually reduces every explication to the big bang.

Some systems falls outside the scope of reductionism and, thankfully, science evolved with new ways of dealing with reality. New theories, opposed to reductionism, allow predictions and explanations at every level of hierarchy. No one actually expects to extract laws of psychology from forces between atoms (mainly because it would be computationally intractable). The fact that under specific circumstances, the complex behaviour of a vast number of entities (be they particles or aircrafts) resolves itself into a measure of simplicity and comprehensibility is called emergence: high-level simplicity emerges from low-level complexity. The high-level behaviours of some systems cannot be deduced from the low-level properties of their constituting elements, as important information is codified in the interactions between these elements; such behaviours are known as emergent phenomena.

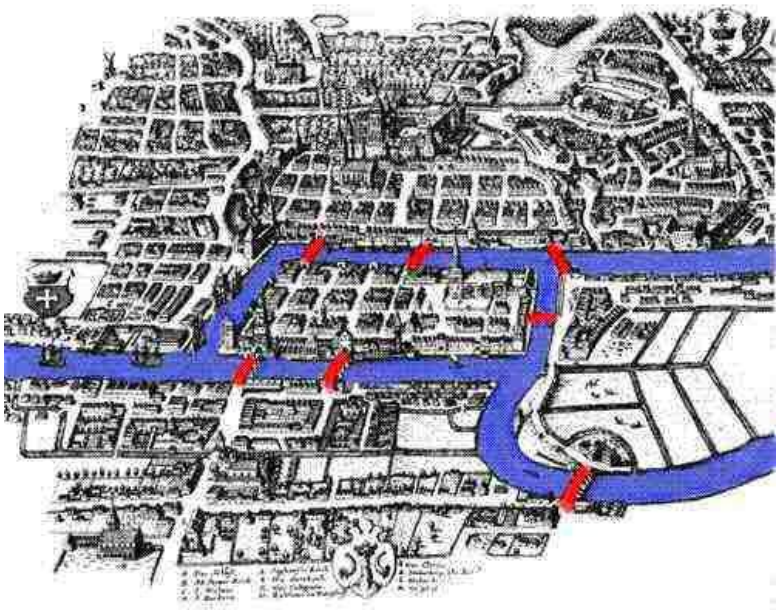

Figure 2.8: Map of de city of Königsberg in Euler's time showing the actual layout of the seven bridges (red) across the river Pregel (blue).

These systems are called Complex Systems: systems composed of a large number of elements interacting between them in a non-linear fashion, and giving birth to emergent behaviours. In fact, the theory of complex network has been extensively applied - from biology to economics and sociology - making interaction between elements of a system paramount to the element themselves.

2.3.1 Complex networks theory

Any description of the complex networks theory should start from its origin, namely from the graph theory developed by the Swiss physicist and mathematician Leonhard Euler [BLW76] to solve in 1735 the Königsberg bridges problem. Now called Kaliningrad (Russia), the city of Königsberg laid on both side of a river, with two island in the middle and seven bridges connecting all regions, see Fig. 2.8. The problem was to find whether a path through all city regions existed, crossing each bridge only once.

Euler's idea was spurred by the wish to exclude any unessential information. In other words, the form of the land regions or their internal structure bring no fundamental information to the

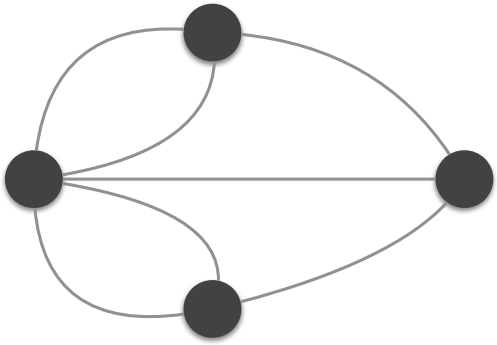

Figure 2.9: Königsberg problem: graph representation.

resolution of the problem. The useful information of the map was then represented virtually through a graph: land were nodes (or vertices) and bridges were edges (or links), see Fig. 2.9.

The concept behind Euler's idea is extremely powerful. Not only the nodes and links of the graph describe the interactions between the element of a system, but they also contain no information whatsoever about the nature of the elements themselves. This remarkable property of graphs enable them to abstractly represent the structure of any system, independently of the nature of the system itself. Now, the nature of the interaction between the elements of the system can be physical (*e.g.* a bridge) or abstract (*e.g.* information), but are usually of some intrinsic value to the network (may this value be positive or negative).

The observed complexity in the structure or behaviour of a system can thus be characterised through a simpler and more elegant network [2]. A change in the way a system is represented, while useful, is not enough to extract knowledge from it. Complex networks theory then evolved as the branch of statistical physics devoted to the analysis and characterisation of such abstract objects. More specifically, we need to develop a large set of measures describing the structure of the system in order to increase our understanding. In the next section, a review of the most common network metrics is provided - including the ones used in this PhD Thesis. For more complete information, please refer to [BLM$^+$06, CRTVB07, COJT$^+$11].

Topological metrics are usually classified in three families according to the relevant scale: micro- (single nodes or links), meso- (groups of few nodes) and macro-scale (the network as a whole).

[2]While graph and network might be as synonymous, the former refers to a simple object - having a regular or random structure - while a network is characterised by a more complex connection connectivity.

The micro-scale focuses on the properties of a single node, that is, observe how this singleton is connected with others; characterises the nature of this connections and even surmise the internal dynamic of the node (whatever entity might it represent). When we want to have a more general image of the network, but still partial, we ought to look at an intermediate level: not a single element nor the entire system, but a subset of the nodes. The notion of meso-scale is elusive and might be by and large defined as any regular structure that affects the connectivity of groups of nodes in the network [ACL+11] such as communities [For10], motifs [MSOI+02] or core-peripheries [Hol05]. Roughly speaking, communities are sets of nodes that are more connected among themselves than with the rest of the network; motifs are patterns that repeats themselves more than expected; and core-periphery structures results from the interaction between a highly connected core with sparse peripheral nodes. Note that all these notion are complex, in that emerging from the interaction of different elements, and not easy to be assessed in a quantitative way. The limitations associated with these meso-scale metrics are further discussed in [ZSM14]. The last type of scale, *i.e.* the macro-scale, considers the system as a whole to account for the global structure of the network. In other words, macro-scale metrics focus on the information flow within the network. For instance, the number of jumps to cross a network from one side to the other might be of importance in social networks; or the understanding of the role played by an element in the information propagation process.

2.3.2 Characterising networks

Any network is characterised by its number of nodes (usually denoted by n) and links (denoted by l). More precisely, it is fully defined by an adjacency matrix (denoted A), of size n x n, whose element $a_{i,j}$ has a value of 1 when a connection between the two nodes i and j exists, and 0 otherwise[3]. The number of links l can simply be derived from A as $l = \sum_{i,j} a_{i,j}$.

As previously introduced, all methods aiming at characterising the structure of a network that one can propose roughly fall into the three aforementioned categories (*i.e.* within the micro-,

[3]Note that this matrix may not be symmetric, *i.e.* $a_{i,j} \neq a_{j,i}$, in that it may exist a connection from node i to node j, but the connection $j \rightarrow i$ may be missing.

meso- and macro- scales). We will now proceed with a review of the most common metrics for each category.

Micro-scale metrics

The simplest example of a micro-scale metric is the degree, defined as the number of connections a node has - that is, its number of connected neighbours. The degree k of a node i is calculated as:

$$k_i = \sum_j a_{i,j}. \tag{2.2}$$

It must be noted that while this measure unveils some information about the importance of a node inside the structure of the network - as it might for instance be of interest to localise the more connected node - additional information is contained within the degree distribution $P(k)$. To illustrate, a long tailed distribution reflects a hub-and-spoke configuration - that is, comprising a few nodes that are highly connected while the others are isolated - therefore excerpting intelligence about the system's resilience to attacks or to random failures. This type of information conferred in a functional form must nevertheless be synthesised into a singular feature, in order to enable its use in data mining algorithms. The entropy of the degree distribution presents itself as a compelling solution [WTGX06]:

$$H = -\sum_k p(k)\log_2 p(k) \tag{2.3}$$

The minimum value $H = 0$ indicates a constant degree across all nodes, while a higher value indicates a more uniform distribution of degrees.

A second - and equally important - metric is obtained through the analysis of individual degrees: the assortativity (or degree correlation) of a system. It defines the presence of degree-correlations between connected nodes. High assortativity indicates high degree nodes' tendency to connect with other high degree nodes. A dissortative (negative assortativity) network indicates a tendency of high degree nodes to link with low degree ones. This tendency is assessed

by comparing the probability of this preferential connections with what would be expected in a random configuration of the network. It has been extensively applied to real-world systems such as social network (assortative), biological networks (dissortative) or technological network (dissortative) [New01b, GI95, DYB03, CCGJ02]. Mathematically, it is defined as $P(k^{'} \mid k)$, representing the probability of a node of degree $k^{'}$ to be linked with a node of degree k. It can be explicitly expressed through the Pearson coefficient correlation as:

$$D_c = \frac{1}{l} \sum_{j>i} \frac{1}{2}(k_i + k_j)a_{ij}. \tag{2.4}$$

For the sake of completeness, we also consider the maximum degree, which is the degree of the most connected node: $k_{\max} = \max_i k_i$.

An additional micro-scale metric of interest is the link density, defined as the proportion of actual links in the network, as compared to the total possible number of links n^2. Mathematically, we obtain:

$$l_d = \frac{\sum_{i,j} a_{i,j}}{n^2} = \frac{l}{n^2} \tag{2.5}$$

This definition demonstrates that the metric is defined within the $[0, 1]$ interval. A network with a null link density would be void, whilst a link density of 1 indicates a fully connected network.

Meso-scale metrics

When the focus is put on a group of nodes, though not the entire set, meso-scale metrics emerge. For the scope of this work, the two most important are the clustering coefficient and the Information Content (IC). Two additional metrics can be mentioned: motifs and communities. However, the concept and assessment of community - defined as groups of more densely interconnected nodes - is not consistently defined and will not be used in this PhD Thesis. More information about the different definitions can nevertheless be encountered in

[RCC+04, NG04, ZMW05]. Motifs are better defined and refers to the concept of clustering [MSOI+02]. Specifically, motifs are subgraphs of three or four nodes that appear more often than what statistically expected. However, such notion again will not be used in our work.

Getting back to the most important ones, the clustering coefficient (or transitivity) measures the density of triangles in the network - that is, the proportion of groups-of-three-interconnected-nodes (triangles) with respect to number of triples (set of three nodes that can be reached from all other, directly or indirectly) [New01a]. If we denote by $N_{\triangle}$ the number of triangles in the network and by N_3 the number of triples, then:

$$CC = \frac{3N_{\triangle}}{N_3} \tag{2.6}$$

more precisely:

$$N_{\triangle} = \sum_{k>j>i} a_{ij}a_{jk}aik, \quad \text{and,} \quad N_3 = \sum_{k>j>i} (a_{ij}a_{ik} + a_{jk}a_{ji} + a_{ki}a_{kj}) \tag{2.7}$$

A clustering coefficient close to 1 indicates that all triangles are closed, which translate in network language the well known social rule of 'the friend of my friends are my friends'.

On the other hand, the Information Content (IC) assesses the presence of mesoscale structures in its broad term [ZSM14], by looking at recurrences within the adjacency matrix of a network. Basically, it consists in iteratively merging pairs of nodes with the lowest loss of information possible. More in details, it identifies the pair of nodes (where both generally share a similar connectivity pattern) that would yield the smallest loss of information, from a Shannon information theory perspective, when both are merged together. The IC is then computed as the sum of all the information lost at each step until the network is shrunk into a single node. Finally, the value is normalised against the average value obtained on an ensemble of random networks. This metric somehow represents the quantity of information encoded into the proper structure of the network. A low information is representative of regular topologies where connectivity pattern are repeated, in other words, of meso-scale structures.

Macro-scale metrics

Only metrics belonging to the macro-scale family remain to be described. They are defined as 'macro' because they account for the overall flow of information within the structure of the system. There are many examples to illustrate such notion, as the number of steps required to go from one extreme of the network to the other side, or the role that a node plays in the propagation of the information in the system. Two of them have already been described as a natural prolongation of a micro-scale metric, namely, the entropy of the degree distribution and the assortativity.

An additional basic macro-scale metric consists in calculating the number of jumps necessary to go from one node to another (on average). It is called the average geodesic distance, denoted here g:

$$g = \frac{1}{\mathrm{N(N\text{-}1)}} \sum_{i \neq j} d_{ij} \tag{2.8}$$

This intuitive definition presents the important disadvantage of diverging when the network is disconnected, as it will exists pairs of nodes i and j from two disjoint parts of the network for which the distance between them d_{ij} is inf. To solve this problem, [LM01] proposed to consider the inverse of the harmonic mean of the distance, and called the metric 'global efficiency':

$$\mathrm{E} = \frac{1}{\mathrm{N(N\text{-}1)}} \sum_{i \neq j} \frac{1}{d_{ij}} \tag{2.9}$$

The name comes from the fact that it quantifies the efficiency of the network in transmitting information from one node to another, under the assumption of a non weighted cost of transmission - that is, that the total cost of transmission is proportional to the distance between the pair of nodes considered.

Finally, we can mention the diameter D of a complex network, that is, the greatest distance

between any pair of vertices, defined as:

$$D = \max_{i,j}(d_{ij}) \tag{2.10}$$

Benchmarking

The attentive reader will have noted that some of these metrics are strongly bound to the structure of the network and/or to its number of link. As such, a comparison between two networks can become quickly irrelevant when different conditions are considered. Therefore, in order to benchmark metrics across heterogeneous networks, it is necessary to normalise them. This is usually performed through a normalisation against a reference model, *i.e.* the value obtained in average in random equivalent networks. Let us explain better this normalisation. The idea is to create an ensemble of random network - called Erdös-Renyi graphs (ER) - of the same number of links and nodes. The metric is then computed over all the randomly created networks and averaged - representing therefore the value that would be expected for the metric in a network equivalent to the studied one (*i.e.* with the same number of nodes and links). The normalisation is then performed through a Z-Score:

$$m^* = \frac{m - \mu}{\sigma}, \tag{2.11}$$

where μ represents the metric average across the random networks, and σ being the corresponding standard deviation.

With this definition, a Z-Score superior or inferior to 0 respectively indicates a value that is higher (respectively lower) than expected. That way, two heterogeneous network could be benchmarked against one another through the corresponding Z-Scores.

2.3.3 Structural vs functional networks

Because of the interactive nature of complex systems, the concept of an accurate rendering of a graph is not as straightforward for implicit interactions as it is when physical and explicit links exist. This yielded the distinction between structural (also called physical) and functional networks.

As we have said, physical network describe explicit interactions, as such, their construction is tantamount to mapping the connections of the real system into the intended links. An adequate illustration is provided by the air transportation system, where airports are connected by physical flights and therefore constitute a physical structure. But in some real-world situations there might not be available - or at least explicit - information about the links: what is available is a certain environment where each element dynamics depend in an unknown manner on the other element dynamics. The underlying topology of such system must be extracted by deriving the unknown functions that connects the nodes between them, thus the name 'functional networks'.

In continually drawing a distinction between structural and functional network, we do not want to understate the importance of non-explicit information. Revisiting the aforementioned air transportation system, we might want to understand how delays propagates through the system - that is, how airports transfer information (here, delays) to other regions. This is of course essential for the complete understanding of the air transport, and gives birth to the need of a functional network representation connecting two nodes (pair of airports) when a shared delay dynamic (or synchronisation) is identified [BS09, PZMB14].

As we shall explain in this section, these non-intuitive relationships — the unknown functions driving the dynamics of a system — are more substantive than they may *prima facie* seem, and their representation is usually procured in a sequence of three steps described in Fig. 2.10, namely:

The synchronisation metric This is a measure - whose choice depend on the phenomenon under study - assessing the presence of a given relationship between the dynamics of two elements of the system.

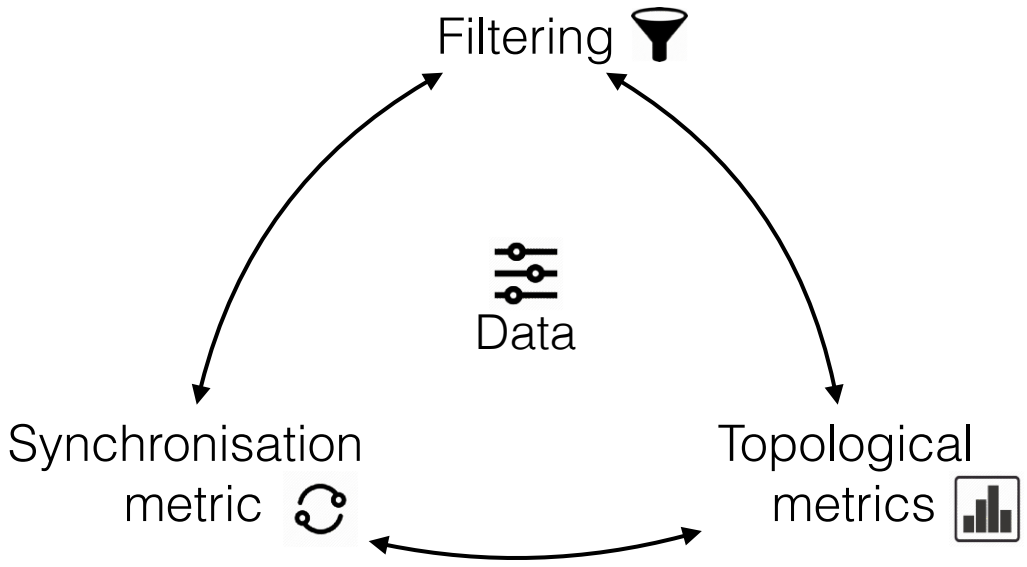

Figure 2.10: Different steps of functional network reconstruction: selection of a synchronisation metric, filtering of the results and creation of a binary network, extraction of knowledge. See the main text for more details.

Filtering Depending on the chosen synchronisation metric, it is customary - although not essential - to eliminate low statistically significant links or to filter values in function of a threshold. Moreover, the binarised version of the network, *i.e.* by discarding weights, is usually considered.

The topological metrics Metrics, as the ones described in Section 2.3.2, are extracted to describe the structure of the resulting networks.

Note that some users might bias the outcome of a study by using what may subjectively seem a 'good' threshold, based on his or her own experience during the filtering phase; to later find that such choice was far from optimal. This should therefore exhort an infrequent - or at least cautious - use of arbitrarily chosen thresholds. It must also be said that a synchronisation metric is specific to a given problem. In other words, to render a given implicit aspect of a system, the synchronisation metric must be adapted to the characteristics of the system and of the problem under analysis. For instance, let us consider the problem of detecting delay propagation between airports, as will be discussed in Section 5. The study of linear vs. non-linear dependencies between delay dynamics - besides the distinct functional networks and resulting topologies that it yields - are based on two different questions, leading to distinct

but complementary conclusions (*i.e.* how does delay propagates in both linear and non-linear manner). The the synchronisation metric used in both cases cannot be the same, as the first must derive the linear part of the propagation while the second focuses on the non-linear part. Thresholds and topological metrics depend on the characteristics of the synchronisation metric, which in turn depend on the specific questions about the data (*e.g.* Is there a correlation between the dynamics? Are there causality hidden patterns?, etc.) that are considered. Indeed, if the synchronisation metric is a statistical test, threshold might possibly be the significance level (usually called α) chosen by the user (usually set at 0.05), while other non-statistical metrics measuring a propriety of the interaction between two nodes (*e.g.* the coefficient of a linear regression, etc.) might be filtered through another threshold (*e.g.* we can imagine keeping only coefficient superior to 0.5 if it shows an increase of an effect). The same rationale can be applied to the choice of the topological metrics. For example, it is impossible to apply motifs detection to an undirected causal network. This explains the interconnected elements of Fig. 2.10.

2.3.4 Recent trends

The more complex the environment, the more advanced ought to be the theory that studies it. This implies that, in order to render even more complex environments, science must be provided with new mechanisms - as we have previously seen in the case of functional networks, that can go beyond a simple structural analysis. In particular, complex network theory has recently been extended to incorporate concepts like temporal evolution and multi-layer structures into its definition.

The idea of multi-layer networks emerged by observing the high dimensionality of the connections between the elements of real-world networks - that is, that many systems can be decomposed into several smaller networks. The concept is easy to grasp with the help of some well-selected examples. One may simply consider social networks [VR07], in which people (or groups of people) are usually connected through simple links regardless of the wide range of possible communication channels that might bind a pair of people [Gof74]. One

can further consider Air Transportation networks [ZL13a], which can be considered as projections of different networks in which connections are operated by individual airlines or alliances [CGGZ+13, CZGG+13]. To solve these situations, the traditional network theory has therefore embraced a novel framework that allows the manipulation of multi-layer networks, that is, graphs where several layers of connections are considered [DDSRC+13, LKLG14, ZL13a]. As such, explicit information about the types of connections is incorporated into the network, allowing a more complete description of the system: channels of connections between the elements might be multiple, each layer representing the network corresponding to one and unique channel (*e.g.* relationship, activity, category, airline, etc.), and each node might have different connection at each level. For example, layers might represent the nature of the connection, in the case of social networks, friendship, vicinity, kinship, partnership; or even channels of communication between people, messages, twitter, email, etc.

Another trend entails the attention paid to temporal networks, *i.e.* networks whose edges are not continuously active. Communication networks perfectly illustrate such situations, as links between two people can suddenly be de-activated or activated as a function of their relation. The fickle - rather unpredictable - behaviour of the edges makes disease contagion or e-mail information diffusion studies challenging. Additional reviews on the topic can be found in the Literature [HS12, HS13].

2.4 Complex Network in Air Transport

As explained in Section 2.3, the theory and application of complex networks have experienced such a growth in recent years that is is not surprising that research in air transport has also been affected. It is true that air traffic system can naturally be seen as a complex network at various levels, where nodes would be airports, sectors, navigation points, etc. Therefore, complex network theory has enabled several studies within the air traffic system framework, aimed at improving our knowledge about the intrinsic dynamics of the system and subsequently improving our capability of managing and controlling it. For instance, the description of the

topological structure of the operational network is of great importance for the different airlines business strategies, for it can assess the alteration in passengers' mobility when direct or indirect connections are considered; or give insights about the adaptiveness of air transport through time to fluctuant passenger's needs or external economical fickle forces such as deregulation. A more advanced form of the theory could study the dynamics taking place on the network - that includes the propagation of delays in the airport network or the propagation of other disturbances (*e.g.* capacity violation) within the adequate network, as the sector network [ABL06].

As we have said, air transport is increasing worldwide and its growing challenges will inevitably be linked to complex network theory. We present here a review of its divers existing application.

2.4.1 Different types of air traffic networks

Many complex systems can naturally be represented by one or more networks, as they can be perceived from different level of interpretation. On can think of the human brain to illustrate such affirmation. It is possible to represent the brain as a set of nodes (neurons) connected by link (synapses), but also as a set of brain regions interchanging information. Similarly, the air transport system can be viewed from different angles, and therefore represented by different networks. This property of complex network theory allows to tackle different problems, focusing on the relevant information of each one of them. Confronted with this vast menu of possibilities, it is important to determine which networks to investigate as a function of the problem under study.

The majority of studies done so far have been performed in networks where nodes are airports [ZL13a]. Indeed, important problems in air transport research are related with the mobility of passengers or the propagation of delay. In those cases, traffic oriented details (such as navigation points, virtual space points through which the aircraft must pass) are irrelevant. A network where nodes (representing airports) linked whenever a direct flight between the two airports exists is far more adequate towards delay or mobility studies. A network connecting the navigation points of the airspace would unwieldy yield added knowledge about delay propagation

or passenger mobility. But even choosing airports to be the nodes of a network, several sources of information, like scheduling, types of flight or airlines, are disregarded by projection. To put that another way: airport networks can be naturally decomposed into many sub-networks by considering separately flights of the same airline or alliance, types of flight, etc. The difference between considering a single or multiple network sub-types is important. For instance, two nodes (representing airports A and B) might be connected considering all airlines but disconnected considering only a subset of airlines; this situation can arise provided A and B are connected through a indirect flight, composed of two direct flights of two different airlines. As such, looking at those differences might provide information about the strategy of different type of airlines (*e.g.* local airlines versus low-carrier). This multilayer properties of the air transport network have been extensively studied; for instance, [LC04, CZGG$^+$13, CGGZ$^+$13] investigated the interdependencies between sub-networks corresponding to different airlines. Multi-layer, or multiplex, networks are then defined as graphs composed of several layers, where different types of links (*e.g.* for each airline) might connect two nodes (airports).

An airport network, be it a single sub-network or a projection of various ones, has also a natural weighting scheme. Indeed, we said that two nodes are linked when the corresponding airports are connected through a flight. But the network in itself is a static representation of a dynamic process - that is - the information exchanged (here flights, but it might also be a more abstract notion like delays) within a given period is projected into the graph. Were we to consider a snapshot of the system, no information whatsoever about the traffic of passengers would be contained; it is precisely considering a larger time window that connections between airports occur, at different frequencies. The fact that the number of connections of two distinct pairs of airports might be distinct in the time period under study (*e.g.* there might be a larger flow of passenger, or a larger number of flights, between A and B than between B and C) might be specified into the graph by pondering the links, for example by using the frequency of connections or the volume of transported persons. Therefore the weighting scheme would be given by the number of flights, beyond themere binary connectivity. Note that graphs can be directional - that is, a link might go from A to B but not from B to A. That be told, in has been empirically observed that most airport networks are very close to be symmetric, *i.e.*

roughly the same number of flights (or passengers) exist from A to B than from B to A.

In practice, graphs in passenger-oriented networks (*i.e.* with airports as nodes) are seldom directional. However, in some cases it could be so; let us consider an example. Some flights cannot take off or land on time because of a delay in another flight: these are called reactionary delays. To investigate the aforesaid phenomenon, [CTZ13] used a directed graph in which a link is created between two airports whenever the Granger causality test (see Section 2.2.1) indicates that the delays suffered in one of them is caused by the delay in the other. Such approach narrows the representation to airports involved in reactionary delays; therefore the resulting topology yields knowledge about whether delays generated in certain airports propagate through all the system or remain confined within a relatively small region.

While we could stop here, as airport networks are the only type that will be used in this PhD Thesis, for the sake of completeness we here review some additional alternative ways of reconstructing air transport networks.

Crew networks Reactionary delays, *i.e.* delays caused by the delay of another flight, are partly due to the late arrival of passengers, or because of the crew itself [PMO13]. Thus, nodes can represent crews linked between them when sharing the same aircraft.

Sector networks The complex system conception of airspace leads naturally to its partition in hierarchical operational pieces - sectors - mainly for simplifying its management and optimise its capacity. Each sector is then managed by a pair of controllers, with flight plans tuned in order to not violate the maximum traffic load of each of them. The concept of network of sectors has recently been proposed in [GVC+14], where nodes are sectors, and links between two nodes are representative of a same flight passing through the two sectors during the considered time window. Community detection has been used on top of this network as an approach to improve airspace design.

Navigation points network Until technology will reach a maturity such to allow a safe free routing, aircraft have to travel through air routes defined by fixed navigation point. While more safe, this strategy implies the appearance to potential bottlenecks in central zones

of the airspace, where several aircraft converge. To improve our understanding of routing and airspace design, [GVC+14, KQJWBXB12] considered a new network where each navigation point as a node, linked when at least one flight goes directly from one node to the other in the considered time interval.

2.4.2 Topologies of airport networks

While a few works have studied the air transport system from a different perspective, as discussed in the previous section, the majority of the studies have hitherto been done considering airport networks. One may thus ask: what are their general characteristics? Do air transport (airport) networks have specific properties that define them?

As a first step in a basic investigation on the topology of a network, the most important indicator to be considered is the degree distribution, as introduced in Section 2.3.2, as it allows to highlight specific structural characteristics of the system - as the presence of hubs, etc. In the review performed by [ZL13a], a truncated-power law distribution was found in most networks, hallmark of the presence of "scale free" topologies. If such structure is a consequence of a preferential attachment mechanism, this may yield important information about the dynamics of the system's growth, which has shaped the current European hub-and-spoke structure.

2.4.3 Characterisation of delays

The analysis and characterisation of the propagation of delays is one of the most important research topics in air transport management, mainly because of three reasons: their profound implications on safety, when delays are not handled through ground programs [Duy93]; their negative contribution to the environment, with the additional emission of CO and NOx - for each minute of ground delay, 1 to 4 kg of fuel are consumed [CDLHJ07]; and their economical impact, through a reduction of the system's efficiency. The latter was estimated at up to 1.3 billion Euro to European airlines in 2007 [Ber08], and entails additional economical impact on passengers [CT11]. This is what fostered a special attention from the scientific community

to the study of delay-related phenomena. In this section, we propose to review the major contributions hitherto appeared.

On the economic level, some studies used data to build additive models that quantify the impact of delay costs [CTJL09, FKHS13, CT11]. This was done with a high level of granularity: costs are partitioned into independent components, according to experts' opinions on their sources (*e.g.* extra crew cost, passengers delay cost, fuel costs, etc.). The exact models behind each component of the final cost of delay remains unfortunately obscure to the reader in Europe, due to the confidencial aspect of airlines data that supported these projects.

The assessment of delay causes [SR10, PWNT09, WT12] have also been studied through statistical models, as through Cox regression analyses, revealing key contributing factors to departure (*e.g.* turnaround buffer time or baggage handling, etc.) and arrival delays (*e.g.* weather, etc.), and assessing the quantitative impact of each factor on the suffered delay.

Delay forecast has been the next naturally tackled topic in the Literature. [RB14] proposed a network-based air traffic delay prediction model. Specifically, a complex network representation of US airports has been used to cluster these into different categories (according to the delay usually suffered), for then using the outcome of the classification to predict delays using a random forests regression model.

The mitigation of delays have been extensively studied through different strategies. [DP09] proposed a comparison between the cost of speed variation, that an aircraft would activate to mitigate a delay, and the cost of the delay itself, showing the existence of a changing point where one does not outweigh the other. Other strategies - still based on cost optimisation - rethink the schedule of aircraft rotation, as the operational reliability is directly linked with delays [WC02]. [NPRN14] proposes to refine the computation of Calculated Take-Off Times (CTOTs) in order to maintain the pre-allocated slots and avoid delay creation.

Of special relevance are the studies about reactionary delays, that is, the propagation process according to which one aircraft's late arrival is the cause of the delay of the following flight. [PMO13] tried to estimate the trends in air traffic delay propagation using a queuing network

model with the busiest-35 airports in the US, only to find out that the demand profile of airlines might change as a function of the expected suffered delay. [FRE13] proposed an agent-based model that reproduced empirically observed delay propagation patterns in US, identifying passenger and crew connectivities as the most relevant factors driving delay propagation. Finally, [ACGB08] investigated the relationship between schedules and delay propagation, using flight data of two major US airlines. They created a tree-structure, representing the expansion of each singular delay, to extract the characteristics of the propagation. Among their findings: that delay generally originate at a hub and disappear at the next hub, and that fewer delays are generated in the afternoon but they propagates longer.

It is also worth mentioning the studies shedding light on the appearance of ATC (Air Traffic Management) related delays, as those due to regulations caused by the limited capacities of airspaces [Glo96], and to the maximum number of operations that airports can handle [Han02].

2.4.4 Resilience

Resilience have been originally introduced in material science and ecology to study the behaviour of the system after the appearance of new species or a significant lasting abnormal weather [Gun00]. It has then been naturally expanded to other fields, as management or engineering, under the name of "resilience engineering". The value of resilience in management is intuitive, as it refers for example to the capacity of people to handle complex tasks when exposed to pressure [HWL07].

More generally speaking, resilience would be the capacity of the system to recover its normal functions after internal or external disturbances. The other side of the coin in a resilience analysis is the information gained about the weak points of the system. This is called vulnerability, which would designate the inability to withstand a disturbance.

These notions have been deeply studied by the complex network community, being of special relevance the particular sensitivity of scale-free networks (including Internet, Air Traffic Management networks, etc.) to the removal of high degree nodes [AJB00, CLMR03]. This is of

special relevance when compared to the resilience of random Erdös–Renyi networks, which is fostered by the rather homogeneous role played by each node.

Resilience have been brought into the context of ATM in 2009 under the impulse of EUROCONTROL, when they recognised the crucial importance of designing strategies to mitigate the propagation of delays or other disturbances [EUR09]. The resilience of the ATM system has therefore been extensively studied [Glu12], looking at the appearance of disturbances as diverse as severe adverse weather [KR09, Jan15] or terrorist attacks [GCLR06], and their subsequent possible effect on the system [ZL13b, CZGG+13, WSC15, DW16].

[CC04] analysed how random failures or targeted terrorist attacks might corrupt the topological properties (average degree, clustering coefficient, diameter and efficiency) of the US system. Such analysis consists in removing airports at random, thus simulating extreme disturbances as emergency situations or critical weather; its results showed a real sensitivity of the topology of the system to the airport removal. While the study is well founded, it is based on an even deeper problem that we will tackle in Chapter 4, as airports may be de-activated because of an attack but also because of a lack of data availability. [HARA13] further used a similar approach to assess the network's resilience to flight path disruptions, that is, flight paths that might become unusable for instance because of extreme weather.

Finally, when thinking about disruption in ATM, the Eyjafjallajokull volcano eruption of 2010 easily comes to mind. [WDM12] explains the severe consequences in terms of a truncated scale-free network and the degree distribution of the ATM system. Such analysis is also related to our work, as discussed in Chapter 4.

Chapter 3

Data Description

3.1 Main data set description

Most of the studies performed for this PhD Thesis have been done using time series extracted from the Flight Trajectory (ALL_FT+) data set, as provided by the EUROCONTROL's PRISME group. The data set encodes information about planned and executed trajectories for flights within or crossing the European airspace. It must be noted that general aviation and leisure aircraft are excluded. This data set contains different types of information about each flight, the most relevant being:

- Flight ID, aircraft type, departure and destination airports and airline operator.
- When relevant, information about regulations assigned to the flight [1].
- A description of several types of profiles (*i.e.*, trajectories) for each flight, *i.e.* the last filed flight plan, the last regulated trajectory, and the real trajectory reconstructed through radar information.
- Finally, the scheduled and real take-off and landing times are provided.

[1]Note that this information has not been used in this PhD Thesis, and is here included for the sake of completeness.

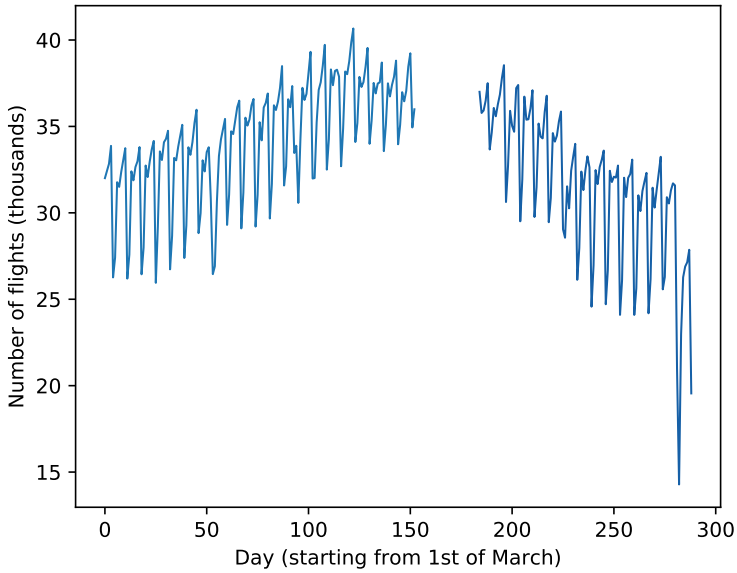

Figure 3.1: Evolution of the number of flights per day. The data set covers the 1st of March up to the 19th of December.

The data set covers the whole European airspace for a period ranging from the 1st of March 2011 to the 19th of December 2011. However, within this time window, several days do not include radar trajectories, only the last filed flight plans being available. These days have been discarded in the following studies and can be identified in Fig. 3.1 as the empty time segments, this latter figure depicting the evolution of the global number of flights included in the data set as a function of the day. The reader can appreciate the strong variability in the number of flights, corresponding to the weekly and annual seasonality.

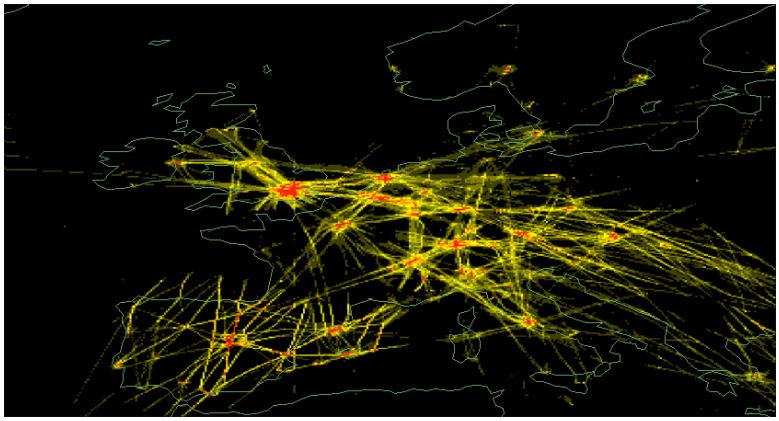

Figure 3.2: Map of the density of flights over European airspace, according to the ALL_FT+ data set.

It is also of interest to visualise the geographical coverage of the data set. Fig. 3.2 represents a map of density of all flights, without including any filtering or pre-processing; the flight density in each region is calculated through the number of trajectory points in it included.

It can be noted that flight information is missing in some specific geographical zones. To illustrate, this is especially evident in the west part of France, where radar information is missing and flights seem to disappear. These kinds of problems must be taken into account before starting any kind of analysis, specifically as the scope of this PhD Thesis is not focused on the improvement of the available data sets - *e.g.* through segments interpolation. As a first step, the data set has thus been pre-processed, in order to discard all flights presenting one or more of the following abnormal patterns: (a) incomplete radar trajectory (generally the case of intercontinental flights); (b) segments with higher-than-sound speeds (suggesting errors in radar data); (c) two consecutive radar points separated with more than 30 minutes (missing radar segments). These filters reduce the number of available flight by a 46%. The resulting spatial distribution is displayed in Fig. 3.3, where it can clearly be seen that some regions (*e.g.* west France coast, north-western part of Spain, and Turkey) are depicted in grey, as no flights have passed the aforementioned filters. Most of the remaining flights are concentrated within central Europe, among others due to the better radar coverage and the high traffic of this air

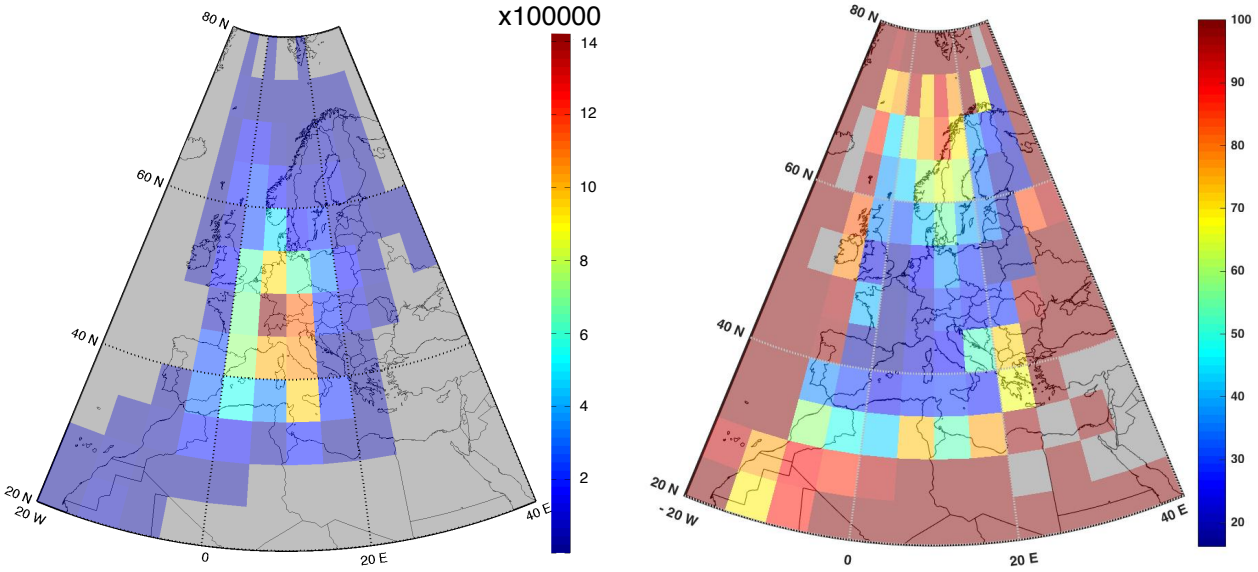

Figure 3.3: Spatial plot of (left) flight density after filtering and of (right) the percentage of discarded flights, for the ALL_FT+ data set. Reprinted with permission from [BPZ16].

space.

To summarise, ALL_FT+ data give use access to the trajectories of flights (both the actual flown and the scheduled one), which - considering the respective departure and arrival times - allows to extract exact delay information.

3.2 Complementary data sets

Additional data sets have been used to complement the information obtained through ALL_FT+. Specifically, data covering other airspaces will be required to create benchmarks and better evaluate the European performance. To this end, two complementary data sets have been considered:

On-Time Performance data set. A data set covering the United States. It has been obtained from the Research and Innovative Technology Administration (RITA) of the United States Department of Transportation. It includes reported detailed delay information (including the causes) for the non-freight US flights of the 16 US airlines that generate more than 1% of passenger revenues.

OpenFlights. A open source repository of worldwide regular flights and airports data. Freights are not included. No information about routes nor delays are available. It has been accessed at `http://openflights.org` on August 17th of 2015 to extract the airport and connection information of Australia, Brazil, Canada, China, Europe, India, Russia and the US.

Tab. 3.1 reports all the data sets that we considered throughout this PhD Thesis, and specifies some basic statistics and available features - along with the number of airlines and the number of airports covered, the availability of aircraft types and time measures (as delays) is indicated.

Finally, we have been awarded access by the Chinese Aviation Communication Corporation (ADCC) to a Chinese delay-related data set. This access was nevertheless constrained to the

Country	Data set	No. of airports	No. of airlines	A/c avail.	Time avail.
Australia	OpenFlights	112	12	No	No
Brazil	OpenFlights	119	12	No	No
Canada	OpenFlights	204	24	No	No
China	OpenFlights	185	17	No	No
Europe	OpenFlights	497	153	No	No
India	OpenFlights	71	8	No	No
Russia	OpenFlights	104	36	No	No
US	OpenFlights	595	81	No	No
Europe	ALL_FT+	1854	100	Yes	Yes
US	RITA	286	16	Yes	Yes

Table 3.1: Summary of data sets' characteristics. Reprinted with permission from [BCPZ16].

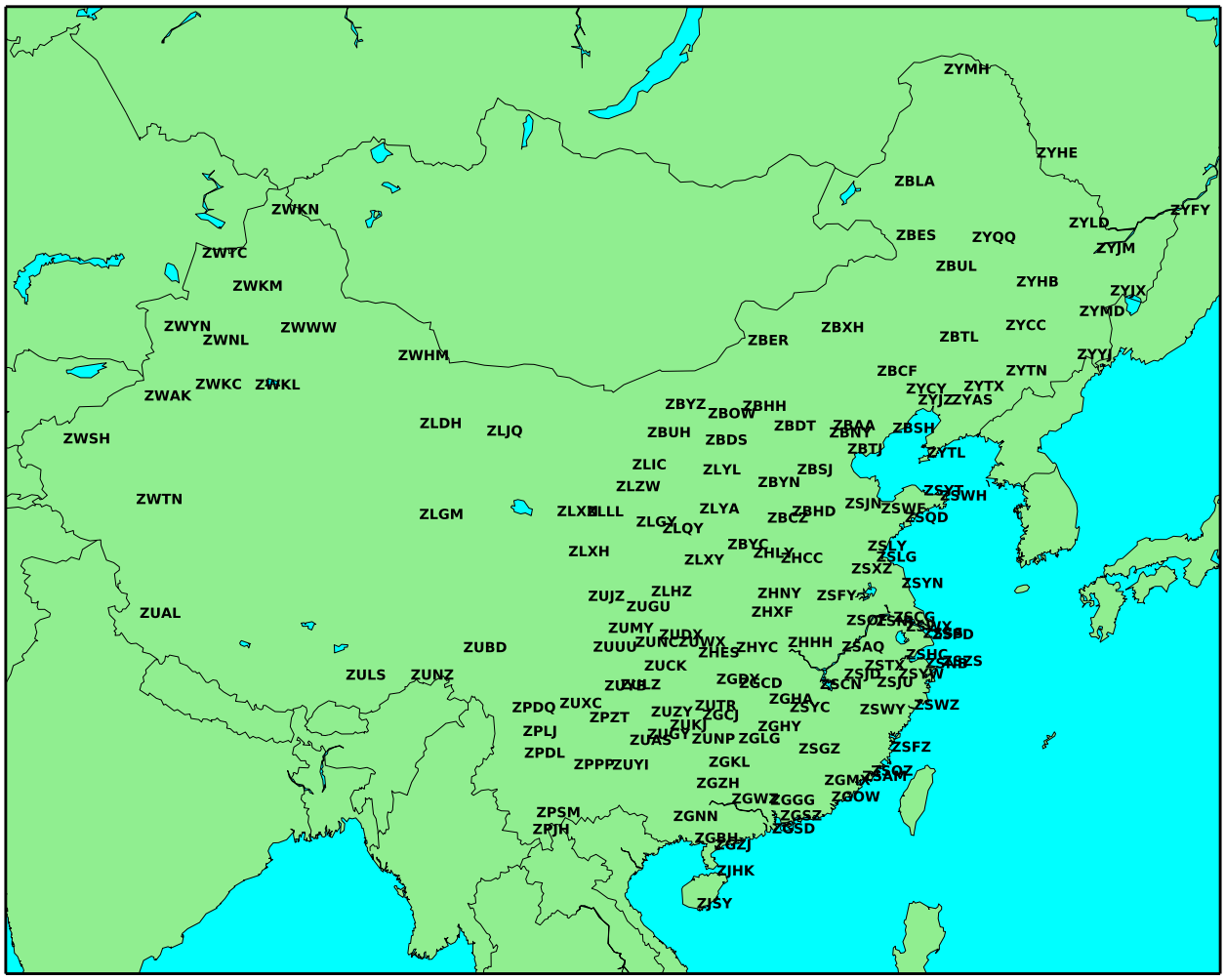

Figure 3.4: Map depicting the position of the 152 airports included in the ADCC's Chinese flights data set. Reprinted with permission from [ZBY16].

development of a specific case study that will be presented in Section 5.1.3, and because of its specificity it has not been included in Tab. 3.1 - note that the data set has afterwards been deleted. It contained high-level information for flights crossing the Chinese airspace from the month of August 2015. Specifically, it provided the following data for each flight: (a) schedule and real departure and arrival times; (b) airline carrier code; (c) aircraft type. To match the needs of our approach, some pre-processing have been executed, including: only domestic flights have been considered - that is flights departing from and arriving locally; and flight from or to non-ICAO airports (denoted *ZZZZ*) have been discarded. See Fig. 3.4 for a graphical representation of the geographical region considered. The final data set comprised 152 airports and 45.151 flights after filtering.

Chapter 4

Structural network representation

As previously introduced, many real-world complex systems - like the internet or the human brain [COJT+11] - have benefit from the theory of complex network to gain all-purposed knowledge about their structure and the corresponding dynamics. Air transportation has been no exception in that sense, as multiple studies buttressed by complex network theory endeavour to extract meaningful knowledge, with examples ranging from simple topological analyses and their evolution through time, to the embroilment of air transportation into epidemic spreading, through studies about the resilience of the system to external attacks or the assessment of the dynamics of passengers [ZL13a, LSS14].

Creating a network representation of a given system consists in (a) mapping the elements under study into nodes and (b) establish links between pairs of nodes when a given relationship is detected between them [PZMB14]. Such process (not that straightforward for functional networks, see Section 2.3.3) might *prima facie* seem trivial in the case of air transport system, as it would suffice to assign a node to each airport and link them if a direct flight or passenger flow or any chosen entity flow (either inmaterial or physical) is detected between them. However, the definition of nodes and links is knottier than initially thought for two main issues.

The first reason behind the difficulties in air transportation representation are linked to the customary prior sampling process, that is, that only a subset composed of the most-connected airports is considered, or that connections correspond to a subset of airlines. The above-

mentioned sampling is mainly purposive, as there might be a need to reduce computational costs; or merely a will to focus on a specific airline or airspace. The sampling might also be caused by the reduced availability of data (small airports or airlines are typically less prone to have clean and available data) or spawn from a bias within the data (which may be related to the way it was collected, *e.g.* different airlines might have different reporting guidelines). A simple look into the literature serves to demonstrate our point. For example, China counted 128 airports and 1165 connections in [LC04], 144 and 1018 in [WMWJ11], and 203 airports and 1877 connections in [DZL+16] - not to mention the additional intermediate values that can be found in [WW12, ZCDC10]. This can also be noted within the USA air transport network, which was represented respectively by 215, 272, 305 and 732 airports in [CWS+03, XH08, FRE13, JJ12]. Notably, in some cases, the number of airports is not even reported [ACNR07, Bag08]. It should be noted, though, that such counts are significantly dependent on how the data have been collected. In Tab. 4.1 has been resumed some information about the number of airport and airlines reported by the most common data providers. OpenFlights data are not historically available, but represents the only common source of data for comparison across all regions of the globe. It can be observed that data coming from 'official' service providers significantly differ from those of OpenFlights. For Europe, ALL_FT+ data covers four times more airports than OpenFlights but with a third less airline coverage. In the US, a similar pattern happens with RITA data covering half as many airports as OpenFlights with only 16 (largest) carriers.

	Europe	US	China
data source	OpenFlights [1]	OpenFlights	OpenFlights
No. of airports	497	595	185
No. of airlines	153	81	17
data source	ALL_FT+ [2]	RITA [3]	N/A
No. of airports	1854	286	-
No. of airlines	100	16	-

Table 4.1: Flight data sources comparison, by region. Reprinted with permission from [CBZ16].

[1]Open source repository, flights and airport data, worldwide coverage; `http://openflights.org` Flights for June 2015

[2]Provided by EUROCONTROL; all intra-European IFR flights for March through December, 2011.

[3]On-Time Performance data, provided by the Research and Innovative Technology Administration (RITA), US DoT. Intra-US flights, same period as (2).

Those numbers also suffers from additional biases, as 609 European airports are 'officially' reported within the European Civil Aviation Conference (ECAC) area (using data from ACI[4] EUROPE for 2014), and this number jumps to 3347 when all ECAC IFR[5] flights are included - this includes military, cargo and general aviation. The US suffers from a similar bias as Federal Aviation Administration (FAA) data[6] accounts for more than 5000 airports in 2014, as compared with the 595 treated by OpenFlights. China, on top of this long tail bias, also suffers from the creation of new infrastructures, as nine new airports - at Heilongjiang Fuyuan, Hubei Shen-nongjia, Qinghai Delingha, Shanxi Lüliang, Jilin Tonghua, Guangxi Hechi, Sichuan Aba, Guizhou Liupanshui and Hu'nan Hengyang - have been constructed between 2013 and 2014 [oC16]. Those reporting heterogeneities within the data sources, coupled with inharmonious implemented sampling strategies of the proper studies, explain the contrasts between various representations of the air transport network.

No matter the cause behind the data bias, it have important consequences on the observed topological features - that is, some properties may disappear or emerge in a spurious way [SWM05, LKJ06] - which have been largely neglected by the research community in air transport. In the above-mentioned example of China, the clustering coefficient varied between 0.69 [WMWJ11] and 0.73 [LC04]. A more startling example is the case of the Italian network, which was reported to have clustering coefficients between 0.07 in [GM07] and 0.42 in [ZBCB08].

Although the data collected in this PhD Thesis refer to different years or periods, a number of broad comparisons are going to be used to demonstrate the effects of sampling on the insights extracted from the system.

The second problem that one might encounter when dealing with air transportation system is its multi-dimensional nature [Nea14]. Fig. 4.1 resumes the problem encountered when neglecting the multi-dimensionality of a system - that is projecting all its dimension into a single-layer network. Imagine the two scales of grey networks (in the left part of Fig. 4.1), being layers representative of two different airlines. Then, two airports that might not be connected in one

[4]Airports Council International

[5]Instrument Flight Rules.

[6]`http://www.rita.dot.gov/bts/sites/rita.dot.gov.bts/files/publications/national_transportation_statistics/index.html` Accessed May 2016.

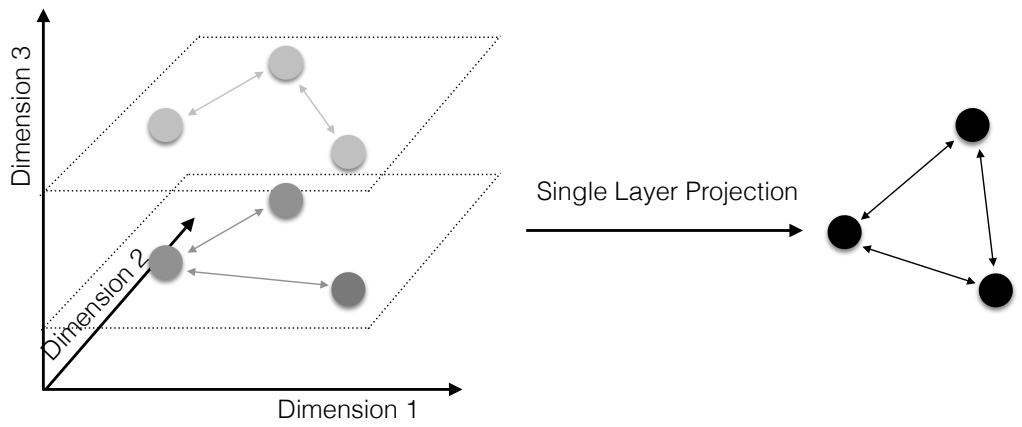

Figure 4.1: Projecting a multi-dimensional system. The original multi-layer system (left), when represented as a single (right) by discarding dimensions, suffers from a loss of information.

of the layer of the multi-dimensional system are however linked in the projected single layered representation - therefore causing a loss of information about the system structure. Within the air transport system, several dimensions may be considered. Some of them come intuitively to mind, like the aforementioned airline carriers, and have thus already been considered in past research [CGGZ+13]; others, like aircraft types or time windows, have mostly been neglected. Note how this second problem (*i.e.* the multidimensionality) is partly connected to the first one (*i.e.* the sample bias), as discarding some dimensions is tantamount to sample links - and not nodes - according to some *hidden/unknown* variables.

The idea in this chapter is to merge both ideas to assess if and to what extent a sampling process biases the topological and dynamical properties of the system's representation, with respect to what would be obtained if the complete system was considered. This comparison will be performed for airports, airlines, aircraft type and time window sampling processes, on eight of the most important air transport networks (Australia, Brazil, Canada, China, Europe, India, Russia and the USA). Additionally, the effects of such sampling will be assessed on delay related metrics for the European and the US Air Transport Management (ATM) system - being the two regions for which time related information was available (see Tab. 3.1).

4.1 Regional contest

4.1.1 Establishing context

The influence of sampling - whether willingly applied or not - on static topological structures or in ATM performance metrics is here assessed. This study is conducted on various regions of the globe and while almost all regions considered (Australia, Brazil, Canada, India, Russia, China[7] and the US[8]) are straightforward to define, it is slightly more complicated to delineate which countries does Europe refer to. Europe, in the sense of air transportation data, may refer to the geographical European Union (EU), but it might as well refer to the area managed by EUROCONTROL, which includes 44 participating countries of the European Civil Aviation Conference (ECAC). The European area defined within the Single European Sky (SES) project (launched in 2000 by the European Commission specifically to tackle the challenge of increasing delays) is in turn composed of the 28 EU members plus Norway and Switzerland. So, there are four different manners to define 'Europe', with the geographical definition being the 'smallest' one, forcing caution when referring to the 'EU' data. Turkey, for example, is in ECAC and a member EUROCONTROL, but not in the EU nor in SES. This situation leads to a certain fuzziness, as Turkey has been noted in 2014 to be the main contributor to 'European' (in EUROCONTROL sense) traffic growth, while, in contrast, remaining unaffected by the costs charging methods imposed within SES [Com15].

For the context of this PhD Thesis, 'European' data will designate the countries of the ECAC, being the area commonly used for ATM delay performance assessment, which is precisely the primary focus of our work. As it will be demonstrated in this chapter, user defined features like the number of airports or airlines (within a defined airspace area) significantly alter performance metric assessment.

[7] Air transport movement data for China often include Hong Kong, Macao and Taiwan but does not account for their airports.

[8] Air navigation services provided by the Contiguous United States (US CONUS', *i.e.* the 48 states located on the North American continent). Alaska, Hawaii and Oceanic areas are excluded.

4.1.2 Market structure

Europe, Australia, Brazil, Canada, India, Russia and the US are all established free markets, with a variety of operator types and with a very limited state intervention in airline planning and operations. Recently, a significant growth in low-cost carriers (LCCs) [Com15] as been witnessed, while the other airlines adopted a merging/grouping strategy, with most of the largest companies now operating as airline groups. LCCs emergence into the market is exemplified in Tab. 4.2 by their presence within the top four airlines (ordered by volume of passengers carried) in both Europe and the US.

In this global context, China is an isolated case, as it is mainly a state-controlled system [CLZ15, CF12, ZR09]. Chinese airlines were merged into three large airline groups (Air China, China Eastern and China Southern) in 2002. Regional airlines - controlled and supported by government - emerged as supplementary carriers on approximatively a quarter of all routes (this having consequences on the hub development in China, as it will later be observed). This state-led system creates obvious disincentives to the emergences of LCCs, as larger airlines benefit from state welfare with some route and schedule advantages.

[9]International Airlines Group, formed by the merger of Iberia and British Airways.

Region	Four largest carrier (groups)	Alliances	Carrier type	Ownership
Europe	Lufthansa Group	Star Alliance	Mainline	Private holdings
	Ryanair	No global alliance	LCC	
	IAG[9]	**one**world	Mainline	
	Air France-KLM	SkyTeam	Mainline	
US	American airlines	**one**world	Mainline	Public companies
	Delta Air lines	SkyTeam	Mainline	
	Southwest Airlines	No global alliance	LCC	
	United Airlines	Star Alliance	Mainline	
China	Air China	Star Alliance	Mainline	State shareholding
	China Eastern	SkyTeam	Mainline	State shareholding
	China Southern	SkyTeam	Mainline	State shareholding
	Hainan Southern	No global alliance	Mainline	Private holdings

Table 4.2: Carrier market structure by region. Reprinted with permission from [CBZ16].

In Tab. 4.2, we describe the extracted information about the market structure in the three biggest regions under study (Europe, US and China) from Flightglobal data[10]. As a side note, this market structure is currently evolving and changing, as we witness the dissipation of the demarcation between mainline companies and LCCs. Many examples of LCC emerging from mainline carriers (Vueling from IAG, Transavia from Air France-KLM, Eurowings from Lufthansa, Delta and United US groups also possess LCC ownership) can nowadays be found, blurring the line of the potential future market structure. In the case of China, the first three airlines do not have LCC ownership.

4.1.3 Flow management practices

The effect of sampling on Air traffic management (ATM) performance measurement will be assessed only for Europe and the US, being the two regions for which data containing delay related information have been obtained (OpenFlights data do not contain delay information, see Chapter 3). Therefore, this section will focus on describing the flow management practices of those two specific regions.

There is a fundamental difference between Europe and the US on their handling of the aircraft flow and delays. Europe adopts a more strategic approach, with planned routes and adaptive airport slots. Specifically, due to its political definition (see Section 4.1.1), Europe has to deeply resort to at-gate holdings because of the implemented ground delay program (GDP). Indeed, the fragmentation of the airspace in various service providers from different sovereign states - hence, the Single European Sky project - makes the versatile US management impossible to clone. In the US, the presence of a unique service provider (Federal Aviation Administration - FAA) allows for the redirection of airborne flights or even the re-routing of an entire flow of flights around a bad weather spot. Therefore, the GDP in the US is left as a last resort solution. On the opposite, Europe rarely uses en-route spacing, except for collision avoidance, and instead prefers to manage conflict through strategic tools (*e.g.* Calculated Take-Off Times - CTOTs - re-allocation) and speed control.

[10] `https://www.flightglobal.com/`. Accessed May 2016

Those differences have consequences on how airlines reports delays, as on-gate holdings can be originally caused by weather, congestion or any other kind of problems, but will be reported as Air Traffic Flow Management (ATFM) control cause; creating therefore differences between Europe and US delay analysis.

4.2 Sampling and performance analysis

4.2.1 Network topology as a function of sampling

Sampling airports

The structure of air transport network is well-known to be scale-free [ZL13a] (*i.e.* with a long-tailed degree probability distribution), characterised by few highly connected nodes (hubs) and a large number of secondary nodes (regional airports). This structural particularity of air transport network is confirmed in Fig. 4.2, in which the normal and cumulative distribution of degrees have been displayed for each of the 10 air transport network considered.

The sampling properties of scale-free networks are well understood [SWM05]. However, the sampling is usually performed randomly, which is not the case in air transportation networks, where large airports are over-represented before smaller ones. Such is the case whether the sampling is purposive (*e.g.* select the most connected airports to lower the computational cost) or reflecting the limitation of the data set (*e.g.* RITA dataset, where only airlines generating more than 1% of passenger revenues are bound to report their data).

In order to mimic the bias commonly present in air transportation network representations, we here introduce a sequential sampling process where nodes (airports) are successively added to the considered network according to their number of connections - highly connected airports first, then the sparsely connected ones. This approach allows us to display the evolution of the networks topology as and when nodes are being added - that is, from the core backbone structure (*i.e.* composed of the most connected airports) to the whole structure. From this, the

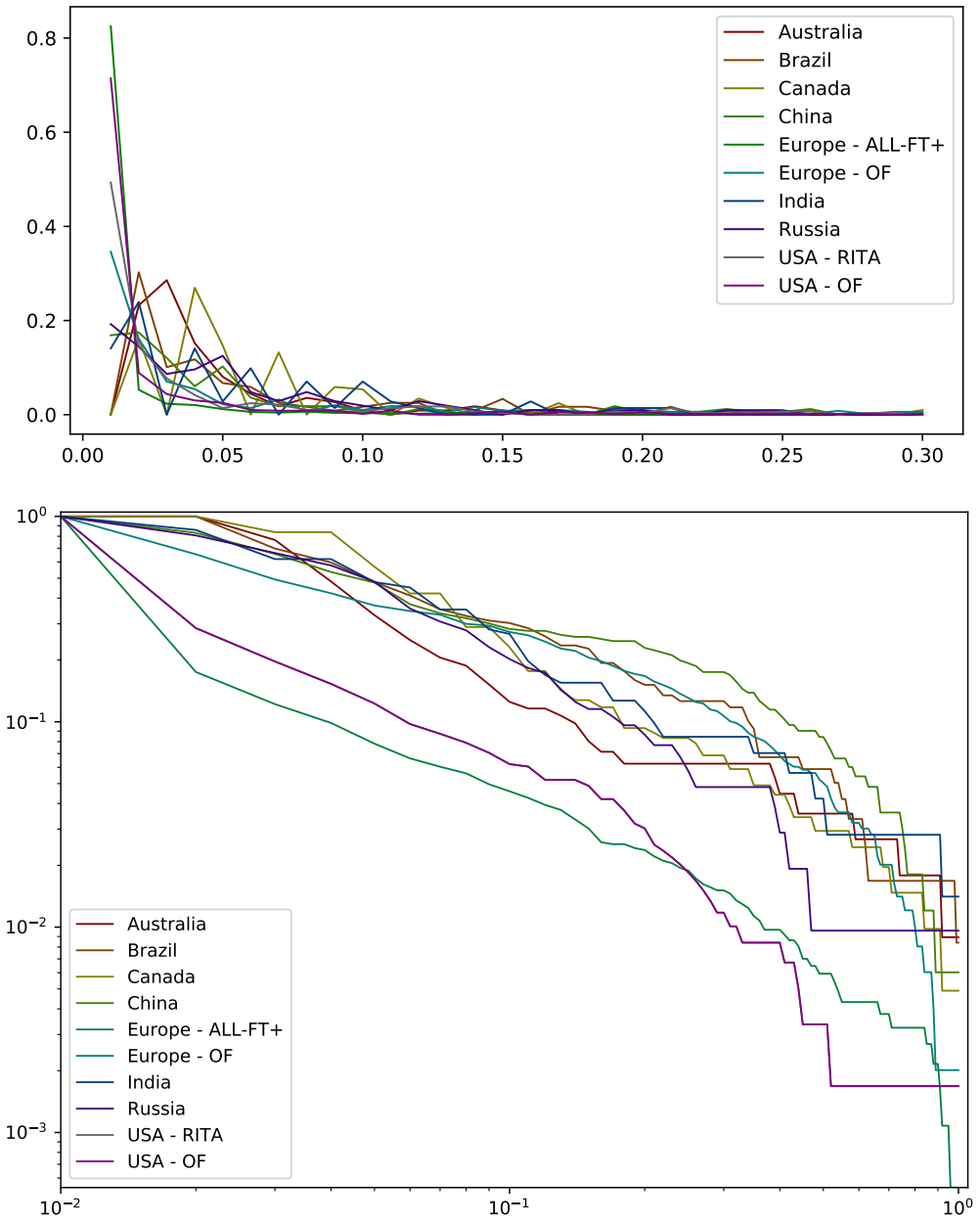

Figure 4.2: Normal (top panel) and cumulative (bottom panel) degree distribution of the 10 air transport network studied. Reprinted with permission from [BCPZ16].

error made when considering incomplete data sets can be assessed, for example when focusing on the most connected airports.

Fig. 4.3 represents the evolution of seven relevant topological metrics (link density L_d, maximum degree M_d, clustering coefficient CC, degree correlation D_c, entropy of the degree distribution E_{dd}, efficiency E and the information content IC - whose definitions can be found in Section 2.4.2) as a function of the fraction of nodes sampled from the original data set according

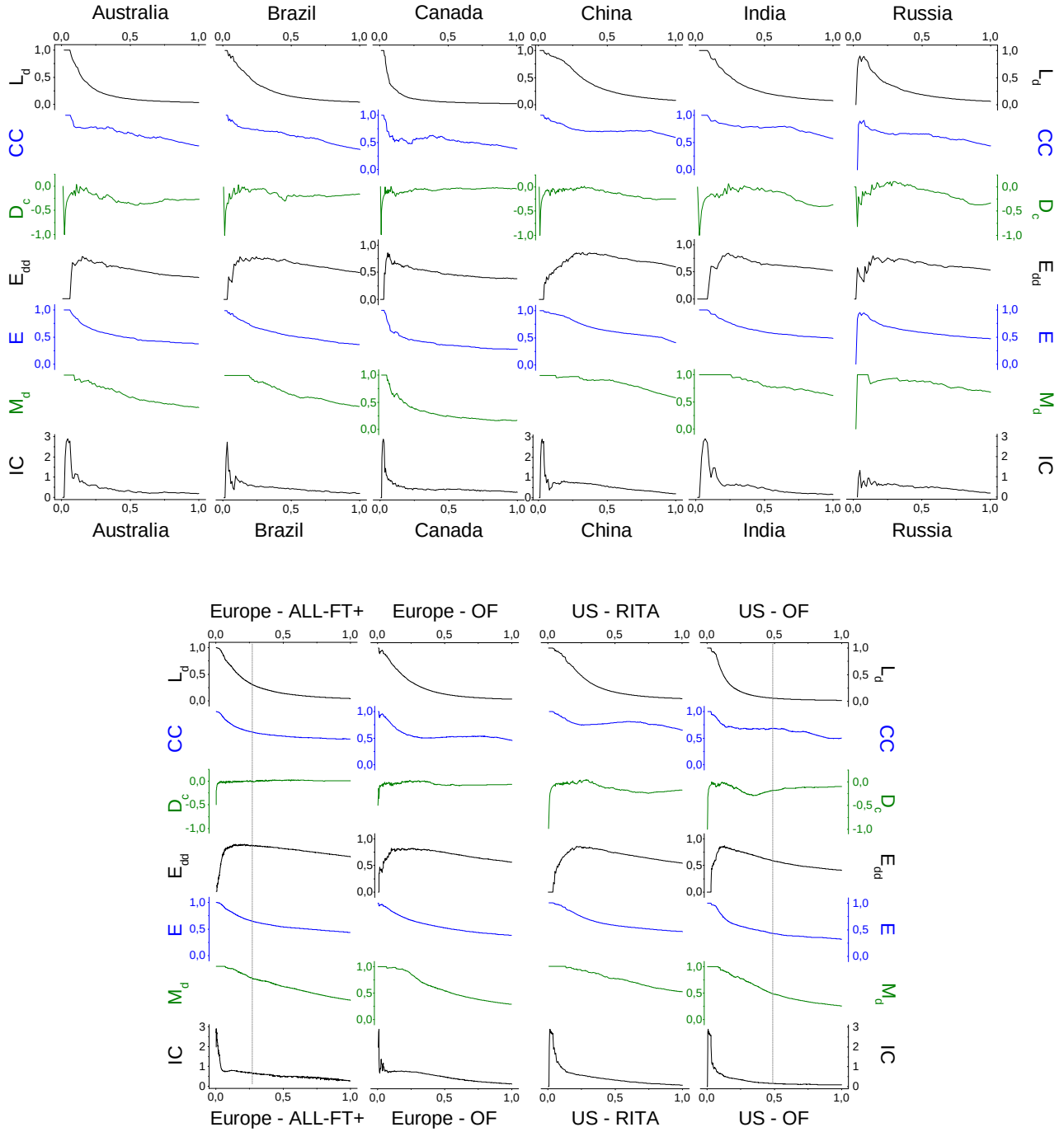

Figure 4.3: Evolution of network topological metrics, as a function of the fraction of airports included in the network. Airports are included sequentially in function of its number of operation (in a decreasing order). Thus, values close to zero in the x-axis describe the core bone of the network; values close to one signify that all airports are considered. Reprinted with permission from [BCPZ16].

to the aforementioned sequential sampling process.

The metrics typically vary quite strongly as a function of the number of airports included - the most notable example being the entropy of the degree distribution E_{dd}, which starts at

0.0, because the first considered airports are inter-connected, and stabilises when the totality of them is included. Additionally, whilst most of the metrics shown demonstrate a monotonic behaviour, some (E_{dd} and IC) change the direction of their evolution. Notably, many of them do not saturate - that is, that they do not reach a stable value even when all airport at disposal are included. All these observations - and more particularly the non convergence of the metrics - reinforce the idea that the network topology is sensitive to the addition of airports, even small ones, thereupon there is no obvious threshold by which nodes may be safely discarded when a sequential weighted (*i.e.* from highly to sparsely connected airports) sampling strategy is adopted. This might be surprising, knowing that past studies have demonstrated that such a threshold exists and is set to 0.6, *i.e.* sampling 60% of the nodes is enough to reach a good approximation of a full scale-free network (*e.g.* the Internet and the online pre-print repository arXiv, see [LKJ06]). This result does not translate to air transport network for two possible reasons: (a) the sampling strategy adopted in air transport not being random might cause a tardiness in the saturation of the metrics and (b) the actual number of airports is much higher than the ones considered by our data sets, therefore the complete dataset considered is not even representing the 60% of the existing airports. The 1854 reported airports in Europe by ALL_FT+ represents approximatively half of the real existing number of airports in Europe (3347) as mentioned in the introduction of the Chapter.

Besides, it is quite remarkable that China and India present a qualitatively different behaviour. While we had no particular information about India allowing to explain such pattern, we can

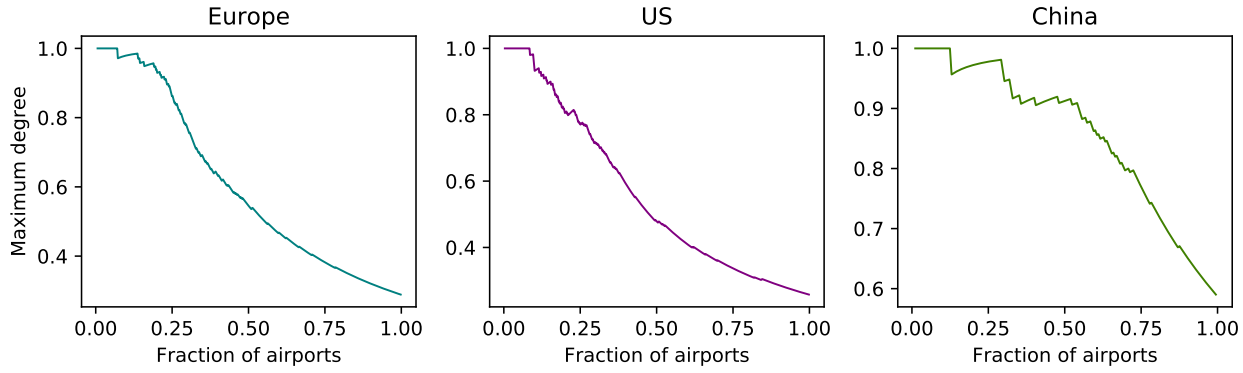

Figure 4.4: Maximum degree by airport sampling fraction for Europe, China and the US (OpenFlights data sets).

bring more explanations on the case of China. It may be observed in Fig. 4.4 that China ends up with a higher maximum degree in comparison with OpenFlights European and US data sets. Such a behaviour may be ascribed to the market structure of China (see Section 4.1.2), where the regionalisation of the country and the national policies have led more airports to have high connectivity.

Finally, let us go back to the information displayed at the bottom of Fig. 4.3, where the topologies obtained from two different data sources for the European and US networks are compared - that is, RITA and OpenFlights for the US and ALL_FT+ and OpenFlights for Europe. Roughly speaking, both pair of data sets present different characteristics - please refer to Tab. 3.1 for more details - as OpenFlights is more complete for the US but includes fewer airports than ALL_FT+ for Europe. In Fig. 4.3, two vertical lines respectively in Europe (ALL_FT+) and US (OpenFlights) represent when the number of airports included has reached the number of airport considered in the smaller data set (respectively OpenFlights for Europe and RITA for the US). If both data set had been collected the same way, then one would expect the value of the metrics at these vertical lines to be equal to the value for the smaller data sets when all available airports are being incorporated. Specifically, the topological metrics for the full OpenFlights European data set (resp. the full RITA US data set) should be equal to the value obtained in the ALL_FT+ European data set when 497 airports are included (resp. the OpenFlights US data set when 286 airports are included) identified by the vertical line. On the contrary, this does not happen, notwithstanding the final convergence to similar values. Given that the RITA data set only includes 16 airlines, one might ask whether an airline sampling might be the cause behind such behaviour.

Sampling airlines

As commented before, the sampling might also be done across other dimensions, and in particular along the airline dimension, whether it is done intentionally to focus the study on one airline or alliance (see for example [LC04, LSS14, RSNC09]); or is the result of data collection procedures (*e.g.* only 16 airports are required to report their movement to RITA - see Section

4.1.3). It is therefore of interest to look at the bias introduced by a carrier sampling.

Similarly to Fig. 4.3, Fig. 4.5 plots the evolution of the seven considered metrics as a function of the fraction of airlines included (largest first), with somewhat contrasting results. Specifically, it seems that a low(er) proportion of airlines is enough to approximate the complete topology. Such behaviour is probably caused by the fact that an airline encompasses both big and small airports, therefore better representing the heterogeneity of the system (in comparison with the usual larger-airports only sampling strategy).

Here again, the Chinese network stands out as an outlier, as few airlines do not suffice for the metrics to converge anymore. Notably, they do stabilise only when one third of the airlines are considered (except for the IC, which seems to converge much later), further reflecting the state-control route and region allocation of the Chinese market - that is, even major airlines might leave some regions of the network under-explored. Also, it can be extracted from Fig. 4.2 that the average degree for the Chinese network is 15, right in between of European (19) and US (10) values, revealing the good connectivity of Chinese airports, given a smaller number of airports, and further suggesting a point-to-point structure rather than a hub-and-spoke system [ZR09]. However, it must be noted that China possesses a comparable number of airports[11] (64) with more than 1 million passengers than US[12] (87) and the same proportion of them as Europe[13] (221 airports out of 609 for Europe and 64 out of 202 for China).

A set of airlines with some common characteristics (*e.g.* region of operation, alliance membership, etc.) might be chosen for the specific needs of a study, therefore rendering inefficient the number of operated flights sampling criterion. The same might be said if the airline sampling corresponds to the limitation of the data set, where only some predetermined airlines provides data (we can imagine that happening in the context of a private investigation project). There is thus a need to contemplate an alternative strategy for sampling airlines that includes more randomness - in other words, not only based on the number of flights operated by the airline. We propose the following two options to emulate such process:

[11] Civil Aviation Administration of China, 2014 [oC16]

[12] `http://www.faa.gov/airports/planning_capacity/passenger_allcargo_stats/passenger/` (Accessed May 2016).

[13] ACI EUROPE, personal communication.

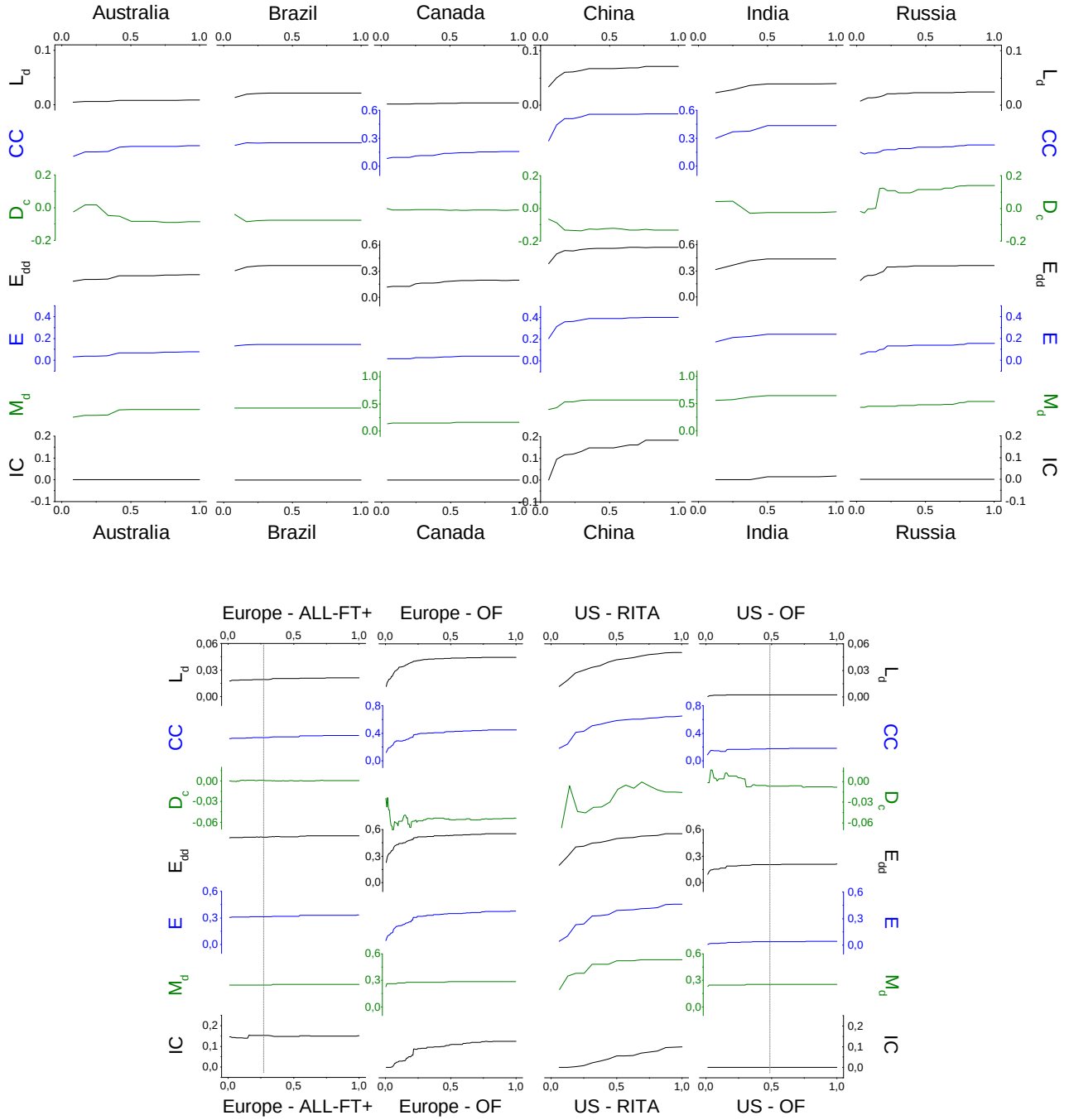

Figure 4.5: Evolution of network topological metrics, as a function of the fraction of airlines considered to reconstruct the network. Airlines are included in decreasing order of number of operations. Reprinted with permission from [BCPZ16].

1. A totally random process where airlines are drawn from the complete pool of available possibilities.

2. A substitution process based on the sequential original sampling. While, at first, airlines are selected sequentially in decreasing order of operations, a second step consists in discarding and randomly replacing some of them by other airlines. To illustrate this process

let us imagine a set with the busiest airlines in Europe (DLH, RYR, EZY, AFR and BER - which are the codes for Deutsche Lufthansa, Ryanair, EasyJet, Air France and Air Berlin). Then one of them is discarded (let us say, randomly, RYR) to be replaced by a, here again, randomly chosen smaller airline, for example SAS - Scandinavian Airlines.

The substitution process is based on the rationale that research studies, while not following the sequential sampling as for airports, seldom have completely random airlines and more generally work with a little set of big airlines.

Fig. 4.6 presents the results for both processes for the European Network. Four metrics (C, E_{dd}, E and IC) are displayed as a function of the number of airlines sampled. The blue line reports the evolution of the metrics through a random drawing process (option 1), while the red and grey circles and whiskers respectively represent the substitution process mean value and standard deviation. Grey circles indicates a 5:1 substitution ratio - that is that for every 5 airlines, one is substituted - while the red circles a 10:1 one. It can be seen that the random

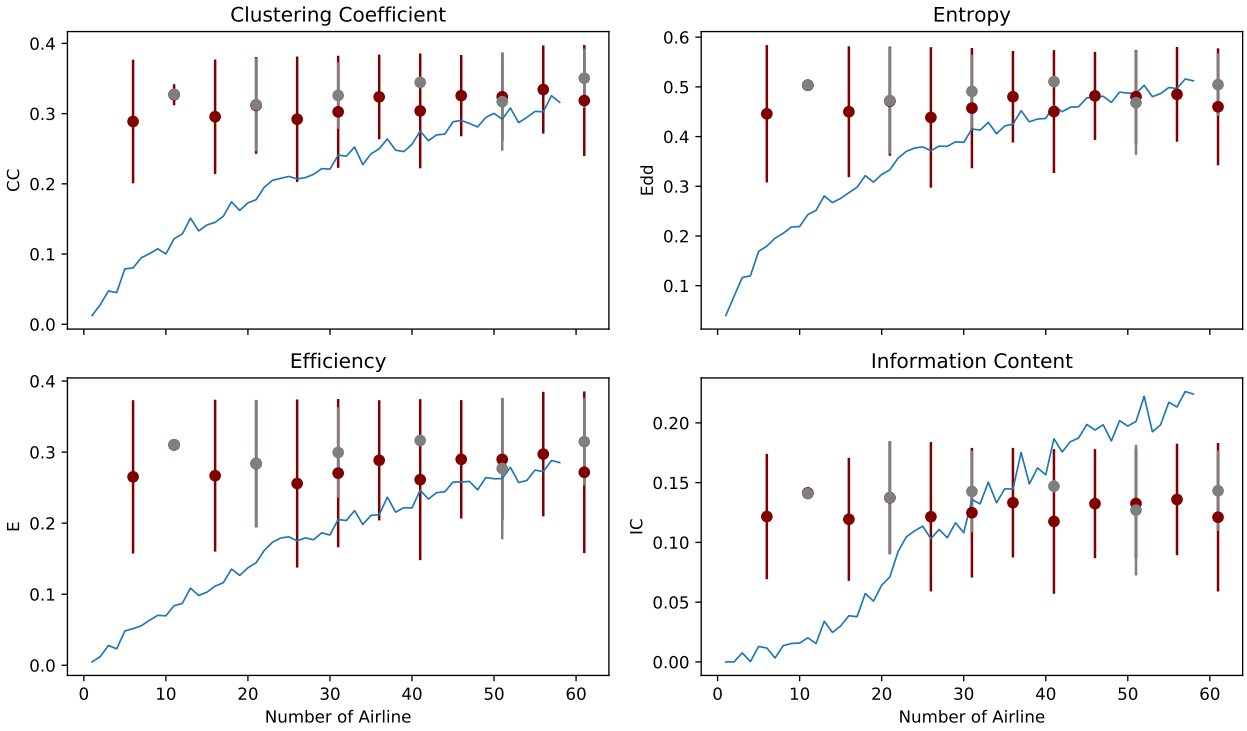

Figure 4.6: Topological metrics evolution as a function of a mixed sequential-random airline sampling process for the European (ALL-FT+) network – see main text for details. Blue lines indicate the evolution of metrics for a completely random airline sampling process. Reprinted with permission from [BCPZ16].

process converges lately, as there is a high probability of picking small airlines when the sampling is done randomly. On the contrary, the 5:1 substitution process approximates the final metric values with only one step, as it ensures that the core of network (created by the biggest airlines) is represented.

Sampling aircraft types and time windows

Air transport networks are seldom filtered or sampled according to aircraft types or specific time windows (at least, purposively), however, for the sake of completeness, this analysis will here be performed.

Excluding military aircrafts - which may bias the following results, but for which data was not available - we are here to study the effect that the different types of aircrafts have on the network. It can be appreciated in Fig. 4.7 that none of the four main topological metrics converges before all aircraft types are included (each aircraft type is introduced in decreasing

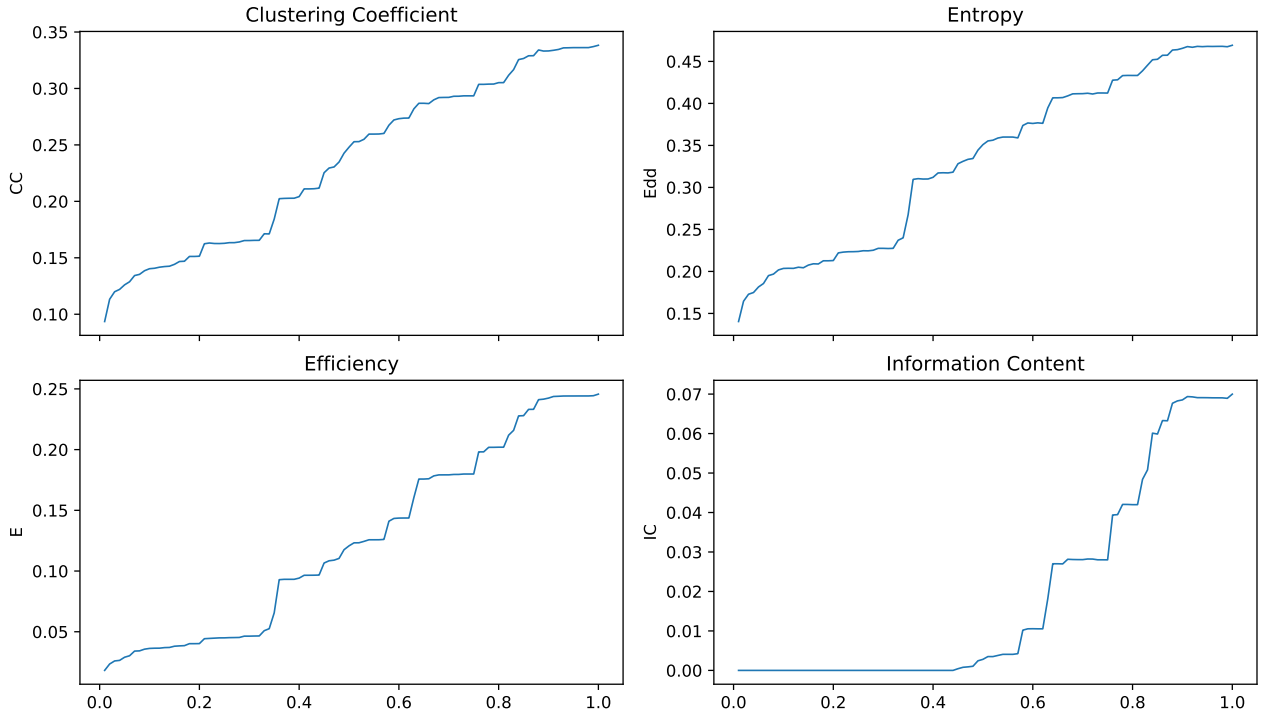

Figure 4.7: Evolution of topological metrics of the European (ALL-FT+) network, as a function of the fraction of aircraft types included. Aircraft are included in decreasing order of number of operations of their type. Reprinted with permission from [BCPZ16].

order of frequency). Specifically, note how the mesoscale structure of the system only appears above 0.6, that is when turboprops aircraft - smaller aircraft commonly connecting hubs with regional airports - are included.

Also, the time window considered can strongly affect the topology of the system, as traffic changes during the peak hours and night - the same can be said about summer and winter (more weather problems in winter) or across the different days of the week (less traffic during weekends). The focus on a given time window yield useful knowledge about the dynamics of the system under specific conditions, see [FRE13]. However, in order to synthesise the topological characteristic of the whole network representation to the system, Fig. 4.8 shows that large time intervals are necessary - to include weekly and seasonal effects. Indeed, for a number of days considered, a window have been dragged all over the dataset selecting every consecutive days combination possible in order to avoid eventual biases. It resulted in 200 calculations for each number of days considered (resumed in Fig. 4.8 by its mean value and standard deviation).

4.2.2 Delay performance as a function of sampling

Whilst the sampling of airports (and airlines on a much lower scale) have a non-negligible effect on the topology of the air transport representation, one might wonder if that transcribes also to other metrics of importance for ATM performance assessment. We decided to focus on delay measurement as it is one of the main points of this PhD Thesis. A vast Literature can be found about delay propagation. A relevant part of it have been presented in Section 2.4.3, characterising the dynamics and topologies of the delay propagation process. But, if the network's topology is bound to its representation, then how do airport or airline sampling strategies affect on the delays observed in the system, and subsequently on their propagation?

Sampling airports and airlines

Similarly to the previous section, Fig. 4.9 shows the evolution of the average arrival delay observed in the network as a function of the number of considered airports or airlines (which

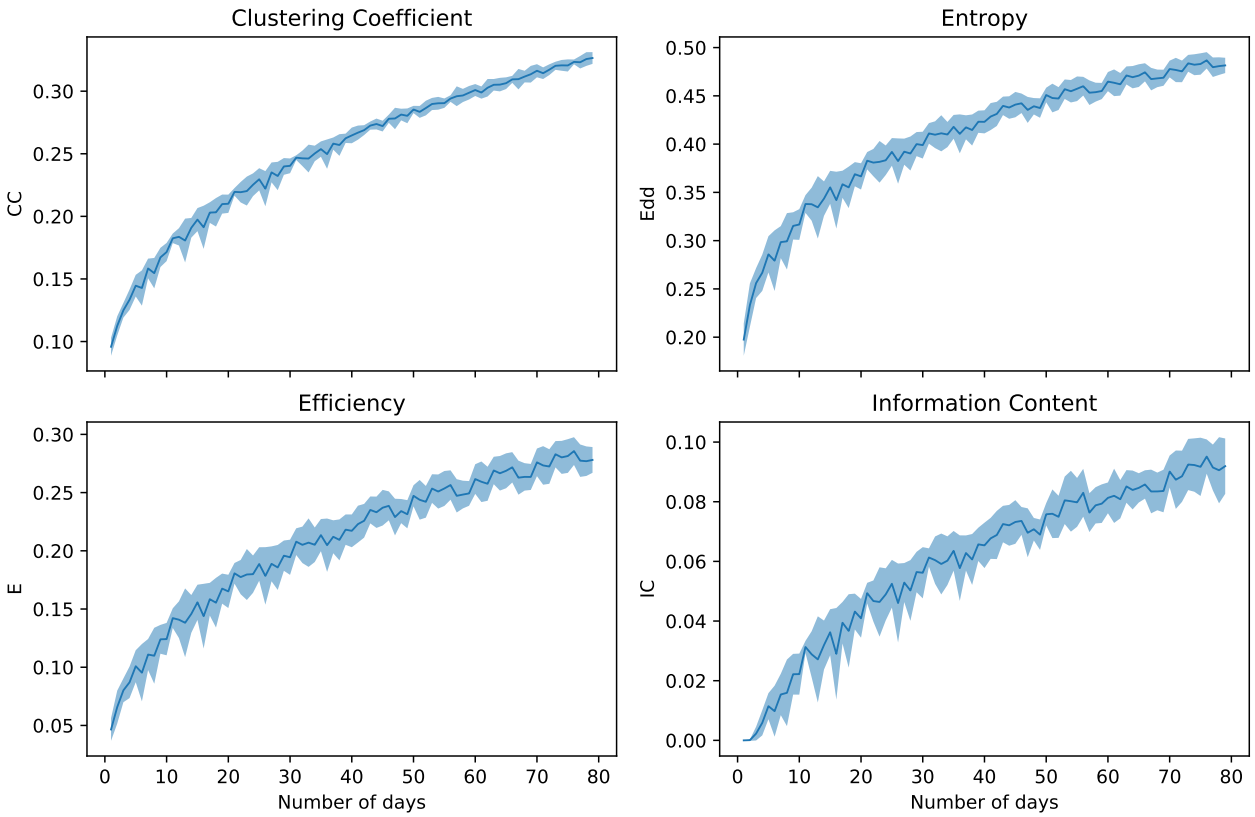

Figure 4.8: Evolution of topological metrics of the European (ALL-FT+) network, as a function of the number of days analysed. Each point is the average of 200 time-window calculations, and shaded area indicates the obtained standard deviation. Reprinted with permission from [BCPZ16].

are included in a decreasing order of their number of connections/operations). The black solid lines represent the average delay of the network calculated by discarding the negative values (*i.e.* flight that lands ahead of schedule), as common practice in the industry. The green solid lines include the negative values in the computation of the average delay. The dashed blue lines represent the proportion of flights that have been considered (right-hand axis).

It is remarkable that, for the airport sampling case, whether negative delays are included or not, the average delay follows a similar pattern, that is, a initial increasing phase reaching a local maxima when a few airports are included (less than 15% of the total number of flights), for then decreasing until convergence as more airports are included. Visibly, sampling too few airports leads to a non-negligible over-estimation of delay, as in the case of including the 10 biggest airports in the case of Europe. This information is of special relevance, knowing that EUROCONTROL reports on European performances are based on the top 30 airports [Com15]

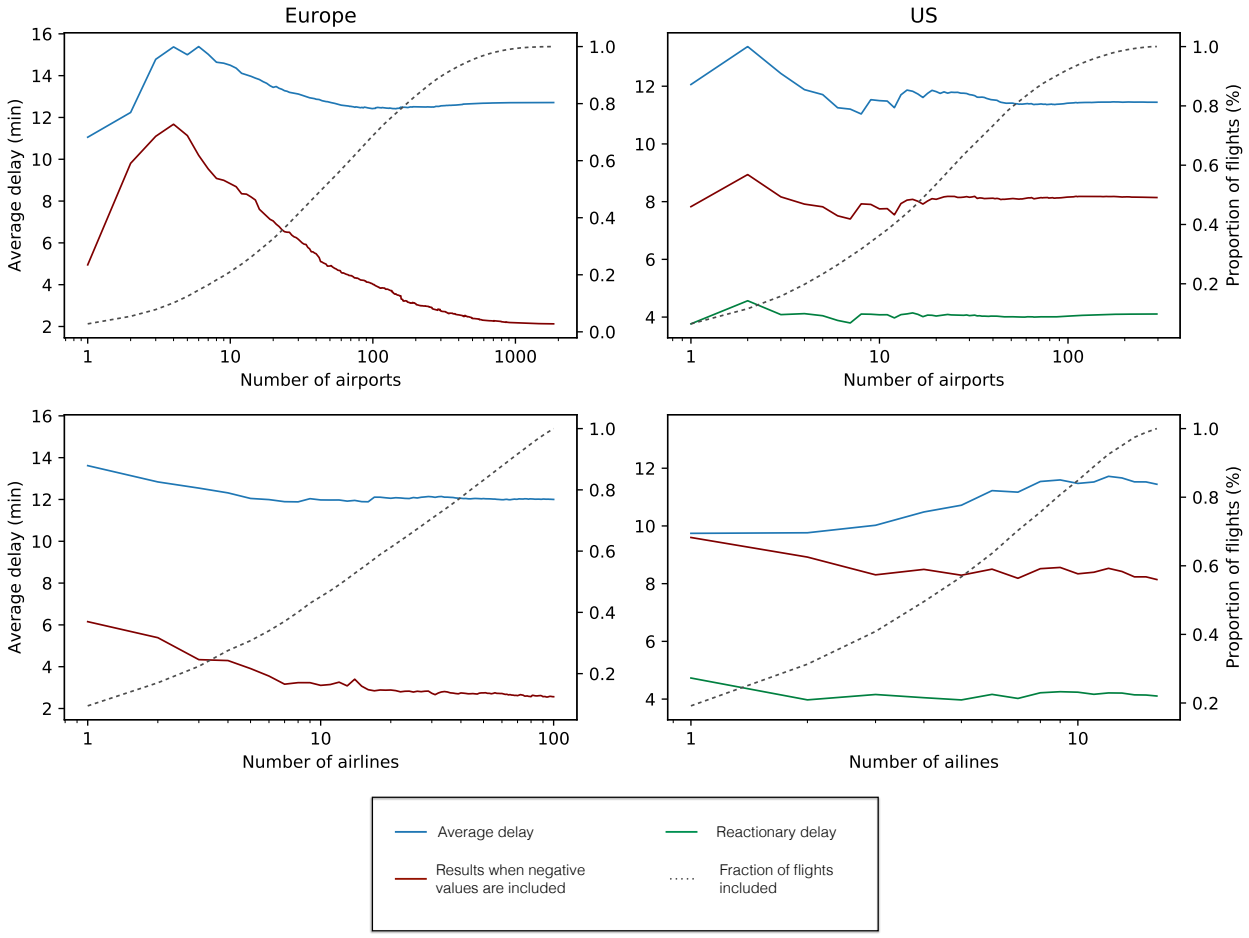

Figure 4.9: Evolution of average network delay as a function of the number of airports and the number of airlines. Reprinted with permission from [CBZ16].

or on the arbitrary number of 34 busiest airports in each region for reports comparing Europe and US performance [EUR14]. In Fig. 4.9, 34 airports corresponds to approximately 45% of the flights in Europe and 70% of US traffic; more importantly, it corresponds to a roughly estimated error of 2.2% for Europe, and 1.5% for the US. Whilst these errors suggest a quite robust approximation, one must be cautious when a new procedure aiming at lowering the delay of the system claims a change of the order of the percentage point. An analogous analysis has been done including sequentially airlines in a decreasing order of their number of operations (bottom Fig. 4.9, same colour coding) with contrasting results. First, Europe and US follows a similar behaviour when negative delays are included, as the net delays decrease as airlines are added - and a few of them are sufficient to approximates the final value. However, it seems that excluding negative delays leads to two distinct behaviours for the two regions, as European delay decreases with new airline entries while US delay increases. All values nevertheless seems

to converge to the previously obtained delay (*i.e.* obtained through the incorporation of all airports) suggesting that this opposite behaviour of Europe and US is not due to the low number of airlines in the RITA data set. On the contrary, it suggests that smaller carriers in US display more variability, as negative delays have more weight.

Considering the most complete network possible yields an average US arrival delay of 11.4 min, which can be compared with the 13.4 min delay reported by the FAA[14]. The difference between the two values is due to the reporting protocol in US (*i.e.* only delays superior to 15 min are reported) that excludes small delay values and therefore tends to over-estimate the average. The red dotted lines (Fig. 4.9) reports the evolution of the reactionary delay as a function of the inclusion of airports or airlines[15] in the US networks. The asymptotic value of 4.1 min represent 36% of the delay, which is in rather good agreement with the 41.9% provided by the US Bureau of Transportation Statistics (BTS)[16].

On the other hand, the European average arrival delay of 12.7 min displayed in Fig. 4.9 is fairly superior to the 10.1 min reported in EUROCONTROL Central Office for Delay Analysis (CODA) documents [dig15]. The European threshold for delay reporting is set to 5 min, thus giving a comparatively high resolution approximation of the delay, unlikely to explain the differences obtained. However, it must be noted that the data we used in Fig. 4.9 refer to last-filed flight plans, which may under-estimate the airline-reported delay relative to schedule. Indeed, we have mentioned the over-utilisation of the ground delay program (GDP) in Europe, which might partition a delay - caused by a single delay-generating event - into two parts. One part is assigned to regulation, and corresponds to the time waiting on ground; the other is the remaining effect of the delay-generating event. Our calculations only take into account the second part (yielded by the last filed flight plan), while the European average has been computed taking into account airline reports, which include regulation delays.

Before going ahead, an interesting effect can be observed from the evolution of the average delay

[14] `http://www.faa.gov/data_research/aviation_data_statistics/operational_metrics`, Accessed May 2016.

[15] Such information was only available for the US.

[16] US DoT, BTS: `http://www.rita.dot.gov/bts/help/aviation/html/understanding.html` (Accessed May 2016).

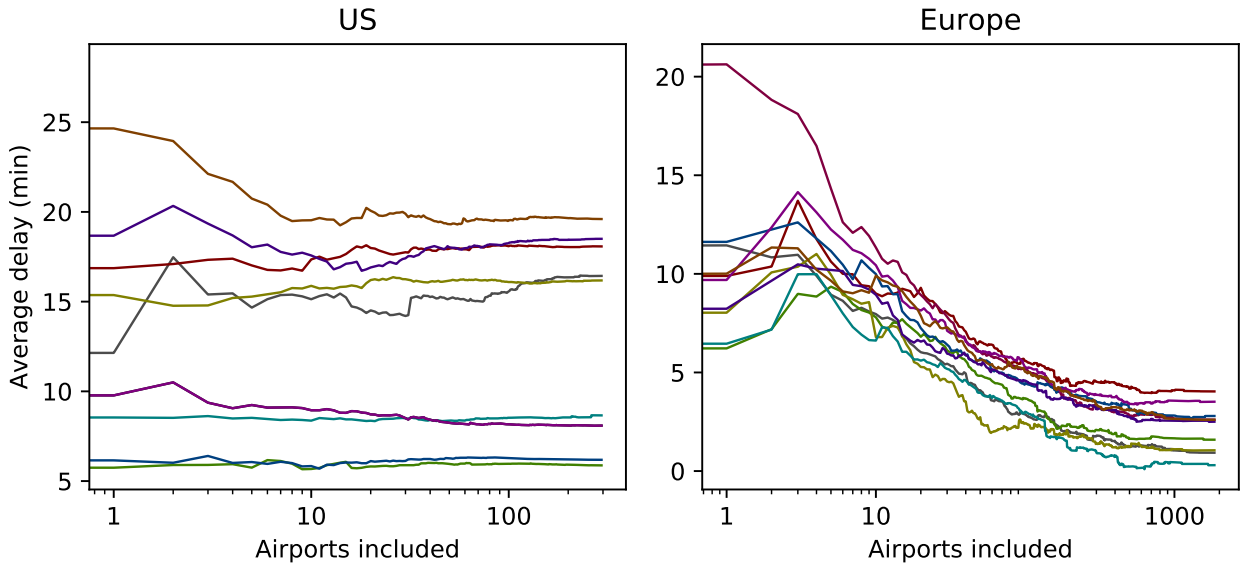

Figure 4.10: Evolution of average delay as a function of airports included for 10 randomly selected days. Reprinted with permission from [CBZ16].

as a function of the number of airports (with sequential sampling) for 10 randomly sampled days (Fig. 4.10). The figure yields insightful knowledge about the dynamics of both European and US systems. Whilst the behaviour described in Fig. 4.9 for Europe is always present - that is, a high peak when the first few bigger airports are included followed by a decrease as smaller and smaller airports as added - the pattern in the US is the result of the aggregation of different days, with different dynamics, rather than a systematic behaviour.

Multipliers analysis

Airport delay multipliers (DM) - the average airport departure delay divided by the average airport arrival delay - have been largely used in air transportation research [CTCZ15, HH13]. Although the limits of its usefulness will be discussed in Section 6.1, its high utilisation makes it insightful to study. This metric basically assesses the role an airport plays into (reactionary) delay propagation in a network, and past studies identifies hub as potential delay multipliers [SWL15]. Fig. 4.11 presents the evolution of the multipliers' distribution as a function of the number of airports considered. Airports are added sequentially according to their number of connections. For the sake of clarity, only four distributions per region are drawn. In Europe, delay multipliers' distributions for networks of 25, 50, 200 and 800 airports are considered.

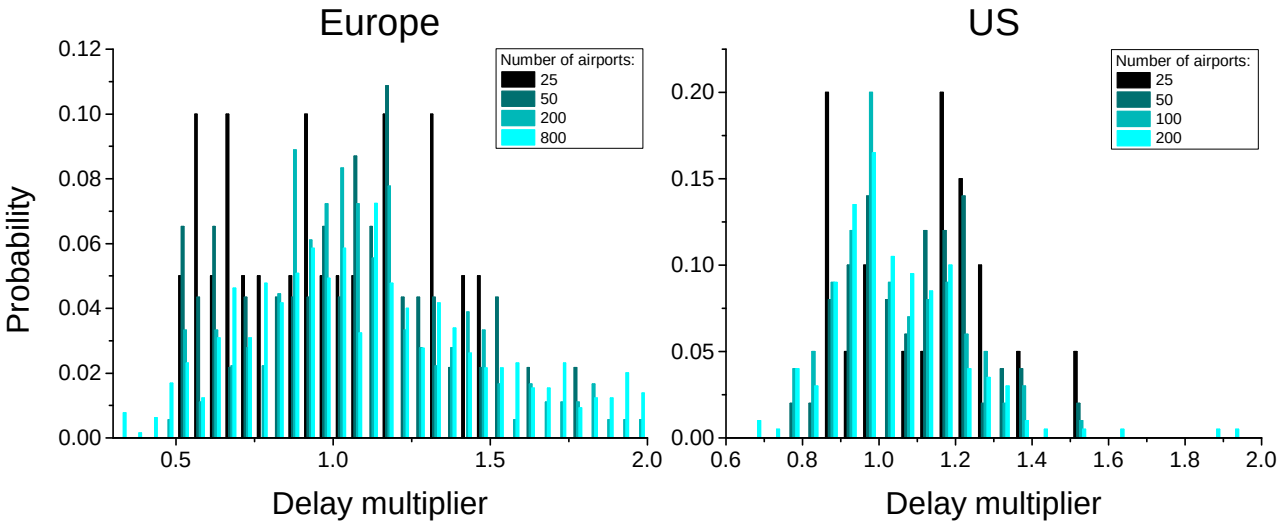

Figure 4.11: Delay multipliers as a function of airports included, for Europe (left panel) and US (right panel). Reprinted with permission from [CBZ16].

Because of the sequential sampling, the 25 airports considered in the 25 airport-network are also considered in the other ones. However, the additional airports considered might change the multiplier value of all the airports, as new flights are included in the calculation, thus the distribution. Actually, it can be seen that, as the fraction of airports considered in the European network increases, more extreme multipliers appear. As expected for previous results, smaller airports have important effects, confirming the reports [CTCZ15] stating that small airports double or triple arrival delays into reactionary delays - probably because of the lower recovery potential of such airports due to their less flexible turnarounds, sparser crews and fewer aircraft resources in comparison with bigger airports.

At this point, we have shown that the sampling strategy indeed affects the delay measurement, but we did not fully answered our preliminary question: whether the sampling affects both delay measurement and its propagation. Let us look now into the effect on the dynamics of the system.

4.2.3 Dynamic analysis

Soon enough the necessity of studying the dynamics of a network - an not only its structure and topology - become obvious in the perspective of improving the knowledge of a system. It

is rather intuitive to conceptualise that the nodes of a network represent elements with their own dynamics, which are linked between them according to a given topology. The structural perspective is therefore tantamount to a roadmap where all dynamics are supposed to take place, disregarding the characteristics of such dynamics [GB08, Zan15]. Let us illustrate this notion with an example. Consider a network representing epidemic spreading [New02]. People or groups of people are linked together in function of some features (*e.g.* friends, collegues, etc.); however, the propagation of the epidemic has its own internal dynamics (like latency time), which renders the final propagation much more complex to predict. In general, the bond between the dynamics of a system and its topology is far from trivial. The topology may have little effect on the dynamics development just as it can strongly condition it. A relevant example for the latter - and in relation with the previously discussed spreading network - is the limited power of vaccination (even in a large fraction of people) in stopping epidemics in a scale-free network [ML01].

This example transposes perfectly to the air transport case, as it can be of high interest to study how the structural-dynamical relationship of the network conditions, from a theoretical point of view, the delay propagation [FRE13] or the resilience of the system [ZL13a]. Whether or not the topological changes of air transport networks impact the dynamics of interest is to be proven. In this subsection, we are to assess to what extend the biased sampling process affects a dynamical process of our choice. For the sake of simplicity, we created a minimalistic model of delay propagation (explained further below) that emulates a random delay diffusion, inspired by random rumour spreading models [MNP04]. Whilst we are perfectly conscious of the model incapacity to reproduce the complexity behind the delay propagation patterns, its simplicity presents the main advantage of not masking the link of the results with the topology of the network behind a complex model - that is, that if the sampling process affects the output of this simplistic dynamics, the topological-dynamical relationships can be extrapolated to more complex models.

Model definition

The model basically assumes that delays are generated in a random fashion at an airport, and then propagated through all its connections. All sources of complexity for delay propagation are discarded (*e.g.* aircraft, crew, passenger connectivities, etc.). This simplistic model will be able to highlight the distortion introduced in the dynamics by a biased topological structure (which is itself due to a inadequate sampling strategy).

Let us denote the accumulated delay at an airport i as d_i, and assume that it is proportional to the delays of connected airports[17]:

$$d_i = \frac{1}{\lambda} \sum_j A_{i,j} d_j, \tag{4.1}$$

λ being a constant. This equation can be rewritten as:

$$A\mathbf{d} = \lambda \mathbf{d}, \tag{4.2}$$

A being the full adjacency matrix described in Section 2.3.2 and $\mathbf{d}$ the vector of centralities $(d_1, d_2, \ldots, d_n)$. $\mathbf{d}$ corresponds to the eigenvector associated with the largest eigenvalue of the adjacency matrix A, and is called 'eigenvector centrality' - [Bon07] used it for example to understand the importance of individuals in social networks. In this case, it represents the expected[18] equilibrium of delays when the propagation is random.

Results

The model has been implemented on top of the US network obtained through the RITA dataset. The left panel of Fig. 4.12 depicts the evolution of five elements of the eigenvector centrality $\mathbf{d}$:

[17] The model only drives the propagation of the delays after they appeared, and therefore disregard their generation. That explains why the delay at one airport only depends on external factors.

[18] Such conclusion is possible thanks to the Perron-Frobenius theorem, that ensures the unique existence of $\mathbf{d}$ for positive and non-negative matrices. This theorem also states that the elements of the largest eigenvalue - associated with the only non-negative, real eigenvector - can be interpreted as probabilities, thus our conclusion.

the elements corresponding to the 5 most connected airports in the US air transport network. Here again, airports are added sequentially according to their importance (*i.e.* number of connections) in a decreasing fashion. Note that at each step of the process (*i.e.* each time an airport is added into the network and **d** is recalculated), the values are normalised according to the higher value, so that one airport (the most important one in the sense of propagation) is denoted with the value 1.0. In Fig. 4.12, horizontal dashed lines represent the asymptotic value of the centrality for the five considered airports, that is the value obtained when all airport of the data set are included.

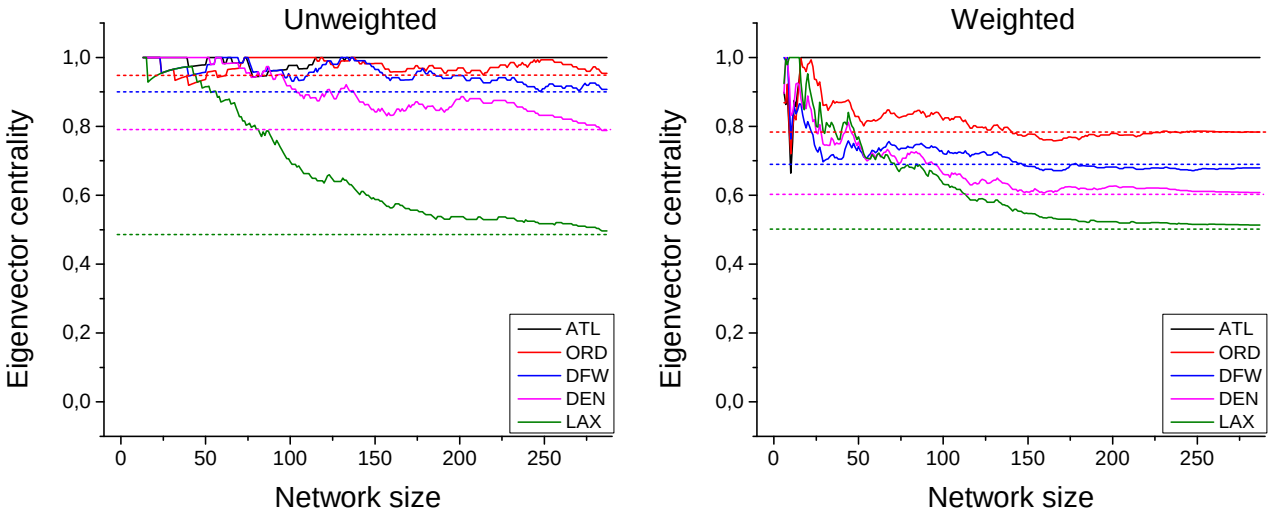

Figure 4.12: Evolution of eigenvector centrality of the top five USA RITA airports, as a function of the number of airports sampled in the network, disregarding (left panel) and including (right panel) link weights. Horizontal dashed lines correspond to the final node centrality of each airport. Reprinted with permission from [BCPZ16].

Centralities converge rather slowly to their final value[19]; however, even with a network of 200 airports, the centrality of Denver International Airport (DEN, pink line) is overestimated a 10% with respect to its final value (from 0.8 to 0.9). More startlingly, the volatility of the value is very high, as the asymptotic behaviour of the curves hardly tends towards the expected value. This bizarre behaviour can be corrected considering weighted links between airports, *i.e.* the strength of the links depending on the number of flights between two airports; as opposed to the binary scenario considered before, when even one flight was enough to link two airports in the adjacency matrix. The right panel of Fig. 4.12 shows a clearer convergence of the values

[19]We here mean the final value for the whole data set. The real centrality value would only be obtained if a dataset with all existing airports and their connection were available.

as a weighted network dwindle the sensitivity of the results to the number of considered nodes. However, it must be noted that 150 airports are still required to get a good approximation of the dynamics of the system.

Whether in weighted or unweighted cases, it is obvious that the repercussions of the sampling do not only extend to the topology of the system but also to its dynamics, as it may lead, in this simple example, to the inadequate identification of the most important nodes involved in delay propagation process if part of the network is disregarded. And linking this back to the effect of the sampling on delay measurement, we can postulate that the knowledge about its propagation dynamics is also conditioned by the sampling of the network.

4.2.4 Structure optimisation

Once it has been proved that sampling airports according to their degree introduces a strong error in the topology of the network - or in its dynamics for that matter - one may reckon the existence of an optimal sampling strategy aiming at minimising the topological bias.

The idea is basically to create a greedy algorithm that initiates with the complete US RITA network and iteratively discards the node/airport that have the least effect on the two topological metrics considered (in Fig. 4.13, the clustering coefficient C in black and the efficiency

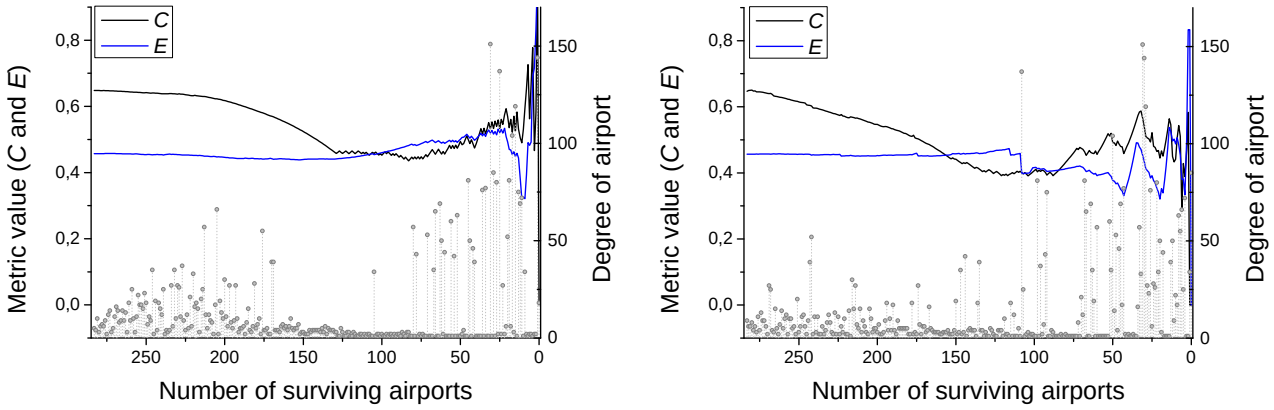

Figure 4.13: (Left panel) Evolution of C and E when the sampling process is guided by a greedy optimisation algorithm - see main text for details. (Right panel) *a posteriori* optimised method guided by E_{dd}. Graphs start from the complete network at left and end with a single node at right. Grey circles (right ordinate) indicate the degree of the airport deleted at each step. Reprinted with permission from [BCPZ16].

E) in blue). A greedy algorithm lacks of strategy because of its local optimisations, however it usually delivers a good solution with acceptable computational costs. In Fig. 4.13, grey dots represent the degree of the discarded node (right-hand axis). Whilst it is appreciable that the efficiency is kept quite stable until only 100 nodes are left into the network (75% reduction), the clustering coefficient happens to be much less stable, as it starts decreasing after a 30% reduction of the network (*i.e.* with 200 airport left within the network out of an original, complete 286-airports network). The insightful information is found looking into the degree of the discarded airports. It seems that the first nodes to be eliminated are neither the better connected ones, neither the less connected ones, but somewhere in between, confirming the importance of small airport into the topology of the system.

The idea now is to reverse those results to extract a satisfactory strategy. It seems that small and big airports are primordial to a better approximation, therefore, the solution may be to maintain the proportion of highly and sparsely connected airports, or in other words, to maintain constant the distribution of degrees or its entropy E_{dd}. Right panel of Fig. 4.13 displays the result of a greedy algorithm eliminating the node that less perturbs E_{dd}. This approach yields very similar results, in particular for the efficiency evolution (the case of the clustering coefficient is less optimal), suggesting that minimising the error associated with the entropy of the degree distribution is a good criterion whenever node sampling procedure has to be performed. Nevertheless, it must be noted that this procedure is only possible if the final value of E_{dd} is *a priori* known.

4.3 Results and discussion

Adopting an *information processing* point of view has many advantages when complex systems are analysed, the most important one being its data-centred approach, which means that no *a priori* knowledge is required anymore. However, the reliance on data significantly increases the importance of stable and consistent adapted metrics. Given the instability of the system's representations (varying number of airports, etc.), it becomes fundamental to assess the effect

of such representation biases on the metrics.

In this chapter, we have wondered whether the Air Transport topology is stable when its representation is changed for external reasons. This question arises from the data-dependance of the system's representation. The representation might be truncated or partial because of the limited coverage of the data. It may also happen on purpose, in order to study a specificity of the system (*e.g.* a specific region); or for the desire/necessity to reduce computation costs. A quick look at the literature offers a large panel of representations, spanning across different sampling processes and network sizes [LC04, WMWJ11, CWS+03, XH08, FRE13, ACNR07, Bag08], inevitably resulting in unreliable estimations of the topological properties of the system.

Several main conclusions are drawn from this study:

- More than 40 days of data need to be considered to estimate the general topological structure of the system.
- Similarly, a single aircraft type (or few of them) is not sufficient to effectively reconstructing the system's structure.
- Sampling airports based on their number of connections highly perturbs the metrics, as small airports play a fundamental role in the global structure of the system.
- A better strategy would consists in selecting a subset of the most important airlines and a smaller random one, as this leads to a recovery of the real average value [20].
- The sampling bias also perturbs the dynamics of the network.

It is of utmost importance to understand the message here conveyed. At no moment we are claiming that studies that focus on one specific day, a specific type of aircraft or a subset of nodes are erroneous or should not be done. On the contrary, it makes perfect sense to restrict the network to focus on the studied phenomenon. However, what should never be done is to extrapolate those results to the whole system. If the complete network is to be studied, then

[20] The recovery is on average because of the random substitution, by definition not deterministic and therefore potentially not efficient.

the sampling strategy must be chosen carefully for the proxy to be good enough. When global phenomena as delay propagation are studied, we encourage the research to ensure that his or her reconstructed network is a complete representation of the system, and that some dimensions (*e.g.* aircraft types, airport sizes, etc.) are not distorted, at the risk of extracting erroneous and highly volatile knowledge.

Beyond those specific conclusions, we trust that this chapter allowed a deeper reflexion about the general notion of *information processing* networks. Metrics, by definition, need to be intelligible, sensitive to a specific aspect of performance and consistent to allow comparatives. However, metrics are intrinsically dependent on the data set they are applied on. We have clearly demonstrated that even recognised and over-used metrics present non-negligible sensitivity to the data - that is, to the representation of the network we consider. As such, it is clear that progress in performance assessment or structural description goes side-by-side with data. A data-driven approach must be facilitated in order to gain further intelligence about the presented notion of 'sufficient' sample, which is far from intuitive in the case of air transport.

Chapter 5

Functional analysis of air transport dynamics

As previously introduced, a better understanding of air transport architectural interactions may come from the study of how the system processes information. The whole process of delay propagation can be seen as the *information processing* dynamics of the system. Specifically, an airport *receives*, *processes* and *retransmits* information about the system, *i.e* delays. Therefore, to complete the previous statical network analysis, we shift to a functional vision of the network. The information is not anymore represented by physical aircraft, but by the notion of delays: whenever a delay is measured at a receiving airport, it implicitly convey information about the departure airport and the crossed airspaces.

The real advantage of functional networks over classical networks for the understanding of delay propagation process is that they are reconstructed with real operational data. Beyond the fact that both approaches yield a global view of the process, functional networks have no need of an *a priori* information or of limited propagation models (see Section 4.2.3) to represent the process. More importantly, and as opposed to the simplistic model-based networks reconstructed in the previous chapter, functional networks allow for the detection of indirect propagations, *i.e.* links between pairs of airports that have no direct connection between them. This is specifically due to the use of causality metrics, endorsing the extraction of delay-related information within the

network.

In this chapter we shed light on the functional properties of the system; that is, we aim at complementing the extracted information about the structure of the system with knowledge about the delay dynamics taking place on top of it. This process will result in an abstract representation of the system depicting the propagation process, upon which all the power of the complex network framework can be applied to extract intelligence about its dynamics.

A special attention will be devoted to the study of abnormal situations. An example is useful to visualise such idea. Consider a stroke, for instance caused by an ictus, that causes regions in the brain to deactivate, while others take over to recover a normal capacity. This is similar to what can happen in air transport, when an airport is closed over a long period of time; for instance, consider the closure of Brussels airport due to the 22 March 2016 terrorist attack, and how flights had to be rescheduled to nearby airports. This will not just be an academic exercise: studying the dynamics of delay propagation can suggest new strategies for optimising the response of the air transport system.

5.1 Linear phase changes in delay propagation networks

Functional networks (see Section 2.3.3) are very much adapted to the study of propagation of delays. The functional definition of delay propagation network can be expressed as follow: 'Nodes are connected whenever a propagation process of an inmaterial entity (delay) is identified. The identification process entails causation detection between two time series, respectively related to each pair of nodes, thereupon ensuring that a connection implies that the delay dynamics of an airport partly drives the dynamics of the airport on the other side of the link.'

The most used causality metric is the celebrated Granger Causality (GC) test (see Section 2.2.1). GC would allow the detection of any causation - assuming it is of a linear nature - between the delay dynamics of a pair of airports. However, GC under-evaluates abnormal behaviours, an inherent shortcoming that has not been discussed in Section 2.2.1. Indeed, for

all available data to be processed, only an 'average' pattern can be encountered. The Granger test tries to improve the prediction of an entire time series based on linear inputs from another time series. Any intermittent causal relationship between both time series would be smoothed away by the prevailing non-causal dynamic. This shortcoming becomes relevant when one starts to consider that the topology of delay propagation may have a dynamical nature - a hypothesis emanating from the idea that airports are not expected to handle delays under high pressure (*e.g.* high traffic, bad weather, etc.) the same way they do in normal 'average' situations. With GC, such changes in the dynamics might pass unnoticed, as the abnormal delay propagation channel between a pair of airports - active only under specific extreme conditions - is statistically insignificant next to the average predominant (in the data) independent dynamic of each one of them separately.

To compensate for this particular disadvantage of the Granger-based functional networks, have been introduced the Extreme Events (EE) causality metric (see Section 2.2.2), which has been specifically designed to spot causality relationships based on abnormal and extreme behaviours; therefore bypassing the shortcomings of the Granger Causality and allowing the construction of a complementary functional network representing the underlying extreme propagation dynamics. Both metrics will thus offer insights about the standard expected delay propagation process and the disrupted one triggered when abnormal delays are suffered within airports.

5.1.1 Data preparation

We executed the analysis over two data sets: the first one focuses on the European Air Traffic network, the second one on the Chinese network. Delays have been calculated as the difference between the real and scheduled arrival time of a flight as reported in the Chinese and European cases respectively by the ADCC's data set and ALL_FT+ data set (see Section 3.2). Subsequent data represent thus the real delay of a flight - as perceived by passengers. Such measures are more precise than official announced ones, as China for example does not report delay if the flight arrived within 10/15 minutes from the expected arrival time. In Europe, delays are officially reported when their magnitude overpass 5 minutes. Yet, the data extracted

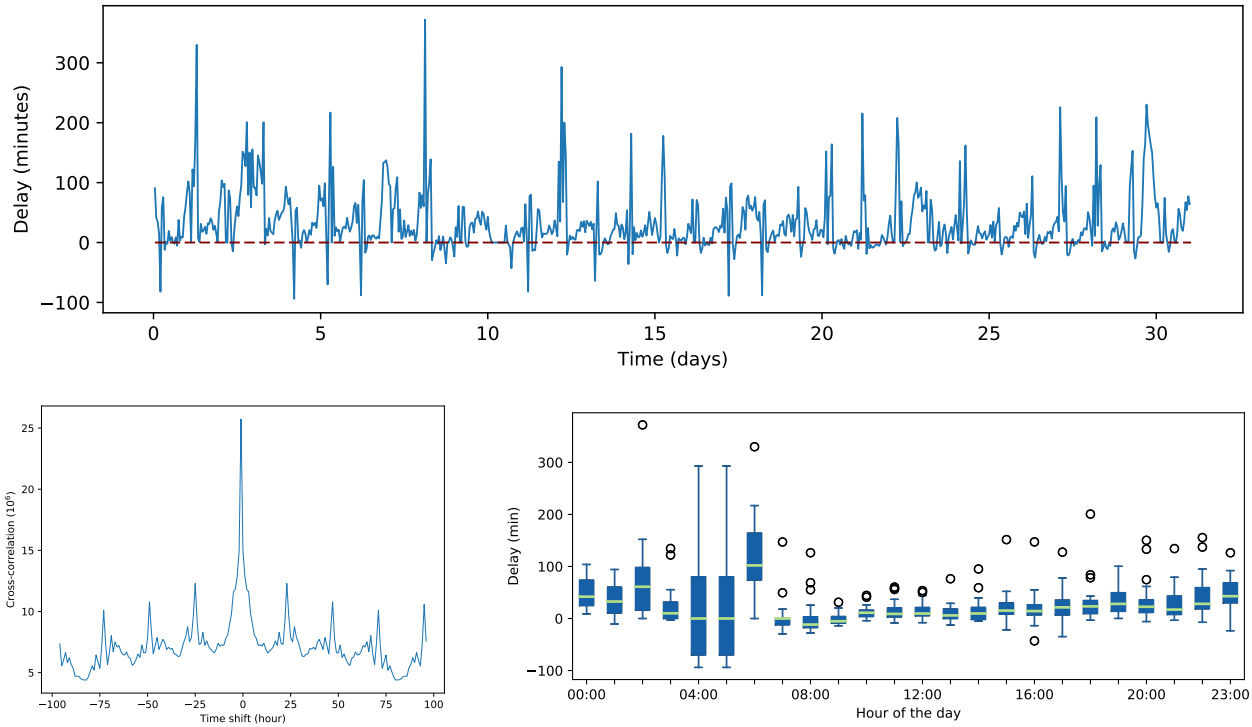

Figure 5.1: (Top panel) Average hourly landing delay at ZBAA - Beijing Capital International Airport. August 2015. (Bottom Left) Cross-correlation of the delays time series. (Bottom Right) Box plot of the distribution of landing delays as a function of the hour of the day. Reprinted with permission from [ZBY16].

represent a more granular and complete measurement of delays - for instance, yielding both positive and negative values, quantitatively assessing if the flight arrived respectively after or before schedule.

For each airport, a time series has been extracted from the original data set, corresponding to the average landing delay within one hour time window. This resulted in a matrix D of size $(100, 3888)$ for Europe[1] and $(152, 744)$ for China[2], whose element d_{ij} encodes the average arrival delay for airport i at time j, where j is counted as the number of hours passed since the first flight of the data set. Each column, corresponding to the time series attached to an airport (*i.e.* to a node of the network), is highly trended, as delays almost disappear at night and tend to be higher during peak hours.

[1]A subset constituted of the 100 busiest airports (out of the 1854 ones present in the ALL_FT+ dataset) have been considered.

[2]For China, the matrix D is of size $(152, 744)$ as the data set only contained information for the month of August, for 152 airports.

An illustration of the trends of the resulting time series is provided for the airport of Beijing in Fig. 5.1 (a), where the 24 hours (daily) periodicity patterns are confirmed by the cross-correlation plot of panel (b). This periodicity is caused by the high dependency of delays to the hour of the day, as they tend to be higher on the afternoon than in the morning (panel c). This non-stationarity of the data infringes the most important requisite of Granger and Extreme Event causality tests: the stationarity of time series. Consequently, each column i of the matrix D (the Chinese and European one) has been pre-processed to eliminate trends as follows:

$$d_{ij} = \frac{d_{ij} - \bar{d_{il}}}{\sigma^2(d_{il})}, \quad l = j \pm 24k, k \in \mathbb{N}, \tag{5.1}$$

where a subset of time stamps $(l)_k$, corresponding to the same hour of the day than the modified time stamp (or row) j, is considered. For all times stamps (corresponding to the rows of D), $\bar{d_{il}}$ and $\sigma^2(d_{il})$ are computed, as the average and standard deviation of the delay expected at that specific hour. In other words, each value of the matrix D representing the original average arrival delay of an airport at a given hour is replaced by its z-score - that is, how far away from the expected value the delay was. High positive z-scores correspond to delays larger than that expected on average at the same hour and at the same airport. A negative z-score, on the other hand, expresses that a lower delay (with respect to the average delay at the same hour and at that airport) has been suffered. Ergo the pre-processed matrix D is independent of the original magnitude of the delay. A graphical representation of this process has been provided for Beijing airport in Fig. 5.2.

5.1.2 Delay causality and phase changes

Delay Causality network

By applying both causality tests (namely, Granger and Extreme Events causalities) to each pair of airports and their corresponding afore-described time series, we are able to construct

a functional network, in which each active link assesses the presence of a causal relationship between a pair of airports' delay dynamics. A link is considered active when the p-value returned by the test is statistically significant (p-value < 0.01). Also, confounding effects (*i.e.* mistaking correlation with causality) are discarded from the Extreme Event network by eliminating double directional links[3]. In other words, whenever both causalities $A \rightarrow B$ and $B \rightarrow A$ pass the significance filter, the relationship between airports A and B is not considered as a causality, but as a mere correlation [Zan16]. The green and red networks presented in

[3]Granger and Extreme Event Causality tests remain sensitive to confounding effects introduced by third-party element. See Section 7.2 for more details.

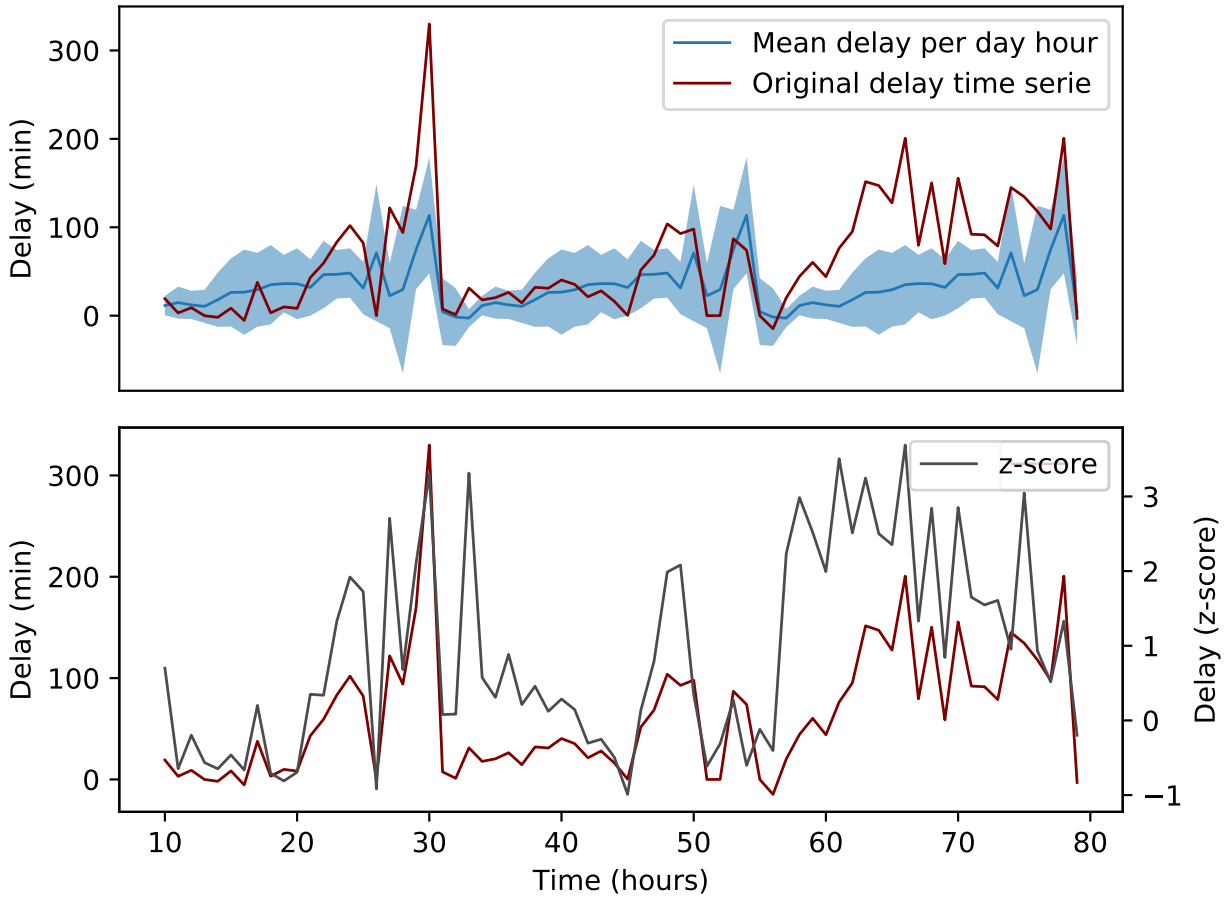

Figure 5.2: Example of the detrending process of the ZBAA - Beijing Capital International Airport time series. (Top panel) The red solid line represents the average delay corresponding to a row D_i of the Chinese matrix D. The blue line and blue shaded area respectively depict the average delay at that hour of the day and its standard deviation. Notice blue line's periodicity. (Bottom panel) The original delay time series (red line) is plotted along with the detrended time series (grey line, right scale). Reprinted with permission from [ZBY16].

Fig. 5.3 correspond to the networks yielded respectively by the Granger and Extreme Event metrics.

A visual inspection is not much useful, except at sensing that the structures of the two networks is not the same - confirming our initial intuition of different propagation dynamics under different delay conditions - and a higher link density for the EE network. To have a cleaner image, Fig. 5.4 represents the sub-networks composed of the 10 busiest airports. Nevertheless, more information can be extracted by looking at some relevant topological metrics for both network. The results are reported in Tab. 5.1. Note that some metrics are complemented with their z-scores between parenthesis (because of their dependence to the number of links of the network, see Section 2.3.2), which correspond to the metric deviation from its expected value.

Tab. 5.1 bears several information: if the Granger Causality (GC) network has fewer connections (lower link density) assessing propagation delays than the Extreme Event (EE) network, it notwithstanding propagates with higher efficiency than the EE network[4]. Also, the lower

[4]Note that it is the z-score value of the Efficiency metric that allows for that conclusion. The actual efficiency value of the GC network is lower than the EE network's one. However, this metric dependence on the link density makes them incomparable.

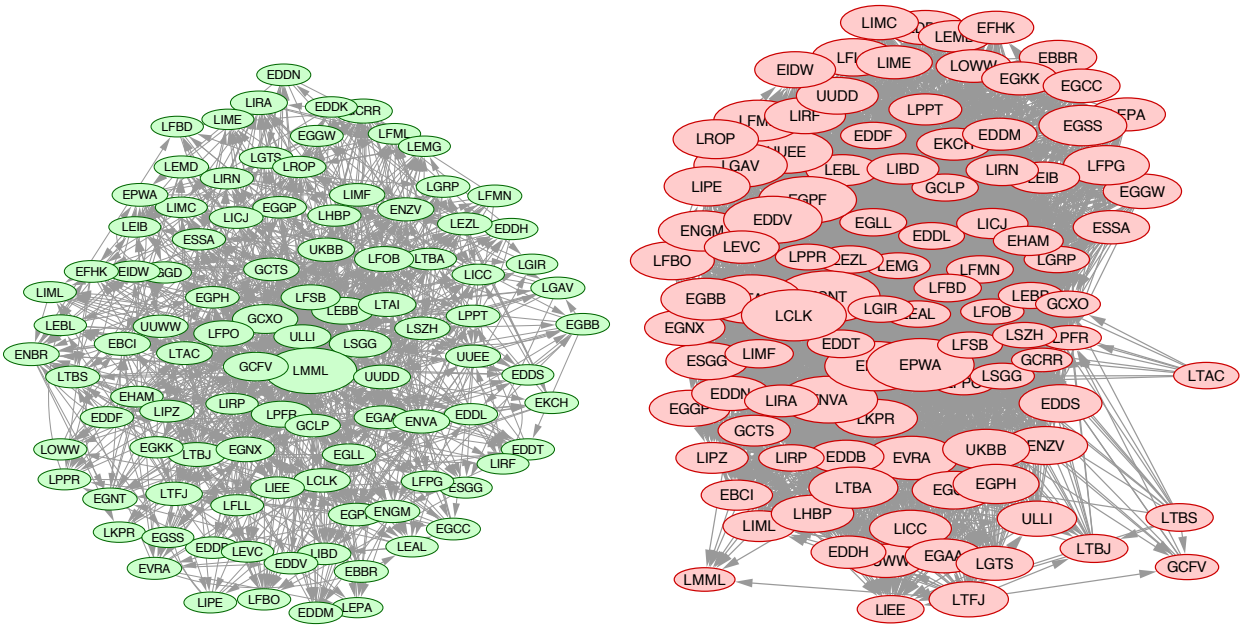

Figure 5.3: Functional networks obtained through Granger (Left) and Extreme Events (Right) causality metrics. Node sizes are proportional to the number of causality connections generated at each airport. Reprinted with permission from [BZ16].

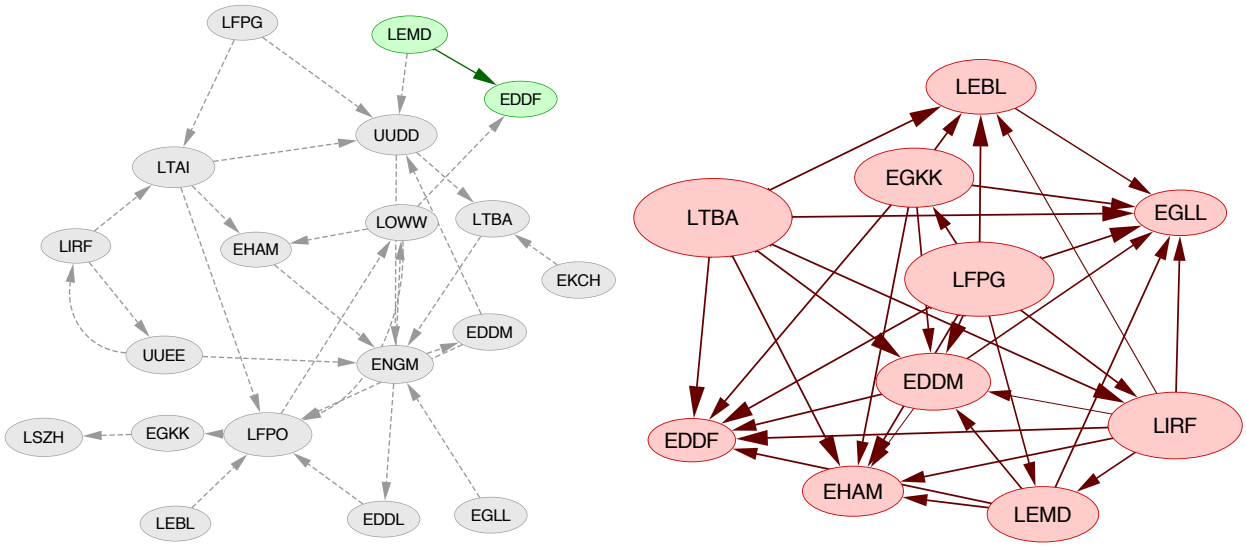

Figure 5.4: Networks of the top 10 airports in number of operations, for Granger (Left) and Extreme Events (Right) causality metrics. For Granger, only two top-10 airports are connected (LEMD and EDDF, in green); for the sake of clarity, airports of the top 20 have been added, in light grey. Reprinted with permission from [BZ16].

Metric	GC Network	E.E. Network
Link Density	0.1820	0.6479
Transitivity	0.2703 (-0.2227)	0.7382 (-0.2421)
Efficiency	0.5910 (1.7912)	0.8238 (0.0423)
Assortativity	-0.2227	-0.2421
Diameter	3	2
Information Content	0.9353	0.2683

Table 5.1: Resume of topological results. Reprinted with permission from [BZ16].

efficiency of the EE network contrasts with its non-trivial structure, as suggested by the low IC. Such combination (low IC, high Efficiency) indicates that delays are struggling to propagates in a system whose structure is not regular, suggesting a localised propagation (*i.e.* within groups of nodes).

A deeper analysis of the networks shows that the propagation capacity of an airport is inversely linked to its number of recorded operations (the best fit being an exponential decay, R-Square of 0.066, see Fig. 5.5 left and centre panels). Bigger airports cause less propagation than smaller airports. In fact, this characteristic (valid for GC and EE networks) is bound to the amount of buffer available for each type of airport (*e.g.* the number of stand-by aircraft or the number of runways that compensate for wind direction changes). Similar observations have been made in

the literature for the propagation of normal delays [Jet09, CTZ13, ACGB08]. Smaller airports thus have a higher propensity at propagating their delay. This is even more evident under extreme conditions (EE network), where the number of outbound links for smaller airports is significantly higher than under normal conditions. Another possible explanation for such result resides in the fact that in smaller airports, having less traffic, any strong delay creates a important increase of the z-score, thus leading EE to interpret it as an extreme event - and GC inversely smooths them down with respect to the average behaviour.

Are these properties related? In other words, are airports propagating under normal conditions more prone to propagate under extreme ones? Fig. 5.5 (right panel) shows the number of outbound connections for the GC network as a function of those of the EE network, therefore comparing the airport behaviours during normal and disrupted phases. The absence of a relationship is quite remarkable. It shows how airports might not propagate their delay under normal circumstances and suddenly become a central node of the propagation process under abnormal events. Surprisingly (and less intuitively), the opposite may also happen. The Malta International Airport (LMML) perfectly illustrates the second case, as its strong importance within the propagation of delay in the normal phase vanishes during a disrupted phase. Such a puzzling GC-centrality has a two-folded explanation: (a) being the smallest airport makes it more sensitive to statistical fluctuations; (b) its touristic appeal makes it over-connected. In fact, more than 35 airlines ensure the connection with LMML. If we compare this number with the connections of the 99th airport (namely, Beauvais-Tillé - LFOB), with just 4 operating airlines, this may explain the all-over-Europe propagation of standard delays. On the other

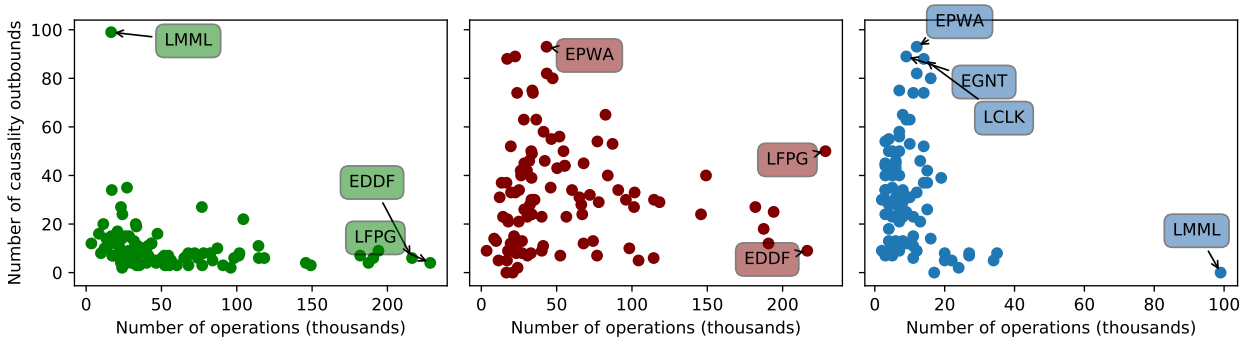

Figure 5.5: Relations between the number of causality links generated by each airport, and its volume of traffic. See main text for details. Reprinted with permission from [BZ16].

hand, the low number of outliers or extreme delay values for LMML, as illustrated by the high kurtosis of its z-score time series, may be partly behind the low centrality of LMML in the EE network, as the algorithm struggles to define properly extreme events. It is important to note that the airports' EE-centrality (*i.e.* the number of outbound nodes) is independent from the time series kurtosis, in the sense that several airports with similar kurtosis actually have more EE outbound links. The absence of an extreme event propagation of LMML is then not only linked to the difficulty to define outliers, but also to its internal dynamics (large delays are handled successfully or smoothed away with respect to receiving airports mean delays).

Validation

The attentive reader might find a contradiction between our approach and Chapter 4 conclusions. Indeed, we have previously said that considering a small subset of airports, selected sequentially according to their traffic, leads to unstable topological results; and, in the presented work, we have precisely selected the 100 busiest European airports to compare the topologies of two functional networks, representing the delay propagation process that takes place on top of the network. However, there is a fundamental distinction with respect to Chapter 4, which makes our approach valid. Firstly, the average delay value attached to each airport in Chapter 4 have been calculated considering only the aircraft flying between the subsampled airports; thus transmitting the subsequent bias into the time series. Here, on the contrary, the calculations are performed taking into account the complete data set. Only after that, a subset of airport is selected. Thereupon, time series contain the complete information of the system dynamics, allowing for a sound causation detection.

Yet the system is still truncated. Only a portion of the complete system is considered - and as such, the extracted metrics value and conclusions are not valid for the entire network. Nevertheless, the comparison of two interpretations of the same system (*i.e.* normal *vs.* disturbed delay propagation phase) is well defined - although its conclusions only hold for the sample of the network that has been considered.

Another fact needs to be clarified: whether or not the explanation provided for the Malta

International Airport (LMML) importance in the delay propagation network is satisfying; or, on the contrary, if it is more likely to be a statistical error within the execution of the Granger causality metric. To validate such result, random noise is inserted into the LMML delay time series before the causality tests is performed. If real information is contained within the original LMML time series, therefore, the causalities must vanish with the increasing magnitude of additional noise. Otherwise, it would mean that the original detected causality was the mere result of statistical fluctuations. We have iteratively introduced a noise drawn from a centred gaussian distribution with an increasing variance - taking into account that, by construction, the variance of the original data is 1.0 for all airports. The addition of a noise with 10% the variance of the original data set (variance of 0.1, figure not shown) is enough to drop the significance of all outbound links from Malta, thus discarding the GC statistical error and reinforcing the idea that causal information is contained within LMML delay time series.

Phase changes

The GC and EE networks have been shown to be quantitatively and qualitatively different - describing two distinct phases of the network: the average flow of delays and the disrupted flow triggered by abnormal events. It is therefore interesting to investigate when the transition between the two phases happens. In reality, though, what happens is nothing like that. The entire network does not start propagating delays in a disrupted way, but only a portion of the airports (those who suffered abnormal delays) are supposedly propagating according to the EE topology, the rest remaining on top of the GC topology. Real conditions would then be a mix of both networks, not exclusively one of them. However, while such study is still to be done (see Section 7.2), studying the transition as a global phenomenon will notwithstanding yield information about each individual airport's transition of phase.

Transition effects are quantifiable according to the design and definition of the extreme event causality test. One of the steps of the EE test consists in tuning a pair of thresholds - unique for each pair of airports - fencing off extreme delays from normal ones. The analysis of these thresholds yields a good proxy of the disturbance's magnitude triggering a phase change.

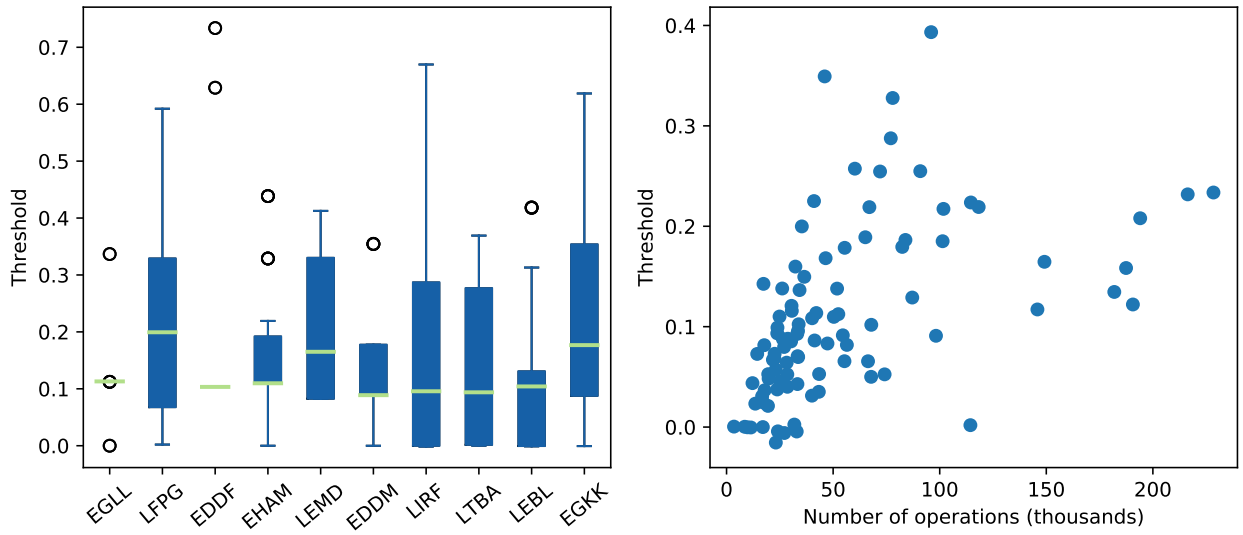

Figure 5.6: (Right panel) Box plot of the thresholds for the phase change, for the 10 airports with most operations. (Left panel) Average airport threshold as a function of its traffic volume. Reprinted with permission from [BZ16].

As we said, a threshold is defined for an airports and for each of its connexions - that is, that for an airport with 20 outbound extreme event propagation links, there will be 20 distinct independent thresholds. Fig. 5.6 (left) presents a box plot with the distribution of these thresholds for the top-10 airports (in terms of traffic). The horizontal line of each box represents the average surprise necessary for each airport to enter the disrupted phase. The threshold is measured on the processed time series of z-scores, which represents, at each hour, how much above or below expectancy is the average arrival delay. Accordingly, the average threshold, plotted in Fig. 5.6, indicates the surprise (encoded in term of z-score) necessary to switch phase. In other words, it specifies the value above which the time series are considered extreme. In the right panel of Fig. 5.6, we can observe that this average threshold is positively correlated (linear fit, R-square of 0.011) with the size of the airport, *i.e.* with the number of operations. In other words, the busiest the airport, the higher is the surprise needed to trigger a change phase. Moreover, the traffic of the airport is highly correlated with its size, and available resources, which perfectly explains why hubs (in Europe) are more resilient to disruptive perturbations, whilst small airports react only to smaller fluctuations.

So far, we have merely been looking at the thresholds values, which suggest that small airports differ from the busiest ones by being more sensitive to delay fluctuations. Yet, no information

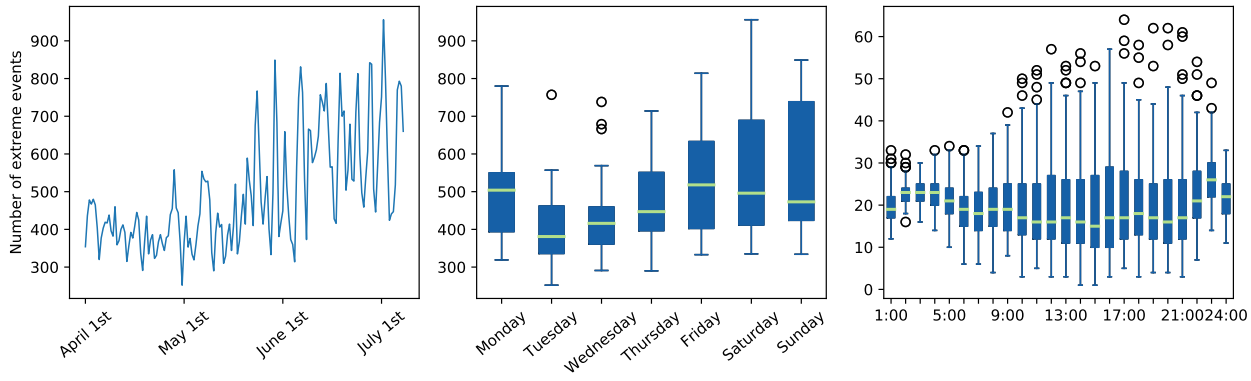

Figure 5.7: Evolution of the number of extreme events through time (Left), and as a function of the day of the week (Centre) and of the hour of the day(Right). Reprinted with permission from [BZ16].

about the number of times such threshold is overpassed is given, *i.e.* the number of times a link of the EE network propagation process is being activated. Note that each airport might have several thresholds, as specified before, implying that an airport A might be in the disrupted phase with respect to airport B but behaving in a normal way with airport C. Fig. 5.7 (left) presents the evolution of the total number of links activation as a function of time, showing a fickle/bursting behaviour that tends to be even more frequent during summer time - possibly because of the higher traffic levels, in turn causing more delay problems. causing more delay problems.

Fig. 5.7 centre and right box plots presents the distributions of the number of EE-link activations as a function of the day of the week and of the hour of the day, with neither of them showing significant perturbation of the average number of link activations. This was of course expectable for the latter, considering the way our time series have been made stationary. However, the number of link activations suffers from a significantly higher variability during peak hours, further confirming the critical influence of traffic volume on triggering phase changes. Note that the normalisation only stabilised the average number of EE-link activations per hour, considering that it has been done over the records of the same hour, therefore averaging summer and winter observations and resulting in the visible trends in the left panel.

We have stated earlier that thresholds precise the amount of surprise necessary to trigger the appearance of EE-links into the delay propagation patterns. In term of terminology, this amount

was called 'surprise', for it represents how far the triggering delay is from expected average suffered delay (z-score). Such notion is of low interpretability, or at least, is poorly intuitive. To solve this problem, we want to transform it into an approximation of the actual critical accumulated delay (within one hour) that drives the airport into the disrupted propagation phase. The relation between the critical delay and the threshold between two airports i and j is (reversing the z-score function, see Eq. 2.11):

$$d_{il}^{c} = \sigma^2(d_{ij})\tau_{il} + \bar{d_{ij}} \tag{5.2}$$

d_{il}^{c} being the critical delay accumulated within one hour in airport i that would trigger a propagation from airport i to airport l. τ_{il} is the threshold between airports i and l, that is, the surprise in terms of the z-score required to switch phase. Note that the hour j is still present in the equation. However, to avoid the external-airport l-dependency, we want to approximate the average of the critical accumulated delay that triggers the changes - that would be the intuitive time counterpart of the average threshold of an airport. Thus:

$$d_{i}^{c} = \left\langle \sigma^2(d_{ij})\tau_{il} \right\rangle_{l} + \bar{d_{ij}}. \tag{5.3}$$

As the average is made on the variable l (*i.e.* the airports) and not on the hour j, it can be rewritten:

$$d_{i}^{c} = \sigma^2(d_{ij})\tau_{i} + \bar{d_{ij}}, \tag{5.4}$$

with τ_i being the average threshold for airport i. In this last equation, d_i^c is the average critical accumulated delay (within one hour) leading to phase transition for airport i. As it has been discussed, this value is a rough approximation. Nevertheless, it remains an interesting indicator of the stability of airports against severe disturbances, by providing a time proxy of the cumulated delay triggering abnormal propagation behaviours. Fig. 5.8 (left) represents the magnitude of this average critical delay for the 10 busiest airports in Europe, highlighting the

average additional surprise delay (red bars) necessary to reach the average critical transitional delay, and by segregating it from the average expected delay (blue bars).

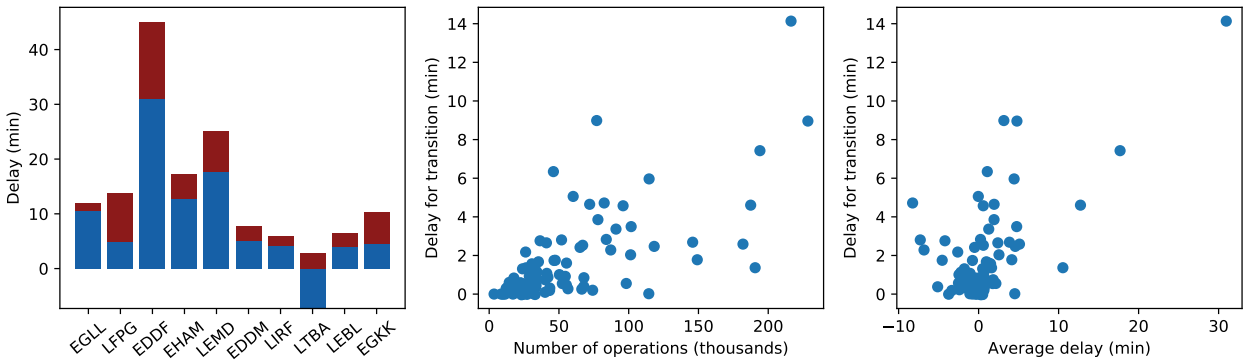

Figure 5.8: Additional delay required to trigger a phase change, for the 10 biggest airports (Left), and as a function of the number of operations (Centre) and of the average delay (Right). Reprinted with permission from [BZ16].

These result suggests that, whilst big airports have been shown more resilient to phase change because of their internal resources, their sensitivity to disturbances is heterogeneous, as it depends on the daily expected delays. For example, both Heathrow (EGLL) and Frankfurt (EDDF) need an increase of 10% in the average delay to switch to the disrupted phase. However, the lower expected delays in EGLL (because of tight over-performing internal procedures) makes the additional delay to reach the disrupted phase much lower than the one for EDDF. For that reason, Fig. 5.8 (center and right) display the additional delay for disruption as a function of respectively the number of operations and the average delay of the airport, showing (small positive correlation) that the resilience of an airport to perturbation is both linked to its traffic and to their being accustomed to handle important delays.

5.1.3 Case of China

The study of the Chinese delay propagation network have been narrowed to the standard (Granger-related) patterns in accordance with the data provider's preferences.

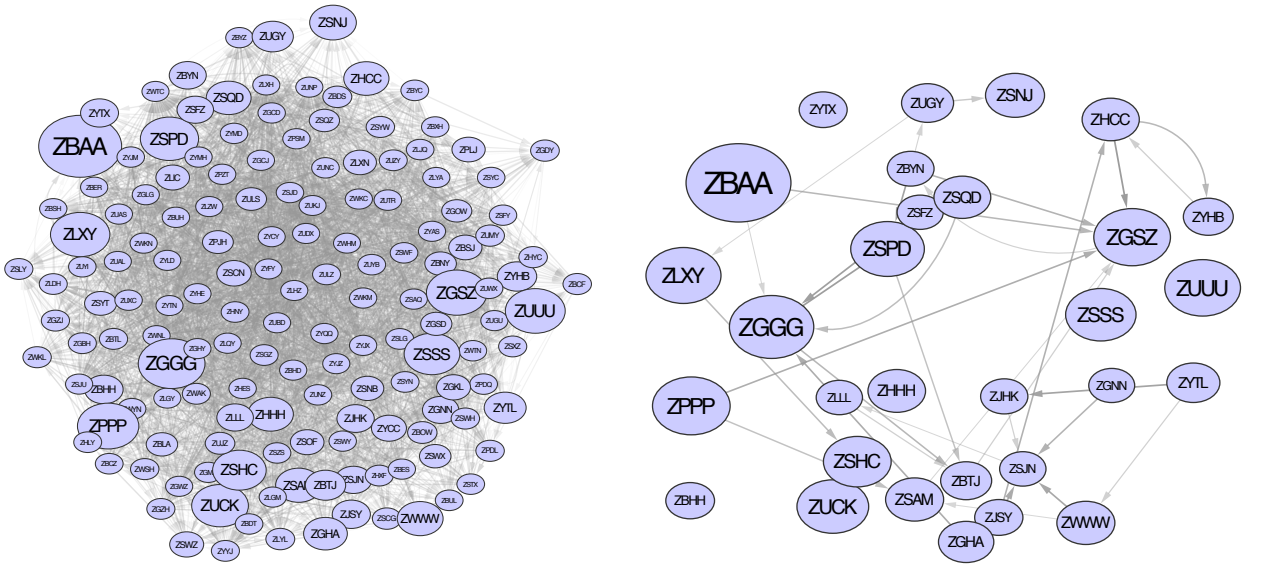

Figure 5.9: Chinese airports delay propagation networks obtained with Granger. (Left panel) Complete network; (Right panel) network corresponding to the 30 biggest airports in term of traffic. The size of nodes is proportional to their number of connections. Reprinted with permission from [ZBY16].

Delay propagation through airports

Similarly to what has been done for Europe, Fig. 5.9 represents the GC delay propagation network. It is worth remembering the reader that, while GC is a powerful tool, it remains ill-designed for sporadic time series, which might be the case of many small airports that suffer delays only at some specific times of the day (therefore making the other hours of the day irrelevant). As such, the number of outbound connections (*i.e.* causing links) for small airports might be inflated (as correlations looking like causalities are more probable with fewer data - see curse of dimensionality in Section 2.1.5). However, the results presented will remain quantitatively valid.

The previous network has been complemented by a sub-network composed by the 30 busiest airports (Fig. 5.9, right panel), where its rather sparse property can be easily observed.

Following the script of the analysis performed on Europe, Fig. 5.10 (left and central panels) displays the number of outbound and inbound relations of each airport, as a function of the number of operations, with similar conclusions holding.

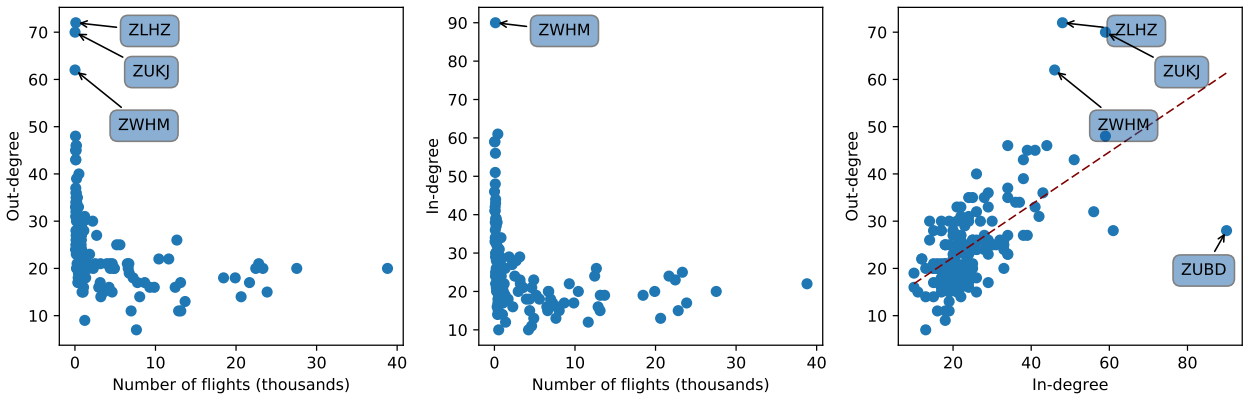

Figure 5.10: Analysis of nodes' role in the airport delay propagation networks. Left and central panels represent the out- and in-degree of airports as a function of their number of operations. The right panel depicts the relation between the in- and out-degree. Reprinted with permission from [ZBY16].

ICAO	Airport name	Num. flights	Out-degree	In-degree
ZLHZ	Hanzhong Airport	54	72	48
ZUKJ	Kaili Huangping Airport	7	70	59
ZWHM	Hami Airport	6	62	46
ZYFY	Fuyuan Dongji Airport	34	48	59
ZWNL	Nalati Airport	84	46	34

Table 5.2: Most central airports, according to the out-degree. Reprinted with permission from [ZBY16].

Similarly to the European airports, an inverse relation is also observable for the Chinese ones, where the smallest airports play the central roles in the propagation of delay. Furthermore, a direct relation can be found between the number of incoming and outgoing links of an airport (see right panel, red dashed line of slope 0.618, $r = 0.643$) suggesting that all airports contribute to the propagation of delays in a symmetric way, as further discussed in Section 5.1.3. Some exceptions can be found. ZWHM, ZUKJ and ZLHZ mostly transmit delays, while ZUBD absorbs them. However, the presence of ZWHM and ZUKJ as all-causing airports must be taken with care. Tab. 5.2 specifies the number of flights actually landing at 5 chosen airports during the time window considered for the study. It is quite remarkable that Kaili Huangping Airport (ZUKJ) and Hami Airport (ZWHM) respectively operated 7 and 6 flights in the considered time window, thus their presence within the most important airports for delay propagation is explained in light of the Granger Causality metric shortcomings.

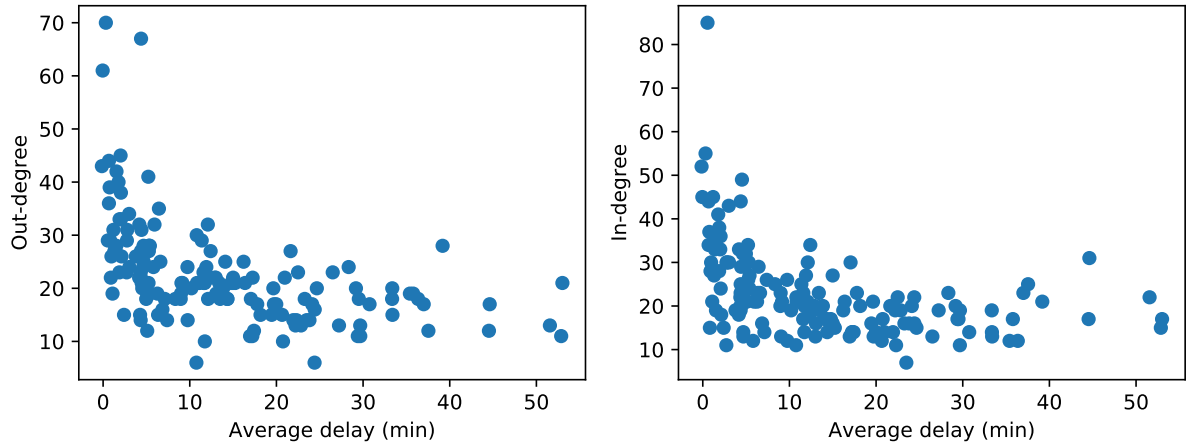

Figure 5.11: Out- (left panel) and in-degree (right panel) of nodes, as a function of the average delay observed in the corresponding airport. Reprinted with permission from [ZBY16].

Additionally, Fig. 5.11 depicts the in- and out-degree of nodes as a function of the average delay observed at each airport. Due to the high correlation between airport sizes and delays, results are qualitatively similar to what reported in Fig. 5.10 left and centre.

The ZUKJ and ZWHM cases might suggest a generalisation of the GC error. Thus, a validation of our results is required. To that end, the time series of Beijing Capital International Airport (ZBAA) and the ones of the 21 airports detected by GC as causing its delay dynamic have been further studied. Specifically, we want to ensure that our results are not the product of noise. Different level of random noise (with zero mean and increasing variances) has been

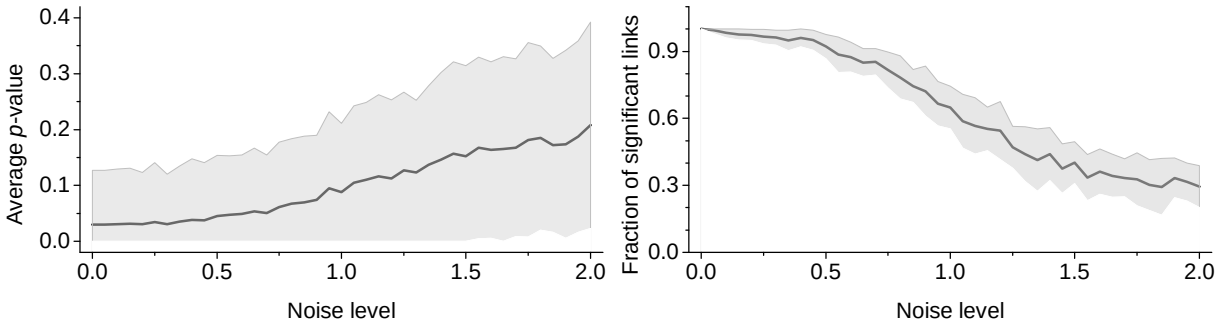

Figure 5.12: Average p-value (Left) and average fraction of significant links (Right) between ZBAA (Beijing Capital International Airport, the largest airport in China) and the 21 airports Granger-causing it, as a function of the quantity of noise introduced in the time series. Black lines and grey bands respectively represent the average value and one standard deviation. Reprinted with permission from [ZBY16].

Metric	Chinese Network	European Network
Link Density	0.2960	0.1820
Transitivity	0.3502 (17.193)	0.2703 (-0.2227)
Efficiency	0.6480 (-0.0149)	0.5910 (1.7912)
Assortativity	-0.0108	-0.2227
Diameter	2	3
Information Content	1.1252	0.9353

Table 5.3: Resume of topological results. Reprinted with permission from [ZBY16].

therefore added iteratively to the causing time series. Fig. 5.12 represents the evolution of the average causalities' p-value between the 21 airports and ZBAA (left panel), along with the average fraction of subsequent statistically significant connections (right panel), as a function of the variance of the noise. It is clearly observable that noise confounds the GC test, such that half of the original links are missed at a noise level of 1.0 (which is tantamount to the standard deviation of the 'normalised' z-score time series that we have considered). If the detected causalities were merely the consequence of random fluctuations, their p-value would be independent of the noise level. As such, the asymptotic fraction of remaining significant links represents the error of GC caused by the time series sporadic behaviour. The real structure of the delay propagation process would then be the group of links that had vanished with the addition of noise. However, ZUKJ and ZWHM cases are different from the European network (see Section 5.1.2) where there was effectively an airport (corresponding to a touristic destination - and therefore over-connected) playing a central role in delay propagation. The difference is partly framed within the number of flights that operates in the smallest Chinese airports. ZUKJ and ZWHM, for example, recording less than 10 flights in one month (*i.e.* a predominance of zeros values in the airports delay time series), fool GC into overfitting and thus into false positive detection.

Finally, Tab. 5.3 reports the information corresponding to the five most relevant metrics for delays propagation, compared to the Europeans' values. The Chinese standard (*i.e.* GC) network is characterised with a high Information Content (low regularity), a high transitivity (many triangles), but a small inefficiency (bad propagation) coupled with a small dissortative behaviour. All these results, grouped together, confirm the description done of the Chinese

market structure in Section 4.1.2: the high regionalisation of the Chinese market creates isolated highly infra-connected sub-areas.

Airlines' contribution to delay propagation

As we have described in Section 4.1.2 and Tab. 4.2, the Chinese market is dominated by three state carriers: Air China (ICAO code CCA), China Eastern Airlines (CES), and China Southern Airlines (CSN). Would it be possible to study the propagation delay network of each one of them? For that, the notion of multi-layer (see Section 2.3.4) is necessary. Indeed, the global propagation process presented in the previous sub-section is obtained by projecting all layers - corresponding to individual airlines - into a single network [Zan15]. However, that projection is not a linear process. In fact, a propagation between A and B detected in the global network might be inexistent (*i.e.* statistically not significant) in each individual layer. The contrary might also happen, as the presence of A-B connections in several layers of the multiplex might be cancelled out when the single layer projection is considered. The non-trivial relationship between individual airlines and their global projection (coupled with the high regionalisation of China's airspace) is making the analysis of these three dominant airlines interesting.

The same process as before is here applied to each airlines' layer. Fig. 5.13 represents the airport in- and out-degrees as a function of the number of the carrier operations at the same airport. A sharper inverse relation than the one already observed in the global propagation network (see, Fig. 5.10) is present here. Indeed, most delays seem to be propagated by and to small airports. Let us consider the following scenario: a late arrival at a small airport is not absorbed (because of the limited resources of the airport), and is then propagated back to a central hub; however, the delay suffered by the hub will be diluted considering its much higher traffic. Nevertheless, we may expect it to be subsequently propagated to another small airport, therefore creating the link between two small airports. Let us validate this scenario by categorising flights into the four aircraft types presented in Tab. 5.4.

Group D - mostly executive jets and very small aircraft - have been discarded, yielding three sets of time series, representing the hourly average landing delay by flights executed by each type of

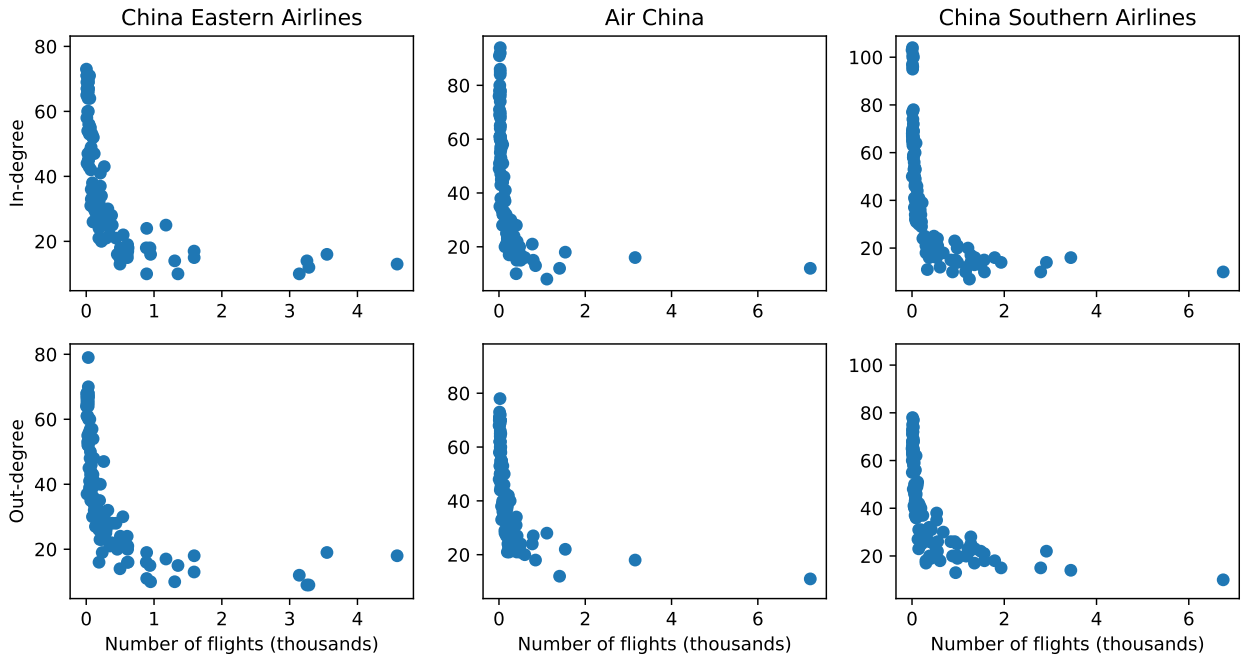

Figure 5.13: In- (top line) and out-degree (bottom line) of nodes as a function of the airport's number of operation, for the three most important Chinese airlines (Left: Air China; Centre: China Eastern Airlines; and right: China Southern Airlines). Reprinted with permission from [ZBY16].

Class	Type	A/C examples
A	Wide body	A330-300, B757-200
B	Narrow body	B737-800, A320
C	Turbo prop	MA60, F50
D	Others	

Table 5.4: Classification of aircraft.

aircraft, and the subsequent GC network. The only statistically significant connection resulting from this test is $C \rightarrow B$, that is, that turbo prop delays propagates to narrow bodies. In other words, turbo prop aircraft, which usually collect passengers from small airports, propagate their delay to larger aircraft - narrow bodies - which usually operate on a longer range. In our previously described scenario, turbo props would represent the first aircraft, from small airport to hub (transporting the delay from the small airport to the hub). The delay at the hub would not be reflected in the time series because of the higher traffic and the hub's higher delay mitigation resources. However it will be propagated to another small airport by medium range flights (narrow bodies) or another regional flight (turbo prop).

Metric	CCA	CES	CSN
Num. of nodes	99	108	114
Link Density	0.6359	0.5235	0.5313
Transitivity	0.7458 (22.644)	0.6638 (29.161)	0.6732 (32.386)
Efficiency	0.8180 (0.0504)	0.7618 (0.0072)	0.7656 (-0.0045)
Assortativity	-0.1678	-0.0474	-0.1356
Diameter	2	2	2
Information Content	0.7313	0.8457	0.7784

Table 5.5: Resume of topological results. Reprinted with permission from [ZBY16].

The similarity between the structure of each airline carrier is quite remarkable. Tab. 5.5 reports the principal topological metrics for the three studied airlines, with very akin values. This is principally linked to the regionalisation of the Chinese market, where a delimited region of the airspace has been handed out in majority to one state carrier. Therefore, the structure might be comparable beyond the fact that they operated in different regions. However, whilst layers are quantitatively identical, they present a higher transitivity and a lower assortativity and Information Content than the global propagation network presented in Tab. 5.3, reflecting a propagation process characterised by the abundance of triangles centred around hub airports. Such a structure supports the idea of a process in which delays are generated at small airports, for then being propagated to one of the hubs of the system and then back again to small airports. This hub-and-spoke structure might seem contradictory with the previously point-to-point structure announced in Section 4.2.1. However, it is important to note that the network represented in Section 4.2.1 links airports when a direct flight is encountered between them. The network constructed in this section is fundamentally different, as it is based on the detection of the delay propagation process, which - as explained - can happen between two airports not even linked by direct flights. As such, the two structures describe different propagation processes, and as such are not directly comparable.

5.2 Non-linear phase changes in delay propagation networks

5.2.1 Context

In spite of its success, GC suffers from the limitation to linear causations, as it neglects the expected non-linearity of delay propagation. However, this restriction is not inherent to the definition of causation. In other words, the two axioms of a causal relationship defined in Section 2.2.1 are still valid for non-linear (and non stationary, for that matter) time series (*i.e.* the cause must still precede the effect, and the former must improve the prediction of the future dynamics). The auto-regressive models behind the definition of the GC are the limiting factors. Their substitution by more complex and non-linear forecasting approaches would therefore allow to surmount the linear limitations, while still fulfilling the causation axioms. ANNs (see Section 2.1.3) appears to be a strong candidate in this context, as for their general characteristic: given some input features x and an output variable y such that $y = f(x)$, neural networks can find a good approximation $\hat{f} \approx f$ independently on the characteristics of f (*e.g.* non-linearity). However, this comes at a cost: large sets of training data are required to extract complex relationships, and the choice of the internal parameters (*e.g.* number of hidden neurones, number of layers, or the weights' initialisation) is usually arbitrary.

One may *prima facie* think that a neural network-based causality could be achieved by simply substituting the auto-regressive models by ANNs. This may nevertheless yield wrong results due to two problems: the *curse of dimensionality* and the stochastic nature of ANNs.

The former refers to the fact that the two models (respectively with and without past values of X) do not have the same number of input features: by including elements from X, the forecasting model may overfit the data, resulting in an over-estimation of the causality relationship. The stochastic nature of ANNs also implies that one single model may end in a local minimum, thus introducing noise in the result.

To tackle these pitfalls, we propose an algorithm called Neural Network Causality (NNC), which

leverages on the idea that not including information about X is tantamount to include such information when its temporal structure has been destroyed. Based on this, NNC constructs a first model using X's and Y's past values to predict Y current observations (main model). It then compares it with a model that uses Y past values along with shuffled X's past values (control models). If the prediction of the second model is significantly worse than the former one, it is possible to conclude that the shuffling deleted valuable information in the past of X, and thus that X causes Y. Thanks to the generality of ANNs and some validation adjustments (*i.e.* curse of dimensionality), the yielded spectrum of possibilities is much wider than the one allowed by the Granger's initial implementation. Most notably, this allows to overcome the aforementioned GC limitations (linearity and the stationarity requirement), resulting in a much general framework for the detection of causality in real-world time series. While the original idea is not new [LWG93, PFT+99], our proposal differs in several important points:

- We do not just present some theoretical results, but a full Python library implementing an ANN-based causality metric. The non-trivial nature of the implementation of ANN models has probably been the main barrier towards a more widespread adoption of this kind of causality metrics.

- We include a complete analysis of the validity of the forecasting models, based on the shuffling of historical data and on the data mining principle of cross-validation (see Section 2.1.4). This allows to exclude situations in which spurious causalities can appear because of model overfitting. The validation procedure is completely automatised, and based on the division of the data into multiple training and validation sets.

- The library is built upon TensorFlow. The TensorFlow API [AAB+16, RG16], initially developed by Google Labs and now available as open source, is a flexible framework for implementing different types of ANNs. TensorFlow is maintained by an enthusiastic community, with new versions and new features appearing on a weekly basis. The use of this library ensures a computational efficient implementation; an easy scalability, up to thousands of processors; and long-term support.

Before proceeding with the analysis of the delay propagation process in Europe, we encourage the reader to refer to the Annex A for complementary information about the mechanism behind NNC Python library, and several results attesting its solid performance as opposed to Granger.

5.2.2 Network analysis

The question of the non-linear propagation of delay has not extensively been studied in the literature, as much of the focus has been centred, for example, on the delay multiplier metric - a metric designed to detect linear correlations [BHBR99]. We will show in Section 6.1 that the function driving propagation is more complex than expected, especially because it presents a high heterogeneity across airports, and that non-linearities might be triggered when high magnitude delays are suffered. Specifically, we will show how some airports present a supralinear growth of the outbound delay for severe perturbations, while others seem to have enough operational buffers to control extreme situations. High delays are therefore more propitious to *explode*, as they can be non-linearly propagated; ergo the importance of detecting the non-linear propagation patterns.

By applying the NNC causality test to each pair of airports and their corresponding stationarised delay time series, we are able to construct a functional network - analog to Fig. 5.3 - representing non-linear propagations within the system (Fig. 5.14). Note the yet strong presence of LMML (Malta airport), reinforcing the idea - suggested earlier by the GC Network - of its central role in delay propagation[5].

Does LMML propagates delays in a linear fashion as suggested by GC? Or in a non-linear fashion as suggested by the NCC network? The answer lies within the characteristics of both causality metrics. First, GC is able to detect non-linear couplings[6] under small perturbations, *i.e.* small delay magnitude. Secondly, NNC (with default parameters) does not detect linear causation if the relation is weak (see results in Annex A). Finally, both successfully detect

[5]NNC metric results are confirmed through a validation process by performing a shuffling of the time series (see Annex A for more details).

[6]Such is done considering the linear part of a non-linear function (which is equivalent to extracting the Jacobian of that function) as a good approximation for small magnitudes.

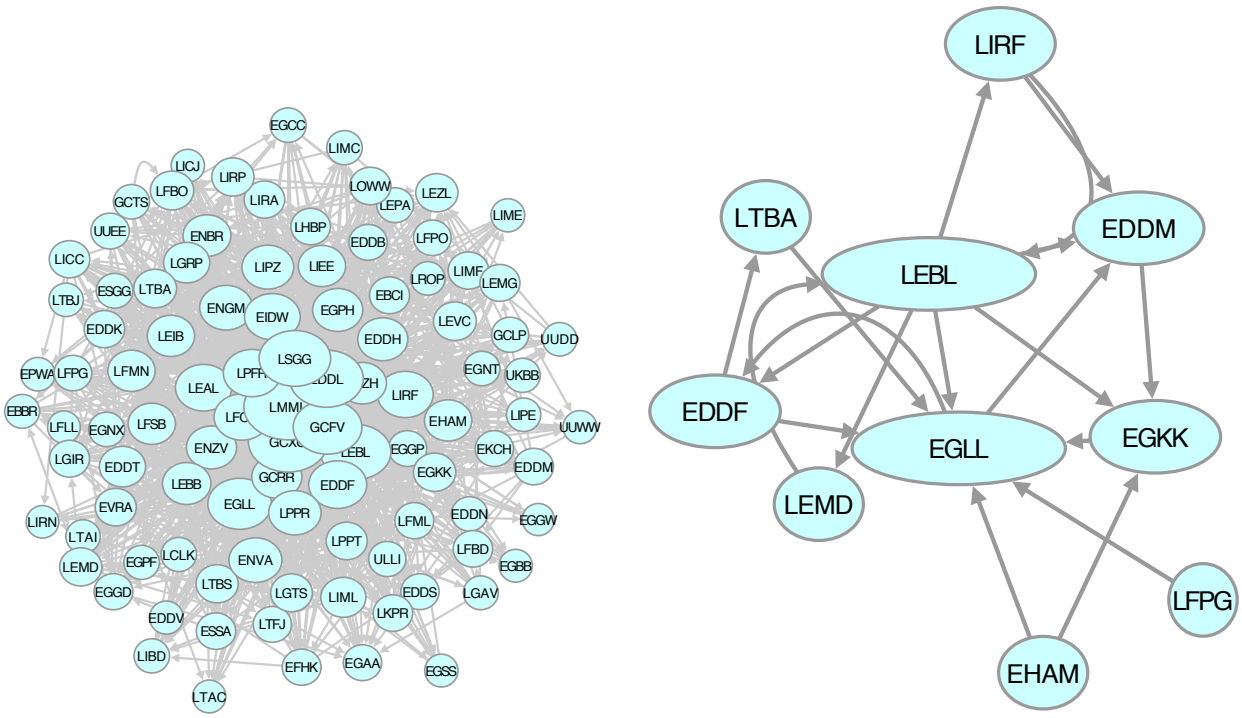

Figure 5.14: The left panel represents the complete NNC network, the right one the network corresponding to the 10 biggest airports. In all cases, the size of nodes is proportional to their degree.

non-weak linear relationships or strong non-linear causation for small delays. As such, the deactivation in the NNC network (Fig. 5.14) of the links present in the GC network (Fig. 5.3, left panel) yields a representation of the non-linear propagation network (see blue network in Fig. 5.15). Similarly, deactivating the NNC links in the GC network only leaves weak linear relationships - see green network in Fig. 5.15). Finally, the network composed of links present both in NNC and GC networks represents linear and small perturbation's non-linear propagations (see yellow network in Fig. 5.15).

The LMML airport's utter centrality, being spotted by both NNC and GC, suggests either a strong linear propagation of LMML delays or their systematical non-linear propagation when their magnitude is low. The second option, *i.e.* non-linear propagation of small delays, is in accordance with the EE network's results. Indeed, the low magnitude of the delays at LMML combined with their low variance results in a stationarised time series with high kurtosis, and subsequently to the deficiency of extreme unexpected delay, thus explaining the inexistent role of the LMML airport in EE propagation.

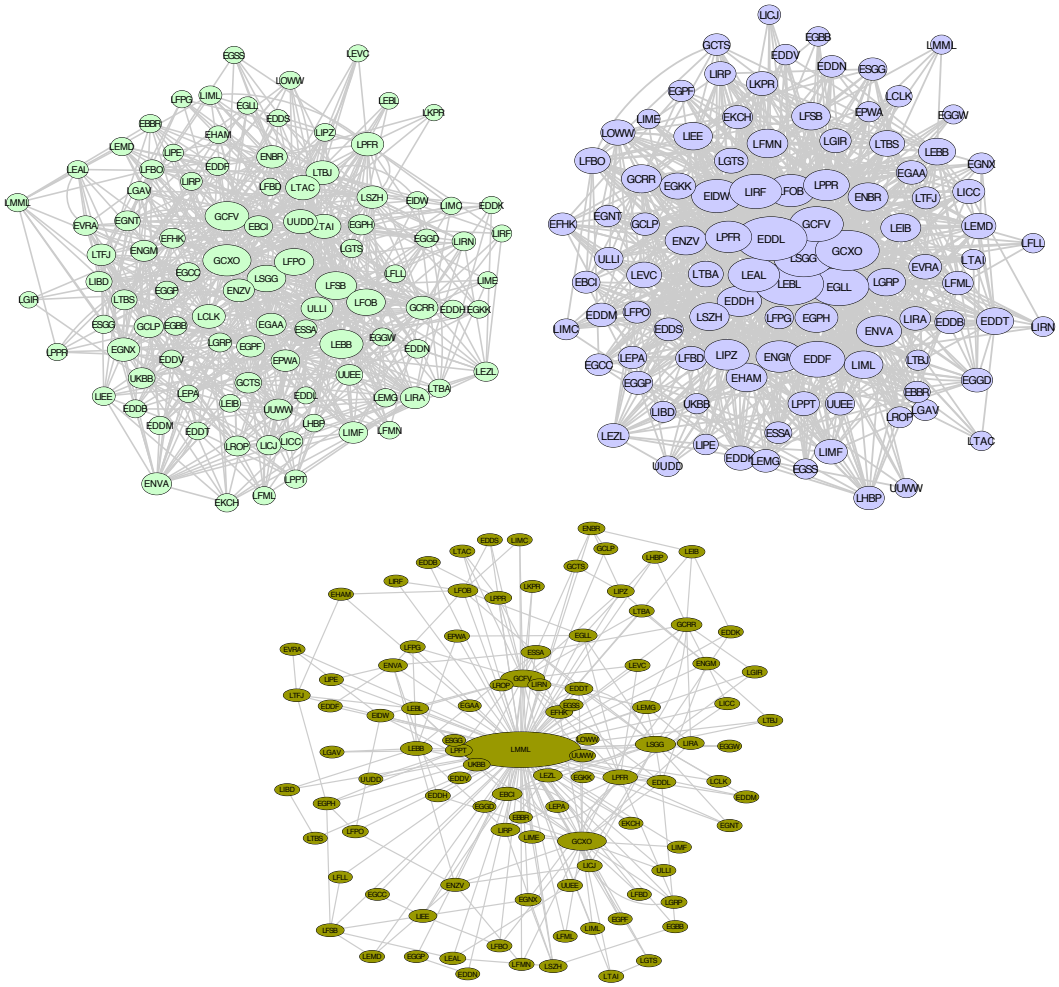

Figure 5.15: Linear, non-linear and low magnitude delay propagation networks, respectively in green, blue and yellow. See main text for more details. Size of nodes is proportional to the node's number of connections.

Also, big airports seem to propagate their delays in a non-linear way. It can be appreciated how the connections between the 10 busiest airports are more densde in Fig. 5.14 than in Fig. 5.4. The connections between big airports are therefore non-linear, but also convey important (*i.e.* high magnitude) perturbations [7]. To fully characterise the centrality and the role of the airports in both the weak linear (green network) or non-linear (blue network) networks, we have to analyse the ratio of inbound per outbound delay propagation connections. Fig. 5.16, left and right panels, displays such ratios respectively for the weak linear network and the non-linear one, as a function of the number of operations registered at the airports. It is easily observed how large airports have significantly much more inbounds than outbounds connections when

[7] Otherwise GC would be able to detect them

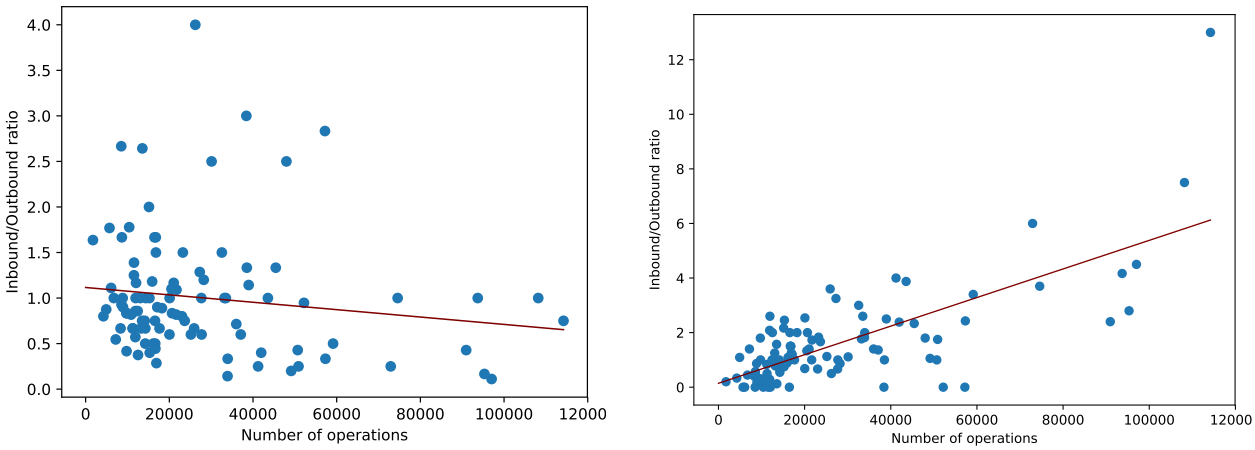

Figure 5.16: NNC Network links analysis. Left and right panels represents the number inbound link per outbound one for, respectively, the weak linear (green network) and non-linear (blue network) propagation networks.

the propagation is non-linear (right panel). Inversely, small airports have more outbounds than inbounds links. This is further corroborated by a positive linear fit ($R^2 = 0.518$). On the other hand, the ratio of inbounds and outbounds connections seems to be decorrelated from the airport size for the weak linear propagation network ($R^2 = 0.021$, left panel). This distribution suggests a tendency of delays to propagate non-linearly from smaller to bigger airports while small linear propagation channels appear more randomly within the network. Such difference between both networks is further confirmed by the lower IC (more mesoscale structure) and the higher dissortativity (*i.e.* the tendency of nodes to link with nodes of different degree) of the non-linear network, as reported in Tab. 5.6. Both networks are nevertheless characterised - in different measures - by a high transitivity and a low efficiency.

Metric	Weakly Linear Network	Non-Linear Network
Link Density	0.076	0.110
Transitivity	0.163 (22.363)	0.146 (5.290)
Efficiency	0.178 (-2.613)	0.123 (-5.021)
Assortativity	–0.184	-0.357
Information Content	0.786	0.651

Table 5.6: Resume of topological results.

5.3 Results and Discussion

It is important to highlight the differences between the approach presented in this chapter and the previous one. On one hand, Chapter 4 focused on a static characterisation of the model, and only approached the dynamics of delay propagation through airport-centred metrics (DM) or simplistic toy models. On the other hand, the work presented in this Chapter focused on the interactions between pairs of airports (encoded in data). In other words, the described functional networks link airports when there is an actual transfer of information (in this case, delay) between them, might it be through direct flights or indirect ones, thus highlighting the propagation dynamics.

To construct the functional networks representing these delay dynamics, we used different causality metrics. Note that the use of a causality metric allows for the description of the delay propagation phenomenon from a global perspective, without thorough explanation of the mechanism driving it. Indeed, as delays are measured at airports, they encompasses indistinctly the effect of airports on the propagation, along with other potential causes like ATFM, late aircraft arrival or passengers connections. All of this, without requiring specific data on each one of these aspects.

In this chapter we extract the propagation delay network through three distinct causality metrics. The first one (GC) aimed at displaying the average expectable propagation flow within the network. The second one (EE) shed light on the disrupted phase, *i.e.* the propagation patterns when abnormally high delay are suffered. Finally, the third one (NNC) is designed to extract the non-linear propagation dynamics. Note that non-linear propagations and propagations triggered by abnormal delay are fundamentally different, see Section 6.1 for more details).

The first two metrics resulted in fairly different network structures, each one describing two phases (normal versus disrupted) of the propagation process, which was expectable as extreme delays are more complex to handle (*e.g.* airline buffers may be inadequate, the closure of a runway may unavoidably reduce the throughput, etc.). To the best of our knowledge, it is the first time such problem was tackled in a quantitative way. We further designed the third

causality metric specifically for the extraction of non-linear behaviour. It appeared clearly that non-linear propagation tends to happen from small airport to busy ones.

Chapter 6

Micro-scale analysis of the system

Truly complete analyses are not the ones referring to a particular scale of size of complexity, nor the ones situated solely at a particular level of the hierarchy - but the ones that contain the deepest explanations at every level. Therefore, the previous macro-scale analysis - however insightful it might be - must be complemented by an analysis from a *transportation* point of view. In this chapter we shed light on what happens on the micro level of the system; that is, we aim at complementing the extracted information about the flow of information within the system with more local behaviours.

6.1 Beyond linear delay multipliers

The first part of this Chapter focuses on one elementary element of the air transport network: airports. However, in order to keep the study consistent with the previously exposed research, we will study the relationship between airports and delays. Chapter 4 already stated the delay performance or dynamics measurement high volatility when the global network is altered by, for instance, a sampling process. In the process, one specific notion was mentioned in Section 4.2.2: delay multiplier (DM) [BHBR99], as an important airport-located tool to understand delay propagation. DM is defined at an airport as the relation between downline (*i.e.* departure delay of the same tail number airplane that had an original delay at landing, denoted as D) and the

initial delays seen by an airport (denoted I). DM is then defined as $DM = D/I$, or $DM = (D + I)/I$[1]. Therefore, high DM values pinpoint a *multiplier* property of airports, as initial inbound delays are, on average, transformed or multiplied into higher outbound delays. Conversely, small DMs suggest the capacity of airports to efficiently dampen, or control, inbound delays. A single - easily computed - number is therefore characterising the capacity of an airport to cope with delays, by assessing whether it is negatively contributing to the delay propagation process of the system or, on the contrary, its internal procedures are adequately reducing the propagation. The metric's simplicity has made it extensively used in the air transport community in order to understand the delay propagation process at an airport [AGE01, ABEG01], across airports [PMO13, FRE13] or, as mentioned before, of an airline [Wu05, ACGB08, Jan05].

However, its simplicity may also be considered a limitation, as no information about the airport internal dynamic is provided - that is, that no details about how and to what extent this multiplication or absorption of delay is done. Specifically, its limitation lies in the fact that it is an average value. Abnormal dynamics are smoothed away by the prominence of normal and expected inbound delays; thereupon, and buttressed by the fact that extreme delays propagates differently (see Chapter 5), DM provides an incomplete characterisation of an airport. The opposite might be true, as a high DM might perfectly be the result of some abnormal events and the poor capacity of the airport to deal with those situations. These questions or uncertainties highlight the necessity of a richer, more exhaustive metric for delays' multiplication.

6.1.1 Metric definition

Data preprocessing

In order to design an appropriate metric, we should beforehand specify and, among another things, preprocess the data we are going to work with. First, flights have been discarded for (a) computational reasons in the absence of executed trajectory data or in case of missing fields; and (b) for a policy oriented reason, as delays of intercontinental flights are not reported. The

[1]Note that this definition, here announced for airports, is perfectly transposable to airlines, among others.

Day	Take-off hour	Landing hour	Origin	Destination	Landing delay	Take-off delay
0	4	6	EDDG	LIMJ	20	100
0	17	17	LIPQ	EGLL	0	5
⋮	⋮	⋮	⋮	⋮	⋮	⋮
285	18	20	ESSA	EDDN	0	0

Table 6.1: Example of the considered data set; see main text for detail. Reprinted with permission from [BPZ17].

remaining 5.7 million flights are extracted from the ALL_FT+ data set (see Chapter 3) as arrays of seven elements, including: the day; the hours of take-off and landing; the origin and destination airports; and the corresponding take-off and landing delays[2]. Negative delays (*i.e.* landing before schedule) are then set to 0.0. The idea behind this common rule resides in the absence - or low magnitude - of reactionary delay (*i.e.* delay caused by another flight's delay) caused by negative delays. Indeed, a flight landing ahead of schedule might trigger delays for departing flights - we could imagine an unexpected use of runway because of the negative delay that would delay the take-off of an aircraft - but will definitively wait until his next planned time of departure to take-off again. On the other hand, the impact of a positive delays is much stronger, as they may postpone the next departure of the same aircraft because of, for instance, turnaround and maintenance minimum times. In this context, the impact of negative delays is negligible, and is set to 0.0 to avoid them compensating for positive delays during the downline (D) or initial delays (I) calculations. An extract of the data at disposal is sketched in Tab. 6.1.

Methodology

We hereby propose a metric that characterises the dynamics of the airport's response to an inbound delay. Specifically, the proposed airport-based metric determines the non-linear relationship between in-bound and out-bound delays. To achieve this, the average in-bound and out-bound delays per airport in 1-hour time windows are calculated. This first step is already contrasting with the DM standard calculation, where the out-bound delay is calculated considering the same flights involved in the in-bound delay, while we believe that other flights, *a*

[2]Computed as a simple difference between the landing (respectively take-off) time and the one expected in the last filed flight plan.

priori uncorrelated, might suffer from a previous flight's in-bound delay.

From there, let us enumerate the steps of the methodology. This modus operandi is compelling as, even if the proposed methodology is *prima facie* easy to understand, it remains important to resume it mathematically. Therefore, the proposed metric consists in:

1. Creating inbound and outbound data per airport and per 1-hour window.
2. Deriving the evolution of average inbound and outbound delays across time windows.
3. Clustering average inbound delays in bins of 1 minute.
4. Creating the corresponding average outbound clusters.
5. Performing a quadratic fit for two shifted inbound and outbound delays clusters.

To achieve step (1), it is required to perform a two-filter selection on the data set illustrated in Tab. 6.1, in order to segregate all flights departing (respectively arriving) at a giving airport. To simplify the description, and without loss of generality, we will consider the example of Madrid airport (LEMD) for the rest of the description. We obtain therefore two subsets of the data, represented as two arrays - $\mathbf{Arrival}_{LEMD}$ and $\mathbf{Departure}_{LEMD}$, respectively listing the arrival and departure delays suffered at Madrid airport:

$$\mathbf{Arrival}_{LEMD} = [(t_0^a, d_0^a), (t_1^a, d_1^a), ..., (t_M^a, d_M^a)]\,, \quad t_k^a \in [0, ..., 6840] \tag{6.1}$$

and

$$\mathbf{Departure}_{LEMD} = \left[(t_0^d, d_0^d), (t_1^d, d_1^d), ..., (t_L^d, d_L^d)\right], \quad t_k^d \in [0, ..., 6840]\,. \tag{6.2}$$

First, note that $(t^a)_k$ and $(t^d)_k$ are two time series running over all the time stamps of the data set. Those time stamps define a specific hour window for a specific day: it is simply computed as the number of hours passed since the hour and day of the earliest registered flight of the data

set ,or in other terms - given that first day and hour available are set to 0 - as 24 multiplied by the day of the flight, plus the hour of take-off/landing (*i.e.* 24 x first column + second or third column in Tab. 6.1). Secondly, $(d^a)_k$ and $(d^d)_k$ respectively represent the arrival and departure delays (sixth and seventh columns of Tab. 6.1). Finally, the two calculated vectors are usually of different length (*i.e.* $M \neq L$) as the number of delayed landing flights has no reason to be equal to the number of delayed taking-off flights within the same hour span. Note that M and L are airport-dependent parameters (here LEMD), as the traffic across airports is quite heterogeneous. However, for the sake of clarity, we avoided the more adequate, but certainly cumbersome, notations M_{LEMD} and L_{LEMD}.

These arrays have to be synchronised, in the sense that each time stamp might appear several times (or not appear) in Eq. 6.1 and Eq. 6.2. Indeed, several flights might land or take-off with delay within the same hour; the opposite is also true, as it is possible that some time stamps did not suffer from delays and are consequently absent from the array (*e.g.* early and late hours with low to no traffic). Also, one might perfectly envision a situation where flights landed with delay while all departures have been conducted on time, leading to the presence of a time stamp on only one of the two previously defined arrays. In order to reduce the noise present in the data, time stamps with less than 3 occurrences (*i.e.* delays) at landing and 3 at take-off are discarded. For each remaining time stamp t_r (*i.e.* with more than 3 delayed landings and departures), the following array is assigned, grouping all delay occurrences with the same time stamp:

$$\left[(d^a_0, ..., d^a_{i_r}), (d^d_0, ..., d^d_{j_r})\right], \quad t_r \in \{t^a_i\}_{i=0}^{M} \cup \left\{t^d_i\right\}_{i=0}^{L} \tag{6.3}$$

The previously mentioned filter implies i_r and j_r to be superior to 3; also array values are usually distinct, as the number of landing delays might be different from the number of take-off operations. The delays are then averaged at step (2), resulting in the following bivariate array

$\mathbf{D}_{LEMD}$, representing the average inbound and outbound delay at each time stamp:

$$\mathbf{D}_{LEMD} = \left[(\bar{d}_0^a, \bar{d}_0^d), (\bar{d}_1^a, \bar{d}_1^d), ..., (\bar{d}_N^a, \bar{d}_N^d)\right], \quad \text{N} \leq \min(\text{M,L}), \tag{6.4}$$

$(\bar{d}^a)_i$ and $(\bar{d}^d)_i$ being respectively the average delay of the landing flights to LEMD at time stamp i and its corresponding average take-off delay.

The aim of this methodology is nevertheless to extract relationship information between inbound and outbound delays (*i.e.* between the elements of $\mathbf{D}_{LEMD}$). In other words, we are interested in knowing what effect would have an increase $\triangle x$ of the average inbound delay on the average outbound delay. For that, step (3) considers the distribution of average inbound delays (*i.e.* $(\bar{d}^a)_i$) present in $\mathbf{D}$, and cluster them into bins of 1 minute, as follows:

$$\text{bin}_i = \left\{k \mid \bar{d}_k^a \in [i-1, i)\right\}, \quad i \in \mathbb{N}_+^*. \tag{6.5}$$

The next step (4) consists in associating each bin with the corresponding departure average delays and average them:

$$\delta_i = \frac{\sum_{k \in \text{bin}_i} \bar{d}_k^d}{\#\text{bin}_i}. \tag{6.6}$$

Note that delays are not necessarily propagated within the same hour - that is, a lag or phase j might be introduced in $\mathbf{D}$, such to detect situations in which an inbound average delay at time stamp i perturbs the outbound average delay at time stamp $i + j$. This is transcribed mathematically as follows:

$$\delta_i^j = \frac{\sum_{k \in \text{bin}_i} \bar{d}_{k+j}^d}{\#\text{bin}_i}. \tag{6.7}$$

At this point, we have partitioned the average arrival delays in parts of one minute and attached to each of these parts - which represent the distribution of the average delay suffered at the

airport - a corresponding lagged average departure delay:

$$\triangle^j_{LEMD} = \left[\delta^j_1, \delta^j_2, ..., \delta^j_p\right], \quad p \leq \frac{\min((\bar{d}^a)_i)}{60}. \tag{6.8}$$

Finally, step (5) performs a square fitting of a polynomial function of degree 2 between the average inbound delay $[1, 2, ..., p]$ (as we clustered in bins of 1 minute) and the corresponding average j-lagged departure delay $\triangle^j_{LEMD}$ (*i.e.* the mean of all average outbound delays happeningj hours after each average inbound delay considered):

$$\triangle^j_{LEMD} = \gamma_{LEMD} + \beta_{LEMD} * [1, 2, ..., p] + \alpha_{LEMD} * \left[1, 4, ..., p^2\right]. \tag{6.9}$$

The previous equation yields two relevant coefficients: β, representing the linear response of the system, or, in other words, the linear relationship between inbound and outbound delays of the airport; and α, the non-liner part of the response.

These two values in their current form are not comparable. We therefore normalise the quadratic coefficient to obtain $\alpha^* = \alpha/\beta$. The methodology we propose finally yields two different metrics:

- A linear coefficient β that represents the correlation between inbound and outbound delays for the phase/lag considered. Specifically, it represents what proportion of the arrival delays is directly and linearly propagated into departure delay. For low positive delay, this coefficient allows to obtain a good enough approximation of the expected outbound delay.

- A normalised quadratic coefficient α^* that gives information about the extreme behaviour of the airport. A negative value of this coefficient corresponds to airports that are able to reduce or mitigate the propagation of high enough arrival delay, such that they do not over-affect departure delays. On the contrary, a positive value means the multiplication of incoming delays into even higher departure delays.

6.1.2 Results

Phase 0

We have initially applied this methodology to the ten most important European airports, in terms of the number of flight movements reported, for a phase of zero. These are: Munich airport EDDM, Adolfo Suárez Madrid Barajas Airport LEMD, Amsterdam Airport Schiphol EHAM, Charles de Gaulle Airport LFPG, Frankfurt Airport EDDF, Heathrow Airport EGLL, Leonardo da Vinci - Fiumicino Airport LIRF, Barcelona El prat Airport LEBL, Zürich Airport LSZH and Vienna International Airport LOWW. Fig. 6.1 reports the results of the methodology for those 10 airports, with the upper panel displaying the fit that has been performed (on a scatter plot of average departure delay as a function of the average arrival delay), and the bottom one plotting the two extracted metrics β and α^*. As it could be expected from the top graph scatter plots, and further confirmed by the bottom β results, outbound and inbound delays are positively correlated.

However, a more surprising result is the distribution of normalised quadratic coefficients (bottom panel), which suggests heterogeneity in the way airports are dealing with high delays. Airports have thus been classified in three categories according to their α^* value: (blue) when it is close enough to zero; (red) when it is highly positive; and (green) when it is highly negative. Note how a negative coefficient (LFPG and LIRF) reports a saturation of the outbound response when high enough inbound delays are suffered at the airport. The opposite (*i.e.* positive coefficient values) is illustrated by EDDM, where high inbound delays provoke a superlinear increase of the response, suggesting that the airport's resources struggle to deal with intensive situations.

The reader may note that the dispersion of points for high delays - probably due to their lower occurrence frequency - might engender statistical fluctuations. Does this invalidate the proposed metric? Can we trust the positive α^* coefficient for EDDM, for example? To validate our results, delays are then reassigned randomly to each flight and Fig. 6.2 (b) clearly show that the randomisation effectively broke the temporal correlation visible in panel (a), transforming

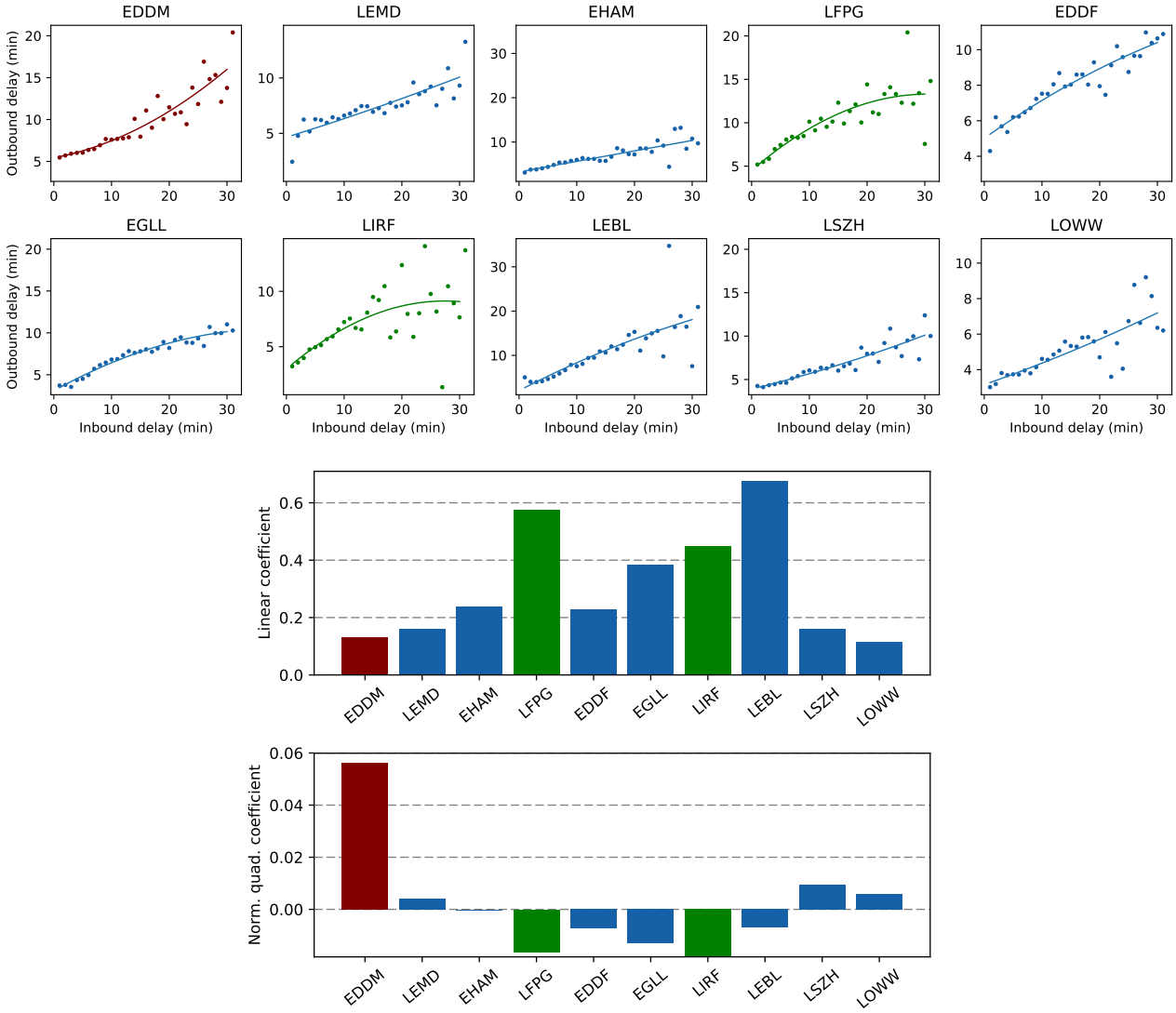

Figure 6.1: Behaviour of the top 10 European airports for a phase of zero. (Top panel) depicts the ten scatter plots of outbound versus inbound average delays within the same hour-window and (Bottom panel) presents sign and magnitude of both the linear and normalised quadratic coefficients. Green and red, respectively, represent airports with highly negative and positive quadratic coefficients. Reprinted with permission from [BPZ17].

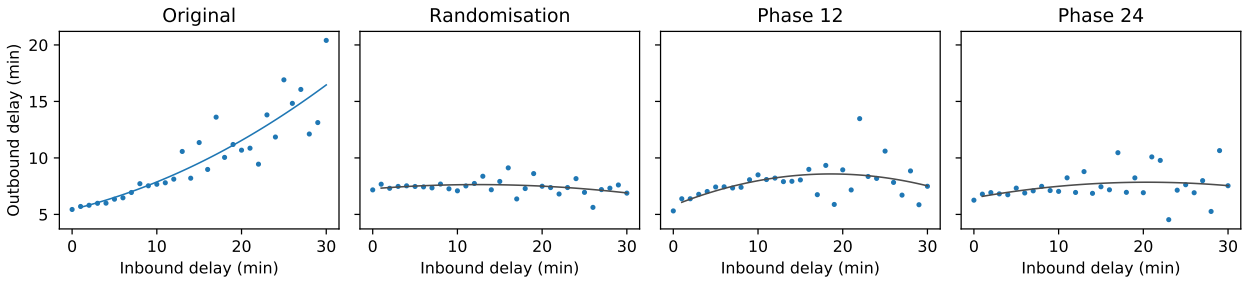

Figure 6.2: Validation of the meaning of the quadratic coefficient. From left to right, the panels represent the scatter plot for the original Munich airport data; a randomisation of the delay data; a phase of 12 hours between inbound and outbound delays; and a phase of 36 hours. Reprinted with permission from [BPZ17].

it into a substantially linear and flat relationship. Panels (c) and (d) represent the relationship when a phase of respectively 12 and 36 hours is considered, following the rationale that no delay is expected to propagate for 12 hours, let alone for 36 hours. In those two cases, the fit is almost linear (with a negative quadratic coefficient too low to be considered relevant).

Also, inbound and outbound delays are considered within the same hour time window, that is, they are matched with a lag of 0 or, in other words, that considered flights are operating within the same hour window. That is strongly suggesting the independence of those flights (*i.e.* that aircraft that are responsible for the inbound average delay are distinct from the ones causing the outbound delay), as not all aircraft have time to operate twice in such short time frame. This basically relates the correlation found between departure and arrival delays to external factors, as high traffic level. [EUR15] reports that the turn-around process requires between 60 and 90 minutes, hence the interest of considering a one hour lag between inbound and outbound delays.

Phase 1

Fig. 6.3 presents the previously performed analysis, now for a phase of one hour, shedding light on the correlation between inbound and outbound delays for a time difference of 60 to 120 minutes; and therefore focusing on reactionary delays, as sufficient time is granted for an aircraft to execute a turn-around. In other words, any found correlation would not be linked anymore to the airport's immediate procedures when handling severe delays, but to the ground

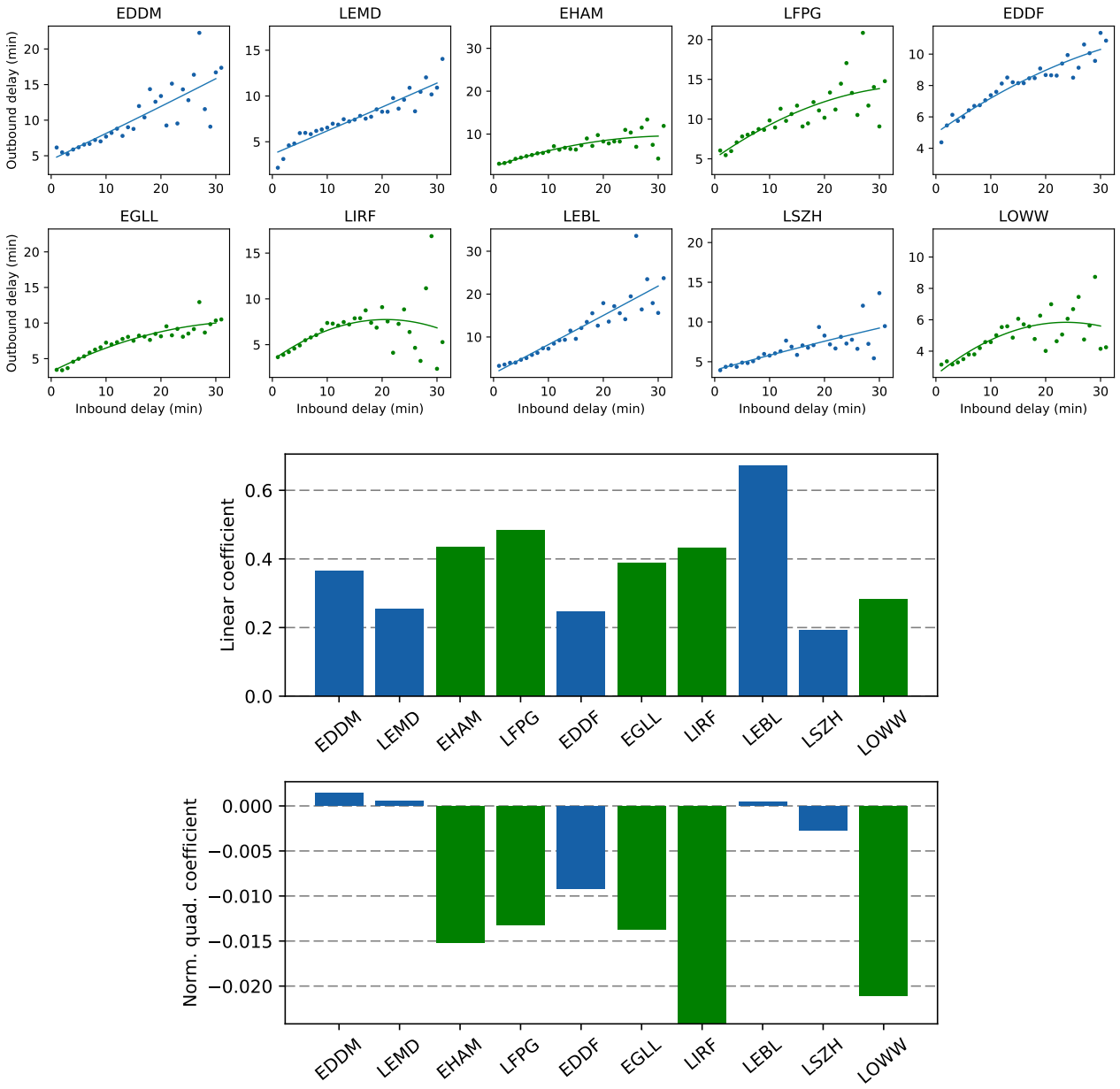

Figure 6.3: Behaviour of the top 10 European airports for a phase of one hour. Panels and colours represent the same information as in Fig. 6.1. Reprinted with permission from [BPZ17].

delay procedures that might further propagate delay. Indeed, a technical failure at an airport is unlikely to last more than one hour, therefore leaving delays of phase 1 to be subsequent to the propagation delay process only.

EDDM does not present a superlinear behaviour anymore. On the other hand, LFPG and LIRF are still - with the addition of EHAM, EGLL and LOWW - showing a strong ability to mitigate those high arrival delays into controlled and saturated outbound delays.

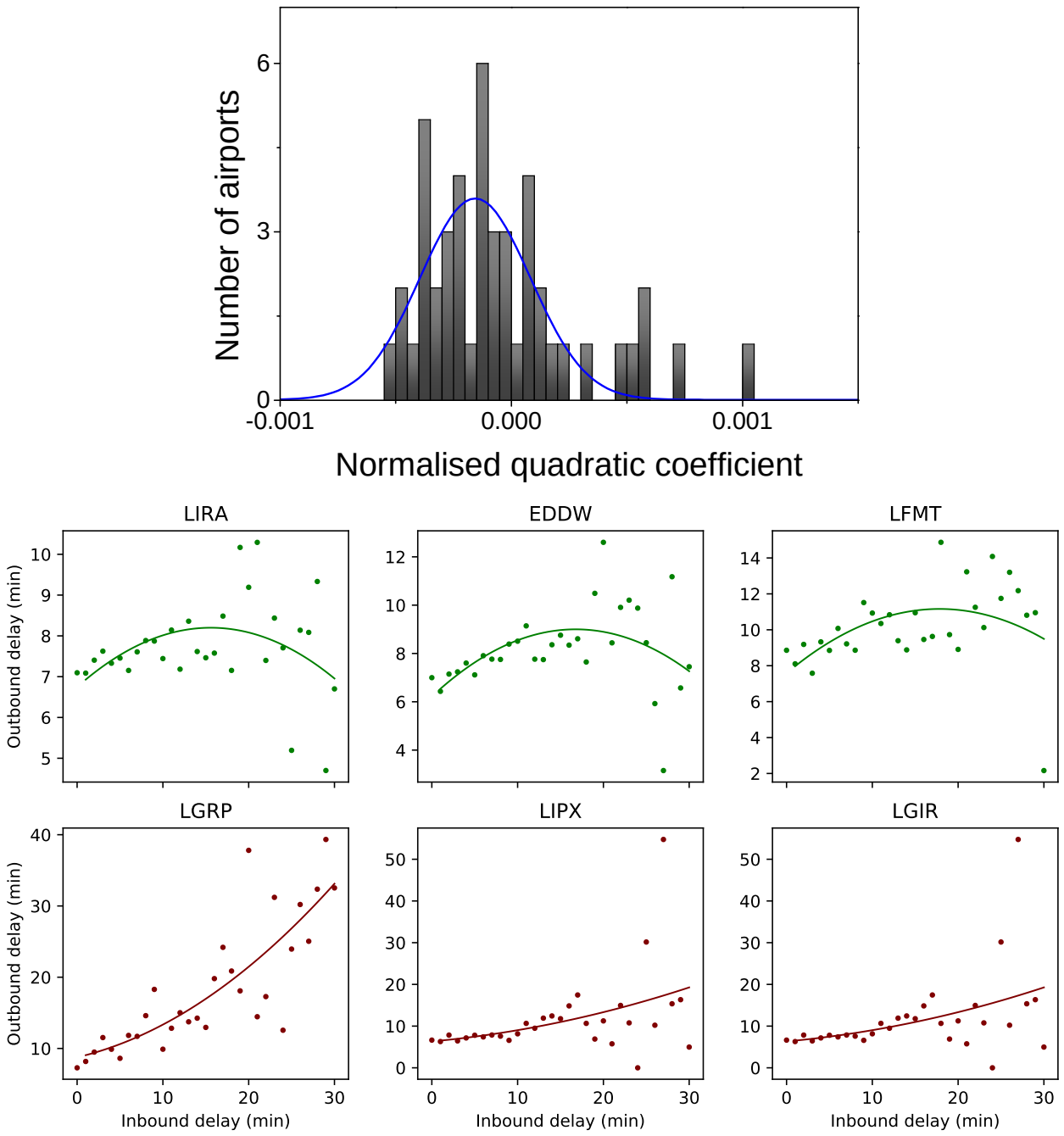

Figure 6.4: Global behaviour of the European system. (Top panel) represents the histogram of quadratic coefficients for the 47 considered airports and (Bottom panel) the three most resilient airports (top line of graphs) and top super-linear airports (bottom graphs). Reprinted with permission from [BPZ17].

Summary for the whole European system

This study have been extended to all European airports. However, to mitigate statistical fluctuations, only airports presenting data for all bins of inbound delays have been considered - that is, an airport is discarded if no average arrival delay of i minutes, with $i \in [1, ..., 30]$, is reported.

Such filter reduces the number of airports considered to 47. Fig. 6.4 (top graph) represents the distribution of their normalised quadratic coefficients, showing that the majority (66%) of the airports shows a negative behaviour (*i.e.* a saturation of the response). However, some airports show the superlinear behaviour that we found for EDDM in Fig. 6.1. The three airports with the strongest superlinear behaviours are plotted in the bottom graph - Rhodes international airport LGRP, Verona Villafranca airport LIPX and Heraklion International Airport LGIR - along with the three most mitigating airports - Ciampino–G. B. Pastine International Airport LIRA, Bremen airport EDDW and Montpellier–Méditerranée Airport LFMT.

Explicative features

The results previously presented are intrinsically linked to each studied airport. Whether they are due to the runway configuration, the size, or any other property, the outbound response is somehow partly framed by the internal procedures of the airport to mitigate the propagation of delays. This should be reflected by the normalised quadratic coefficient being related to some of the airport properties. We have tested six different airport properties as potential explicative attributes for airports' non-linear behaviours, see Tab. 6.2.

The relationship between each one of these features and the quadratic delay multiplier is shown in Fig. 6.5. It can be appreciated that the quadratic coefficient is mostly independent of these metrics, and is of special importance its independence from the airport's number of operation ($R = 0.066$). This contrasts with the results of Chapter 5 (see Section 5.2.2), suggesting a tendency for non-linear delay propagation from smaller airports to bigger ones. However, the reader must consider that such result was obtained in a pair-wise analysis, and that the subsequent relationships might be averaged and statistically smoothed away when each airport in studied independently of the others. It is then expectable that, while almost all airports of the data set propagates non-linearly delays according to the network constructed in Section 5.2.2, only a heterogeneous fraction of them is globally propagating non-linearly, *i.e.* on average across all airports connections.

[3]Notice that this metric is strongly correlated with the number of operations at the airport ($R = 0.805$) as additional runways are constructed in function of the demand

Additionally, the metric's lack of correlation with the standard delay multiplier ($R = 0.067$) suggests that it convey complementary information with respect to the latter. Also note the slight positive correlation with the inbound and outbound delays of the airport ($R = 0.163$ and $R = 0.340$).

Whilst none of the above-listed features considered independently is able to explain the dynamics of the airports delay propagation behaviour, it might be worth to combine them in a predictive model that forecast the value of the normalised quadratic coefficient. Four predictive models have been used to assess whether information is contained within these features: a linear regression model, a LASSO model, a regression tree and a multilayer perceptron (a simple artificial neural network with ten neurones in the hidden layer). More details on these techniques can be found in Section 2.1.3.

As it might be expected, the multilayer perceptron model yields the best results ($R = 0.6653$) explaining 66.53% of the variance of the normalised quadratic coefficient, as its hidden layer allows for the inclusion of non-linearities. All three remaining models behave more or less

Feature	Detail
Airport size	The total number of operations. Note that intercontinental flights, not being considered, might under-rank some hubs.
DM	The airport standard delay multiplier.
Airport delay	The average inbound (black squares) and outbound (green diamonds) delay of delayed flights, in seconds.
Airline Entropy	The entropy of the airlines distribution at the airport - that assess whether one airline dominates the activity at the airport or if multiple airlines are sharing the resources. It is calculated using the first most operating airlines at the considered airport as: $E = \sum_{k=1}^{3} \mathrm{p}(k)\log_2\mathrm{p}(k)$, with $\mathrm{p}(k)$ being the overall fraction of operation executed by airline k.
Aircraft Entropy	The entropy of the distribution of aircraft types operating at the considered airport. Aircraft have been classified into three groups: narrow body (above 80 seats), regional aircraft (below 80 seats) and turboprop. Other types as freight have been discarded or were missing in the data set as wide-bodies as intercontinental flights are not accounted for.
Local Traffic	The number of flights per available runway operating at the airport[3].

Table 6.2: List of features.

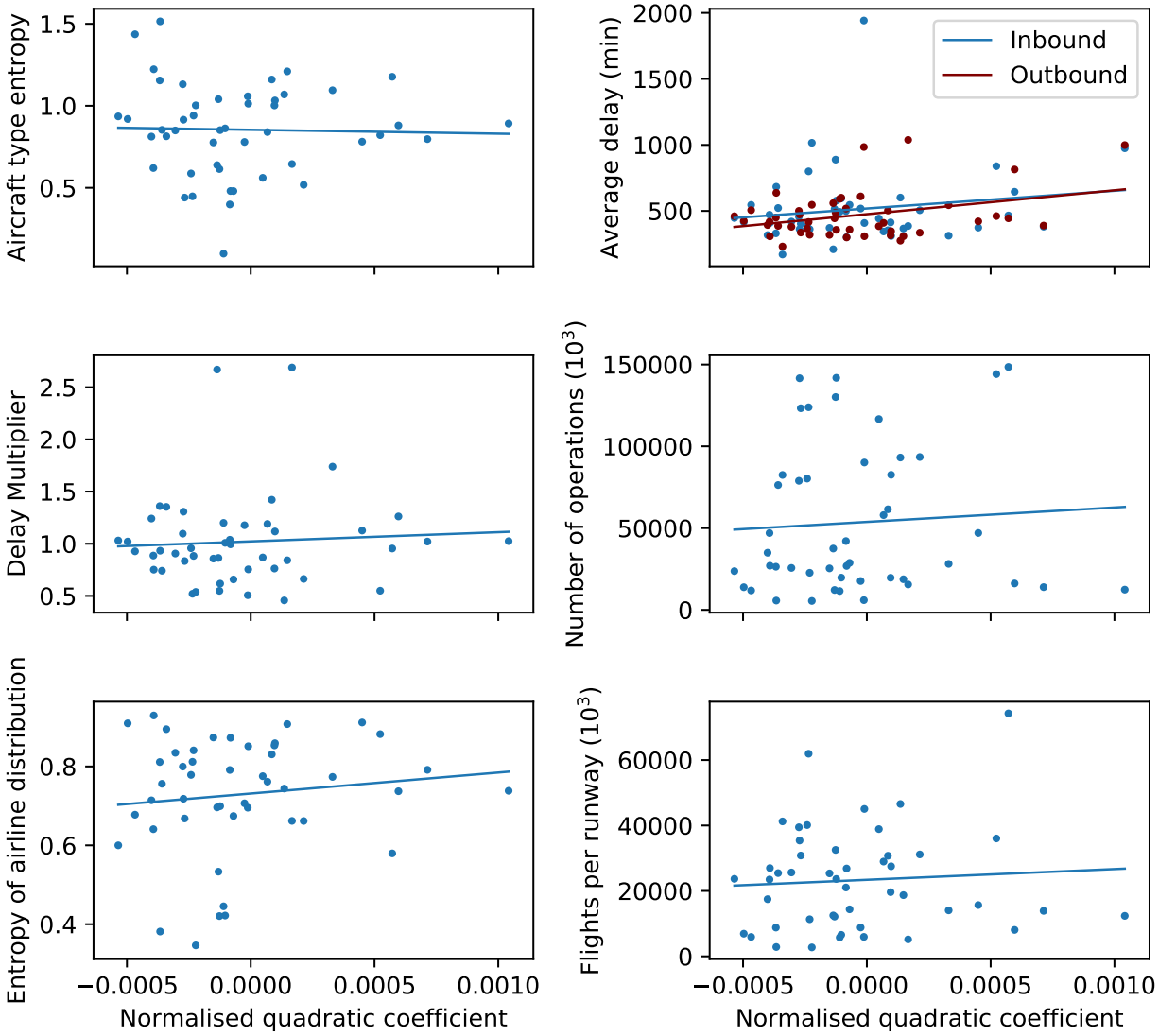

Figure 6.5: Normalised quadratic coefficient as a function of variate airport properties. From top to bottom, left to right, the six panels, respectively, depict the correlation between the Aircraft type entropy and the average delay (inbound in blue, outbound in red); the delay multiplier and the traffic volume of the airport; the entropy of airlines and the number of flights per runway; see Tab. 6.2 for definitions. Reprinted with permission from [BPZ17].

equally, with an R-squared value around 0.54. In spite of its lower performance, Fig. 6.6 represents the linear regression fit, because of its higher interpretability in comparison with the neural network output model. The linear regression model is moreover reinforced by the LASSO algorithm's coefficients, suggesting that no explicative feature can be safely discarded. This result confirms that the dynamics of an airport - delay-wise - are partly framed within its own global characteristics.

It can be added that the R-squared can be increased up to 0.8114 if quadratic relations are

included into the regression model. Nevertheless, such rise comes at the risk of overfitting, giving the number of parameters considered (see the curse of dimensionality pitfall in Section 2.1.5).

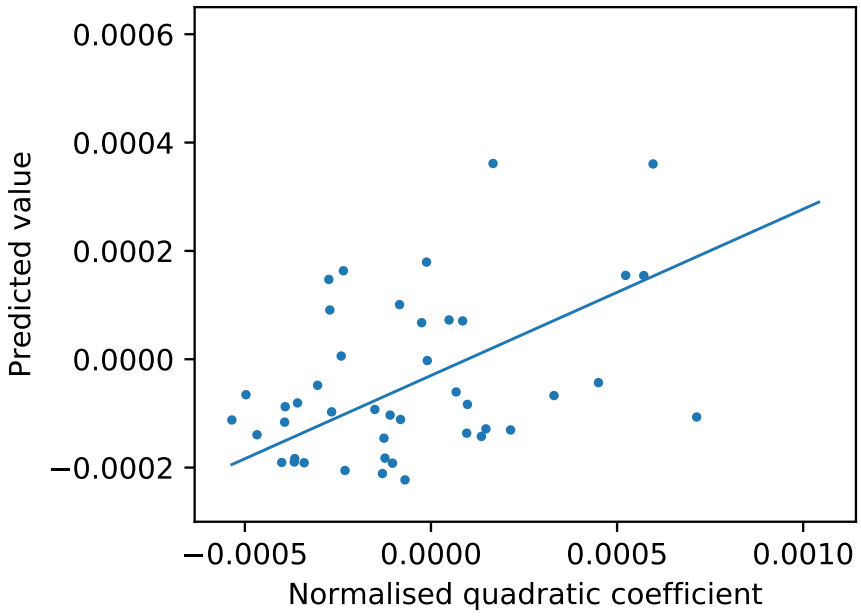

Figure 6.6: Fit for the normalised quadratic coefficient obtained by the multiple linear regression model, using all the parameters specified in Tab. 6.2. Reprinted with permission from [BPZ17].

6.2 Generation and recovery of airborne delays in air transport

The second part of the Chapter focuses on the other main element of the system: aircraft. Flights are the ones actually transporting delays around the system, thus the interest in their relation with delay at a micro-level point of view. Specifically, each minute of delay costs several hundred Euro to European airlines (calculation performed in 2007 [CTA04, Eur07]). Airborne delays - through emission of CO and NO_x - cost much more to airlines if compared to a minute of on ground delay (specifically, one order of magnitude more [CDLHJ07]) - hence the prioritisation of ground delay program in Europe (see Section 4.1.3).

Surprisingly enough, most of the attention of researchers have been devoted to delay phe-

nomenon unrelated to the airborne phase. Examples encompass the quantification of delay impact [CTA04, CTJL09, CT11, FKHS13], the assessment of delay causes [SR10, PWNT09], delay forecasting [ZM06, RB14], the design of strategies for their mitigation [WC02, DP12, NPRN14], the analysis of the appearance of delays due to Air Traffic Management (ATM) regulations, *i.e.* Air Traffic Flow Management (ATFM) delays or caused by the limited capacities of both airports [Han02] and airspaces [Glo96].

However, none or little attention have been devoted to the non-ATFM related delays, and specifically to the causes behind the generation and absorption of delays while the flight is airborne. Whilst the importance of non-AFTM delays in Europe have been under-estimated - mostly because, as said earlier, delays are mainly handled through the ground program - their importance in researchers' minds and industry is expected to grow for two main reasons: (a) a continuously growing traffic and (b) the prospect of a future open-route airspace configuration. In the prospect of the open-route airspace, en-route delays will particularly play a more central role within delay propagation dynamics, as it is fair to expect airspaces to heterogeneously cause or absorb delays. Consequently, in this PhD Thesis, the specification of the causes driving the generation and absorption of delays airborne seems a new interesting challenge to tackle, in light of the future airspace configuration profile.

It is notorious that the reasons behind the apparition of delays are variate, ranging from imperfect accuracy of the pre-tactical planning phase (leading to approximative congestion forecasts) to weather difficulties. Furthermore, negative delays might be generated - that is, a recovery of delay - when the route is shorten, for example. The specification of the cause behind delays is an exercise that requires additional (and usually not available) data. However, a first step toward it would consist in the localisation and quantification of these delay-generating events. Whilst their ecological impact might seem currently controlled, the promise of a much crowded traffic in the coming year highlights the environmental (and regulatory [SSGM12]) importance of airborne delays. We have said earlier [CDLHJ07] that the costs of extra fuel consumption (due to delay) are estimated to be about 6 times higher airborne than on the ground, with 3 times higher NO_x, HC and CO emissions. Thereupon lies the importance of a better understanding of the non-AFTM causes to properly instigate the design of a paradigm to mitigate

them, or at least, to transform them into less environmentally aggressive ground delays.

The challenge ahead of us is far from being trivial, as a simple comparison of take-off and landing delays does not provide insightful information about how a delay was generated. This limitation is tantamount to the one previously encountered with the delay multiplier (DM, see Section 6.1), where only take-off and landing delays are used to characterise the behaviour of an airport. Here we propose a more dynamic approach, where the planned and executed trajectory of a flight are tracked and compared to spot the apparition and recovery of delays. As any dynamic approach, several challenges must be overcome. First, dynamic data - as aircraft trajectories and planned trajectories - are seldom public. Secondly, a consistent metric enabling the extraction of information from the comparison of both trajectories must be created. Finally, this algorithm must be computationally optimised in order to allow the analysis of all the thousands of daily available flights in an acceptable time.

6.2.1 Methodology

Identification of airborne delays

The first step of our study consists in the creation of an algorithm able to detect delay-generating events - or in other words, locate the temporal windows and corresponding geographic positions where a flight is loosing time (or on the contrary, gaining time) against the schedule. As such, an event is defined as an occurrence that modifies the delay of a flight, being the change both positive and negative. Events are then real deviations from planned trajectories, which are tantamount to delays (positive or negative ones). For the rest of the study, the notions of positive and negative delays do not refer to the benefit or problem that a delay might engender, but merely to their effect on the total flight time. Therefore, a positive delay corresponds to a loss of time (*i.e.* a longer travel time) and a negative delay to a gain of time (*i.e.* a shorter travel time) with respect to the initial flight plan. It is important to stipulate one of the main assumptions of this study (its validity and consequences will be discussed further): we neglect the velocity variation of an aircraft, thereupon correlating the notion of velocity and time. A

Algorithm 1 Event Identification Algorithm. Pseudo-code for assessing the presence of delay-generating events. Refer to the main text for further details. Reprinted with permission from [BPZ16].

```
1: def PlannedTrajectory, RealTrajectory;
2: RealTrajectory(:,1) = RealTrajectory(:,1) -
        RealTrajectory(1,1) + PlannedTrajectory(1,1);
3: lerp(k ∈ I ∪ J, PlannedTrajectory)
4: lerp(k ∈ I ∪ J, RealTrajectory)
5: Haversine(latitude, longitude);
6: def PlannedDistanceToArrival, RealDistanceToArrival;
7: def D = Derivative (PlannedDistanceToArrival−
        RealDistanceToArrival);
8: def D = Moving Average (D);
9: #Events = function (Threshold, D):
       def c = Count( Intersection (Threshold, D));
       if c mod 2 == 1
          #Events = ( c + 1 ) / 2;
       else
          #Events = c / 2;
       end
```

change in distance (to the destination) is - with this assumption - equivalent to a change in flight time, thus in delay. This will allow us to freely move from a distance-based metric to a time-based one.

The algorithm proposed for the detection of these delay-generating delays - whose pseudo-code is presented in Alg. 1 - consists in, sequentially:

1. Synchronise planned and executed trajectories.
2. Derive the evolution of the distances to destination.
3. Define events as when the derivative of the distance goes above (or below) a given threshold.

The first step is described in lines 2-4 of Alg. 1. Planned and executed trajectories start at different times, therefore introducing a delay that is not airborne-bound. To keep the focus of the algorithm only on en-route delays, those generated on-ground delays - represented by the initial take-off delay - must be discarded. This is achieved through the creation of a unique

time reference - that is, by a shift in time of the executed trajectory in order to obtain two simultaneous take-offs for both planned and real trajectories. However, this is not yet sufficient to obtain two synchronised trajectories, as each one of them comes with a different resolution - that is, positions in planned and real trajectories are not reported at the same time, making a direct comparison impossible. To deal with this, a set of points is artificially created for each trajectory. Specifically, there are points that exists in one trajectory but are missing in the other. By mentioning a point, we are not referring to geographical points, but to their time stamps. At a given time, a point is reported in one of the two trajectories, but no position is reported in the other. The missing point in the latter is obtained through extrapolation, calculating where the aircraft would be expected to be at that specific time. Therefore, each 2D trajectory[4] is completed by a set of *missing* points, whose time stamps are given by the other trajectory and whose spatial positions are obtained through linear interpolation (lerp). At the end of step (1), both trajectories start at the same moment and their positions are reported for the same set of times.

Fig. 6.7 illustrates the synchronisation process for a randomly selected example flight. The Top panel represents the two original 2D trajectories. The Bottom panel represents the trajectories after the interpolations with additional points, corresponding to a symmetric point in the other route. The synchronised trajectories displayed in bottom figure are mathematically represented as vectors f_l^j of length m:

$$f_l^j = [(i_1, x_1^j, y_1^j)^{(l)}, (i_2, x_2^j, y_2^j)^{(l)}, ..., (i_m, x_m^j, y_m^j)^{(l)}], \tag{6.10}$$

i_k,x_k^j and y_k^j being respectively the time stamp and position of the aircraft l at observation point k. j is 1 when referring to the planned trajectory and 2 for the real position of the aircraft. It can be noted that the vector length m - that represents the total number of available observations - is set for both js by construction (*i.e.* the same number of observations for planned and real trajectory). However, the vector length m can vary across flights and

[4]Altitude has been neglected, as it is seldom used as a way of recovering delay and, except for the Continuous Descent Approach [WH05], has little influence on the delay of a flight.

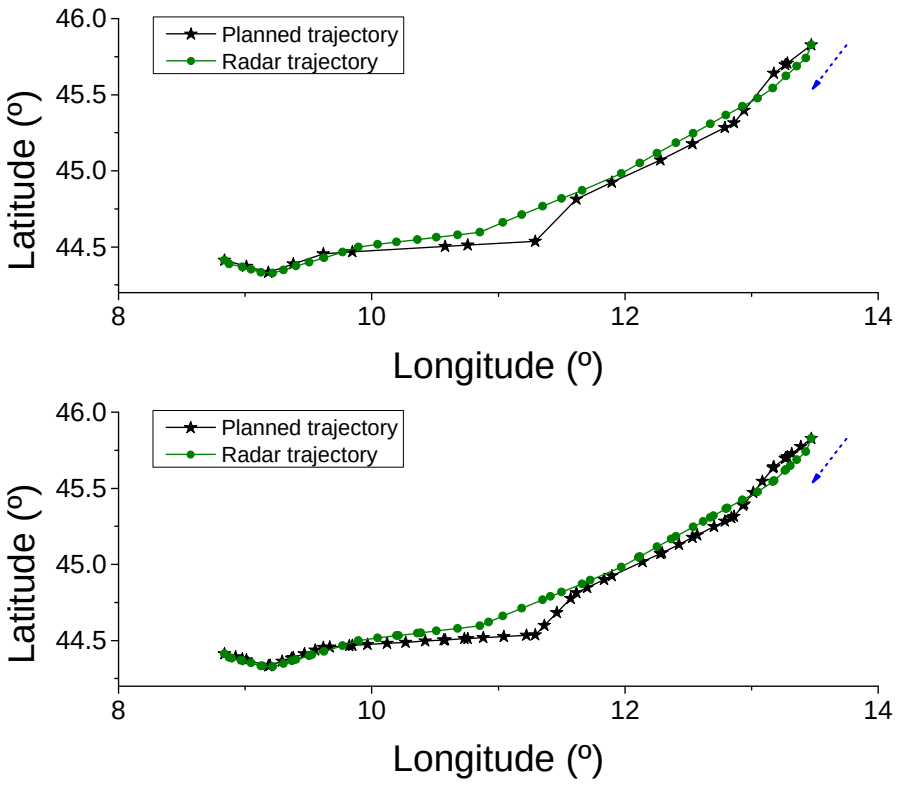

Figure 6.7: Aircraft trajectories synchronisation output. Top panel represents the original 2D planned (black stars) and executed (green circles) trajectories. Bottom panel represents the same trajectories after the synchronisation process, that is after the creation of a shared timeline. Reprinted with permission from [BPZ16].

therefore depends on l; however, for the sake of clarity, the complete m_l notation have been avoided. The same remarks are valid for the time stamp time serie $(i)_k$ - that may vary across flights but is consistent between both planned and real trajectories of a single instance.

The second step of the algorithm is coded in line 5 and 6 (Alg. 1). At this point, both vectors presented in Eq. 6.10 need to be compared, to extract delay-generating events. The most intuitive approach would be to compare the distance of the aircraft to where it is expected to be (*i.e.* the position in the planned trajectory). However, this solution is not linkable with delay, as an increase or decrease of the direct difference yields no *a priori* rule on what is the consequences on the travel time. Indeed, a deviation of the aircraft from the plan (*i.e.* an increase of its distance with respect to the planned trajectory) might correspond to a negative delay, if a shorter route is assigned to the aircraft; or to a positive delay, if the aircraft must make a loop to avoid an obstacle. This solution, while assessing the magnitude of the deviation, and thus the magnitude of the delay, is not able to specify its sign, and as such is not adapted to

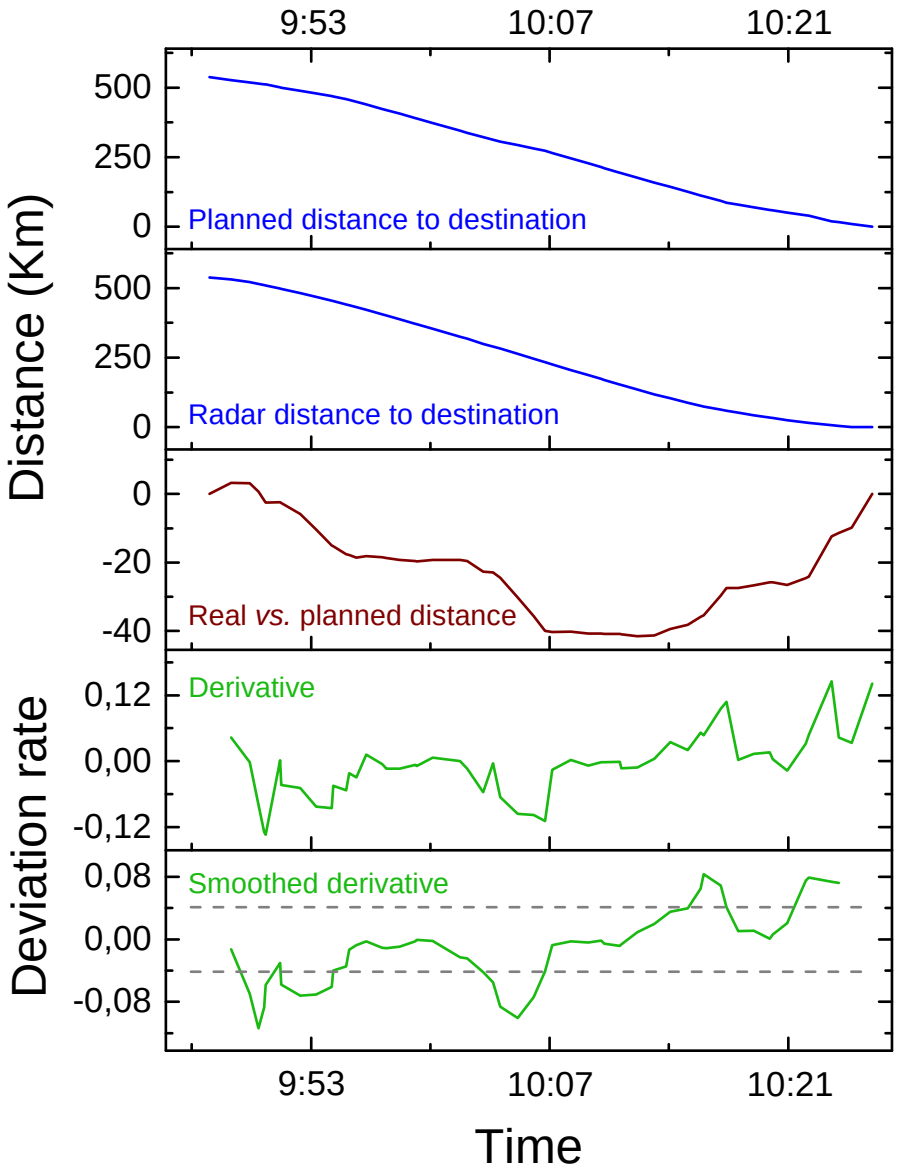

Figure 6.8: Graphical example of the algorithm for detecting delay-generating events. The two top blue panels represent the distance to destination for both planned and executed trajectories of the flight depicted in Fig. 6.7; the central red panel the difference between these two distances; and the last two green panels respectively the standard and smoothed derivatives of the difference. Reprinted with permission from [BPZ16].

our goal. An alternative solution is to compute the distance of both planned and real position with the final destination. A deviation in the course of an aircraft against plan might imply an increase (or a reduction) of its distance to destination (straight line distance), which is tantamount to the occurrence of a positive (or negative) delay-generating event.

To resume, both vectors of Eq. 6.10 are converted to distances from the arrival airport by

means of a Haversine formula [Sin84]:

$$d_l^j = [(i_1, d_1^j)^{(l)}, (i_2, d_2^j)^{(l)}, ..., (i_m, d_m^j)^{(l)}]. \tag{6.11}$$

These vectors are represented in Fig. 6.8 (two top panels) for the same flight presented in Fig. 6.7.

The third (and final) step of the algorithm spans from line 7 to line 9. It consists in deriving from the behaviour of the distance vectors d_l^j the delay-generating events. Let us consider the difference of the real distance to destination with the planned one, represented in middle panel of Fig. 6.8:

$$\triangle d_l = [(i_1, \triangle d_1)^{(l)}, (i_2, \triangle d_2)^{(l)}, ..., (i_m, \triangle d_m)^{(l)}]. \tag{6.12}$$

$\triangle d_k = d_k^1 - d_k^2$ is obtained subtracting to the real distance to destination, the planned distance to destination[5]. Whenever the time series $(\triangle d_l)_i$ is above zero, this corresponds to a distance to arrival greater than scheduled, thus suggesting that the aircraft is lagging behind the planned trajectory. Conversely, whenever it reaches negative values, it indicates that the aircraft is ahead of schedule, time-wise. From this, a simple rule can be extracted: an increase in the difference between executed and planned distances $(\triangle d_l)_i$ is tantamount to a loss of time, therefore a positive delay; whilst a decrease of it implies a gain of time, or in other words, a recovery of delay. An increase or a decrease is certainly better identifiable through the analysis of the derivative of $(\triangle d_l)_i$: a positive (negative) delay-generating event corresponds to positive (respectively negative) peaks (see Fig. 6.8 - bottom panels). Given the commonly noisy behaviour of the derivative - that may lead to false peaks detection -, we considered convenient smoothing the derivative series $\frac{\mathrm{d}(\triangle d_l)_i}{\mathrm{d}i}$ through a short-window moving average.

At the end of the process, the identification of delay-generating events is reduced to the detection of all compact time-windows where the smoothed derivate is above (or below) a pre-defined

[5]Here we have subtracted the planned distance to arrival from the executed one; clearly, reversing the order would only reverse the sign of the results

threshold τ (see horizontal dashed line of Fig. 6.8: two positive and three negative events would be detected) which is, thanks to the constant velocity hypothesis, a time measure.

To summarise, the presented methodology yields three outputs about delay-generating events occurrences.

- A counting of the number of delay-generating events - that is proportional to the number of times the derivative of the distance function intersects the threshold τ.
- The sign of the generated delay is given by the sign of the derivative at that point.
- The magnitude of the delay is given by the area under the derivative function for the time window during which the event is occurring.
- The time stamps corresponding to each event, which also allows pinpointing its exact location.

Before going further, we must discuss some points of the previous methodology that might seem dubious, or cause some bias in the output, starting with our first constant velocity assumption and its possible consequences. More specifically, this assumption allowed us to make a parallelism between distances and time notions, as the methodology transforms distances to destination into delays. However, this hypothesis might be criticised as it is clearly not fulfilled during landing, approach, and departure procedures, when the aircraft is slowing down or accelerating. Furthermore, some might point out that this condition might also be false in en-route situations: (a) different wind conditions (direction or velocity) for planned and real trajectories might engender velocity changes; (b) an Air Traffic Flow Management (ATFM) instruction to avoid conflicts; (c) the flight following a Dynamic Cost Index [DP09] strategy and finally (d) a Required Time of Arrival constraint [WCL01]. Minimal velocity changes are expected for (a) and (b), as deviation from the planned trajectory are usually small (see Fig. 6.7). Also, (c) and (d) are current research topics and yet not implemented. As such, for en-route assessment of delay-generating events, the original assumption of constant velocity is acceptable for now. Whenever changes in velocity will be more commonly used in daily operations, the proposed

methodology would still allow the detection of events generating longer (or shorter) routes, even if it will not be transposable to a time-oriented information. That said, combining it with other time-based metrics as the duration of the flight yield insightful knowledge about the overall efficiency of the system. We can perfectly imagine an inefficient situation where the aircraft flown a higher than planned distance in a shorter overall time, strongly suggesting non-optimal velocities and fuel consumption.

In what follows, we propose several applications for this metric.

Events aggregation

The afore-presented algorithm is parametric-bound, as the choice of the threshold τ must be done beforehand. Its specification might come handy whenever a study of the spatial and temporal distribution of events generating delays of magnitude superior or equal to τ is wanted. Otherwise, it is more interesting to obtain a non-parametric algorithm - and fling off τ. For this, we discretise the pseudo complementary cumulative distribution functions (CCDF) of the delay magnitudes generated by individual events. Separately, positive and negative events CCDF are denoted as:

$$F^{+}(\tau) = P(D \geq \tau) = \int_{\tau}^{+\infty} f_D(t)dt, \tau > 0 \tag{6.13}$$

and:

$$F^{-}(\tau) = P(D \leq \tau) = \int_{-\infty}^{\tau} f_D(t)dt, \tau < 0, \tag{6.14}$$

$f_D(t)$ being a distribution function, *i.e.* the probability of finding a delay-generating event of magnitude t. Therefore $F^{+}(\tau)$ and $F^{-}(\tau)$ are the probability to encounter a delay-generating event which delay generated is of magnitude superior (respectively inferior) to τ. D represents the ensemble of delays considered.

As we are working with discrete data, let us sample τ uniformly in the range $[-m, m]$[6]. Thus:

$$P^+(D = \tau) = F^+(\tau) - \sum_{\tau_i > \tau} p^+(\tau_i), \tau > 0 \tag{6.15}$$

and:

$$P^-(D = \tau) = F^-(\tau) - \sum_{\tau_i < \tau} p^-(\tau_i), \tau < 0. \tag{6.16}$$

$f_D(t)$ has been substituted by $p(\tau_i)$, which is the probability of finding a delay-generating event whose delay is comprised within the i-th bin of the discretisation of $[-m, m]$. More specifically, it can be computed as:

$$p^+(\tau_i) = \frac{\#\text{event}(\tau_i) - \#\text{event}(\tau_{i+1})}{\#Flights}, \tag{6.17}$$

with $\#\text{event}(\tau_i)$ the number of delay-generating events for the threshold τ_i, and $\#\text{event}(\tau_{m+1}) = 0$. Also, $i \in \{\tau_i > 0, \quad \tau_j > \tau_i \quad \forall \quad j < i\}$ - that is, if we subtract to the number of event generating a delay within the i-th to superior bins, the number of events having generated a delay contained from the $(i + 1)$-th bin, then we obtain exactly the number of events that generated a delay which magnitude in contained within the i-th bin. The same rationale is applied to negative delays, with a slight adjustment:

$$p^-(\tau_i) = \frac{\#\text{event}(\tau_i) - \#\text{event}(\tau_{i-1})}{\#Flights}, \tag{6.18}$$

with $i \in \{\tau_i < 0, \quad \tau_j < \tau_i \quad \forall \quad j < i\}$ and $\#\text{event}(\tau_{m-1}) = 0$.

These two distributions can be computed running the proposed algorithm for each of the discrete thresholds within the selected interval. Each one can furthermore be characterised by its first

[6] m has empirically been chosen in order to include every possible event. It has been calculated as the maximum absolute value of the derivative of the distance function between real and planned trajectories from a subset of 1000 flights. Its value here is 0.852

moment - that is, positive and negative distribution are described by

$$\mu^+ = \sum_{\tau>0} \tau * p^+(\tau) \tag{6.19}$$

and

$$\mu^- = -\sum_{\tau<0} \tau * p^-(\tau), \tag{6.20}$$

which respectively represents the total amount of positive and negative delays generated while airborne by the system. These values are intrinsically different from the standard Delay Difference Indicator used in official reports [COD14] as the latter - computed as a simple difference between negative and positive delays - does not convey information about the dynamics of the system. Indeed, no difference is made between a scenario in which DDI is 0.0 as no delay are generated and a scenario of 0.0 because delays are generated and subsequently recovered, the latter case suggesting a resilient system. The computation of μ^+ and μ^- can also be restricted to specific geographic locations and temporal windows to characterise specific behaviours of the system.

Note that both μ^+ and μ^+ are temporal metrics (defined in hours, in what follows) and not distance ones, thanks to the constant velocity assumption. Moreover, being these two metrics extracted from distributions, they do no allow for quantitative interpretations at the individual level.

Delay generation and resilience

The two outputs μ^+ and μ^- can be used to set a rough description of two important aspects of air transport: the total delay generation of the system, and its resilience.

The first one is conceptually very similar to the previously mentioned Delay Difference Indicator - computed as the difference between μ^+ and μ^-. Should the value be positive, this is a sign of a

generation of airborne delays superior to what the system have been able to recover. Reversely, a negative value indicates that, on average, more delays have been recovered than generated en-route. This ability of the system to recover delay leads us to a second important aspect of a system: its resilience (see, Section 2.4.4) - that is its capacity to recover its properties after a disturbance.

In what follows, we adapt the resilience definition saw in Section 2.4.4 to focus specifically on the delay generation properties of the system. We have extracted two metrics allowing for the characterisation of the system's endowment to generate or recover delays. Intuitively, an ATM system would be qualified as resilient whenever it produces, for any amount of positive delay-generating events, as many events of comparable magnitude recovering the overall perturbation. This definition contrasts with the individual perspective adopted for the algorithm design, for we do now extrapolate the results to gain a broader more general information. The definition does not state that a resilient system is able to compensate a delay for the same flight, as otherwise a resilient system would be tantamount to a delay-free one. The shift has been made to pass from an ensemble of flights scrutinised one by one, to an overview of the system as a whole, thus yielding a resilience definition that considers all flights and all delay-generating events, for then comparing the occurrences and magnitude of positive and negative ones. Numerically, it is evaluated as the relation between μ^+ and μ^-: a negative $\log(\mu^-/\mu^+)$ would stand for the failure of the system at compensating positive delays, and a positive value would indicates a resilient system - that is, according to our definition, that more negative than positive occurrences have been generated.

Complementary algorithm for vertical inefficiencies

The previously described analysis has made the strong assumption that aircraft only suffers horizontal (2D) inefficiencies, in the sense that they do not deviate from the planned trajectory in altitude. We propose to complement the previous analysis with an assessment of the vertical profile inefficiencies resulting from ATC mediations. Indeed, an aircraft might have to change its altitude to avoid a crowded sector or, less commonly, to avoid an air collision,

under the instructions of the controllers. Such modifications of the vertical profile deviates from what would be considered an 'optimal' and 'unaltered' ideal profile: climbing-cruising-descent. Therefore, like 2D deviations from plan trajectory, deviations from the ideal vertical profile are consequences of the limitations of the planning model of the system and must be assessed, both temporally and spatially.

Looking at the variations on the real flight vertical profile between each pair of consecutive points, we extract the number of changes in the sign of the variation (which can be -1 if the flight is descending, 1 is ascending and 0 if its altitude is stable). The ideal vertical profile would result in two changes in sign: the first one from 1 to 0 when the ascent is finished, and the second one from 0 to -1 when the descent begins. Flights having only one change ($1 \rightarrow -1$) - corresponding to short trips with no cruising phase - are considered ideal. Therefore, we scan the entire data set looking for flight with more than two changes in their vertical profiles, for which all changes between the first one and the last one are consequences of the inefficiencies of the system. It must be noted that considering the variation as a simple difference between the altitude of two consecutive point might engender some noise, as stabilised points might come with 1 nautical miles (NM) of difference. As such, a threshold on the amplitude of the variation have been set to 2 NMs.

6.2.2 Temporal analysis

It appears that the European ATM system - considered as a whole - can be defined as resilient. Fig. 6.9 (top-left panel) plots the two previously introduced metrics (μ^+ and μ_-) against each other for all the days available in the data set. It can be observed that all points lie above the diagonal (grey dotted line), for the exception of Wednesday 7th and Friday 16th of December. That means that for almost all days of the data set, there have been more negative delay generated than positive ones. But what happened on those two specific days so that the system looses its resilience? Tab. 6.3 reports all reported perturbations in the *Network news* section of the EUROCONTROL Network Operations Portal (NOP).

Indeed, December 16th seems to have been a particularly disturbed day, with the closure of

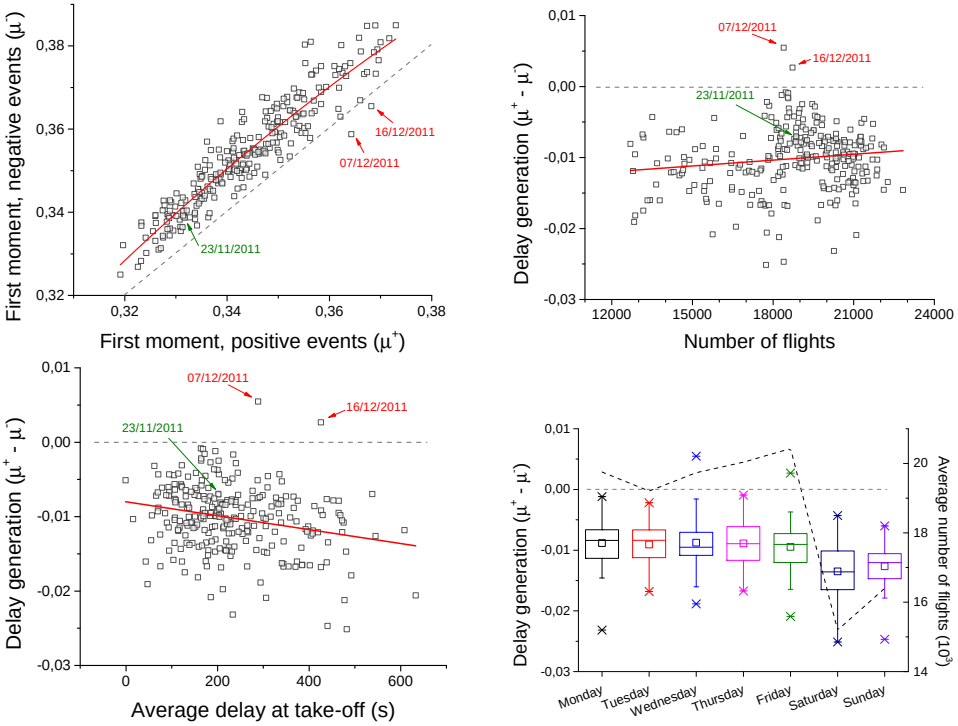

Figure 6.9: Characterisation of the system in the temporal domain. Top left panel represents the relationship between μ^- and μ^+ at each day. The remaining graphs represent the fluctuation of the system's daily response as a function of the number of flights (top right); the average delay at take-off (bottom left); and the day of the day (bottom right). In all, the red solid curves represent the best quadratic fit. The dashed blue line in bottom right panel represents the number of flights as a function of the day. Reprinted with permission from [BPZ16].

two important airports (Manchester and Stockholm) and strong wind reported at big airports (Heathrow, Charles de Gaulle Orly, Brussels, Munich, Rome, etc.). However, the high magnitude of these perturbations, while not frequent, can occur on other days, as reported on the 23rd of November, without affecting the resilience of the system. Also, the perturbations suffered during the 7th of December seem much less important and still affected the resilience of the system, although it touched central airports like Madrid, Heathrow, Oslo or Amsterdam. The resilience of the airborne phase of the system (during a day timeframe) thus appears to be uncoupled with the operational conditions encountered at the airports, though other factors as sectors congestion or closure - more implicated in the direct route of the flights - might be responsible for the non-resilient behaviour of the system on those particular days.

It is quite remarkable that not only most days appear to be resilient, but that - considering a day long of data - the system is able to recover positive delays[7] whatever is the value of such

[7]This does not mean, of course, that airborne delays do not exist, nor that an individual flight cannot be

Date	Airport(s)	Perturbation
23/11/2011	EDDL, EGLC, EGLL, ELLX	Fog
	EBBR, EDDM, EDDT, ESSA	Fog with high delays
	LFPG, LSZH, ENGM, LFOB	Low visibility regulations
	LIRF	CB activity
	LFPB	High delays
	LP**	Portuguese strike action due to commence tonight
7/12/2011	EGLL, EHAM	Strong wind
	ENGM	Reduced capacity due to snow removal
	LEMD	Fog
	EDGG	Regulations applied to staffing and capacity
16/12/2011	EBBR, EDDL, EDDM, EGLL, EGKK, LFPG, LFPO, LFSB, LIRF	Strong wind
	EGGW	Low visibility
	ENGM	Fog and snow
	EGCC	Closed due to snow
	ESSA	Major electrical failure closing the airport
	LTBA	High demand
	GCTS	Global regulations applied
	EDGG	ATC staffing regulations

Table 6.3: List of perturbations for three days of 2011, as reported by the EUROCONTROL Network Operations Portal (NOP). Reprinted with permission from [BPZ16].

delays. In other words, the resilient ability of the system does no seem to saturate, and that is confirmed by the linear behaviour of the quadratic regression included in the top-left panel (red curve). Whilst lacking the data resources to study the cause behind the appearance of these delay generating events and the relationship behind their dynamics (as their proportional relationship leads to diverse question as whether positive events creates negatives ones; or whether the opposite is true, etc.), it can nevertheless be observed that the magnitude of the system's reaction remains partly framed by the traffic and the average take-off delays of that day (see Fig. 6.9 top-right and bottom-left panels). The slight correlations highlighted

delayed: just that generated and recovered delays, on average and on a daily scale, cancel out.

Table 6.4: p-values for the Kolmogorov-Smirnov test, between the distributions of generated delays of the days of the week - see Fig. 6.9 (bottom right panel). Reprinted with permission from [BPZ16].

	Mon.	Tue.	Wed.	Thu.	Fri.	Sat.	Sun.
Mon.	1.0	$9.74 \cdot 10^{-1}$	$4.78 \cdot 10^{-1}$	$9.01 \cdot 10^{-1}$	$7.62 \cdot 10^{-1}$	$4.33 \cdot 10^{-5}$	$1.06 \cdot 10^{-4}$
Tue.	$9.74 \cdot 10^{-1}$	1.0	$8.63 \cdot 10^{-1}$	$9.94 \cdot 10^{-1}$	$7.88 \cdot 10^{-1}$	$9.39 \cdot 10^{-4}$	$3.83 \cdot 10^{-4}$
Wed.	$4.78 \cdot 10^{-1}$	$8.63 \cdot 10^{-1}$	1.0	$7.26 \cdot 10^{-1}$	$7.26 \cdot 10^{-1}$	$1.03 \cdot 10^{-4}$	$3.63 \cdot 10^{-5}$
Thu.	$9.01 \cdot 10^{-1}$	$9.94 \cdot 10^{-1}$	$7.26 \cdot 10^{-1}$	1.0	$9.78 \cdot 10^{-1}$	$1.50 \cdot 10^{-3}$	$1.50 \cdot 10^{-3}$
Fri.	$7.62 \cdot 10^{-1}$	$7.88 \cdot 10^{-1}$	$7.26 \cdot 10^{-1}$	$9.78 \cdot 10^{-1}$	1.0	$3.59 \cdot 10^{-3}$	$1.50 \cdot 10^{-3}$
Sat.	$4.33 \cdot 10^{-5}$	$9.39 \cdot 10^{-4}$	$1.03 \cdot 10^{-4}$	$1.50 \cdot 10^{-3}$	$3.59 \cdot 10^{-3}$	1.0	$4.96 \cdot 10^{-1}$
Sun.	$1.06 \cdot 10^{-4}$	$3.83 \cdot 10^{-4}$	$3.63 \cdot 10^{-5}$	$1.50 \cdot 10^{-3}$	$1.50 \cdot 10^{-3}$	$4.96 \cdot 10^{-1}$	1.0

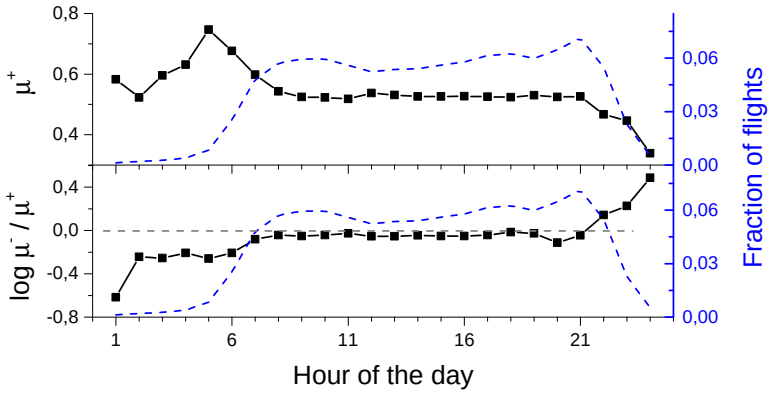

Figure 6.10: Characterisation of the system in the temporal domain as a function of the hour. The dashed blue line represents the fraction of flights as a function of the hour of the day. Reprinted with permission from [BPZ16].

with the linear regressions suggest that less traffic and more take-off delays are coupled with a stronger absorption of airborne delay; this is probably because take-off delays provide incentives to the pilot for asking the controllers for a shortening of the route - which is, in turn, much easier to provide when less traffic (*i.e.* more space in the sky) is observed. Such features remain insufficient to explain the general behaviour of the system's airborne reactions without additional data; however, they bring insights to the understanding of the little fluctuations observable within the magnitude of the system's response (*i.e.* to what is due the difference between μ^- and μ^+). This is furthermore confirmed by the bottom-right panel, where the difference between the positive and negative response of the system have been plotted as a function of the day of the week, and where a Kolmogorov-Smirnov test (see Tab. 6.4) confirms the stronger reaction of the system on weekends - that is when both the traffic and take-off delays are lower.

Finally, Fig. 6.10 resumes the evolution of both the creation of positive delays and the resilience of the system as a function of the hour of the day. It can be appreciated that the resilience is particularly high at the end of the day, while low in the early morning. The blue dashed line specifies the fraction of flights operating as a function of the hour of the day, notably suggesting that the lower proportion of flights during night shall wide up the interval confidence of the high and low resilience reported during the two phases of the night. However, this pattern can also be explained by the fact that flights on their landing and take-off phase are characterised with respectively high and low resilience (see further in Section 6.2.4). Specifically, the high resilience between 9 p.m. and midnight might be due to the higher proportion of landing flights; the low resilience between 1 a.m. and 6 a.m. to the higher proportion of departing flights; and finally, the equilibrium between take-offs and landings during the day generates a stable resilience.

6.2.3 Spatial analysis

The metrics here proposed are adaptive, in the sense that instead of computing them for each bunch of flight operating within the same day, they can characterise the spatial localisation of the events. This is possible as the proposed methodology pinpoints the exact moments in which each delay generating event is occurring. Such moments can be tagged along the trajectory of the flight to identify the airspace region where the event happened. To achieve a better visualisation of the distribution, the entire European airspace has been partitioned into regions of 5°longitude by 5°latitude. Each delay generating event of the data set is then linked to the region where it happened. It must be specified that the spatial position has been recovered by looking at the real position of the aircraft - where the aircraft actually is - at the time of the delay occurrence. Also, as the generation or absorption of a delay can span over a given time, it is possible that two airspace zones end up pinned with an additional delay occurrence, forasmuch as aircraft might change of zones during that time . This process will then allow for the calculation of both μ^+ and μ^- for each and every one of the airspace zones for which more than 50 events were reported. Fig. 6.11 (left panel) represents the subsequent geographical distribution of μ^+ over the European airspace. Grey zones represent discarded airspaces (*i.e.*

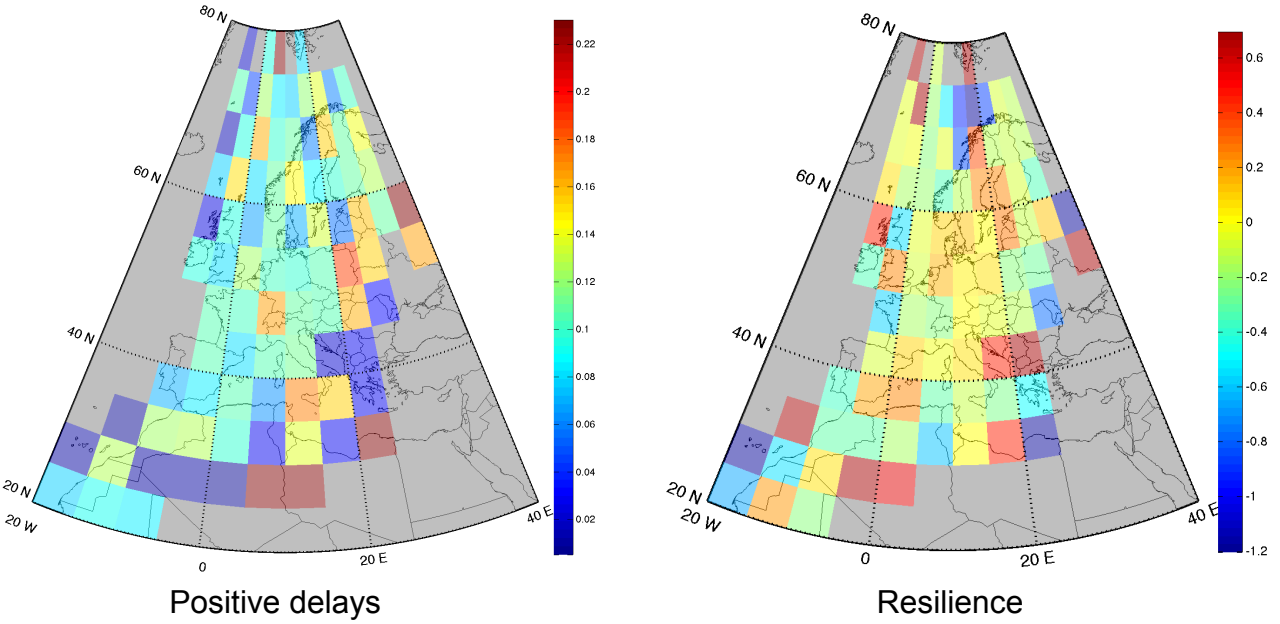

Figure 6.11: Spatial characterisation of the system. (Left) average positive delay generation (μ^+). (Right) average resilience of the system ($\log \mu^-/\mu^+$). Reprinted with permission from [BPZ16].

zones with less than 50 events over the whole 7 months dataset) and blue-to-red colours depict the airspace's tendency to generate low or high amplitude positive delays. The right panel represents the resilience ($\log \mu-/\mu^+$) of each one of these regions, with resilient ones being depicted in yellow to red shades and non-resilient ones in blue to green. The reader must note that comparing such regions between one another is made possible by the independency of the metrics from the number of flights - as they are extracted from a distribution.

It can be appreciated in Fig. 6.11 that central Europe (*i.e.* Switzerland and between Germany and UK) are the airspaces generating the highest delays, which is consistent with Fig. 3.3 where traffic is higher precisely in central Europe. As expected, the delay magnitude is lower over the Balkans, for example. Now, what is interesting is the corresponding resilience of the aforementioned regions. In spite of the superior traffic and the appearance of higher amplitude delays of central Europe, it remains clearly resilient, indicating the ability of such regions to create compensatory negative delays. As opposed to the non-resilience of low-delay generating airspaces as the Balkans (with the exception of the region above continental Greece).

6.2.4 Analysis across the flight phases

The temporal and spatial analysis show where and when the delay events are appearing. Further knowledge can be extracted by adopting an aircraft-centred point of view, aiming at pinpointing the phase of the flight involved in the appearance of delay events. In other words, we propose here to partition a flight into three common phases (departure, en-route and arrival) to assess the propensity of each phase to the generation of delay-related events.

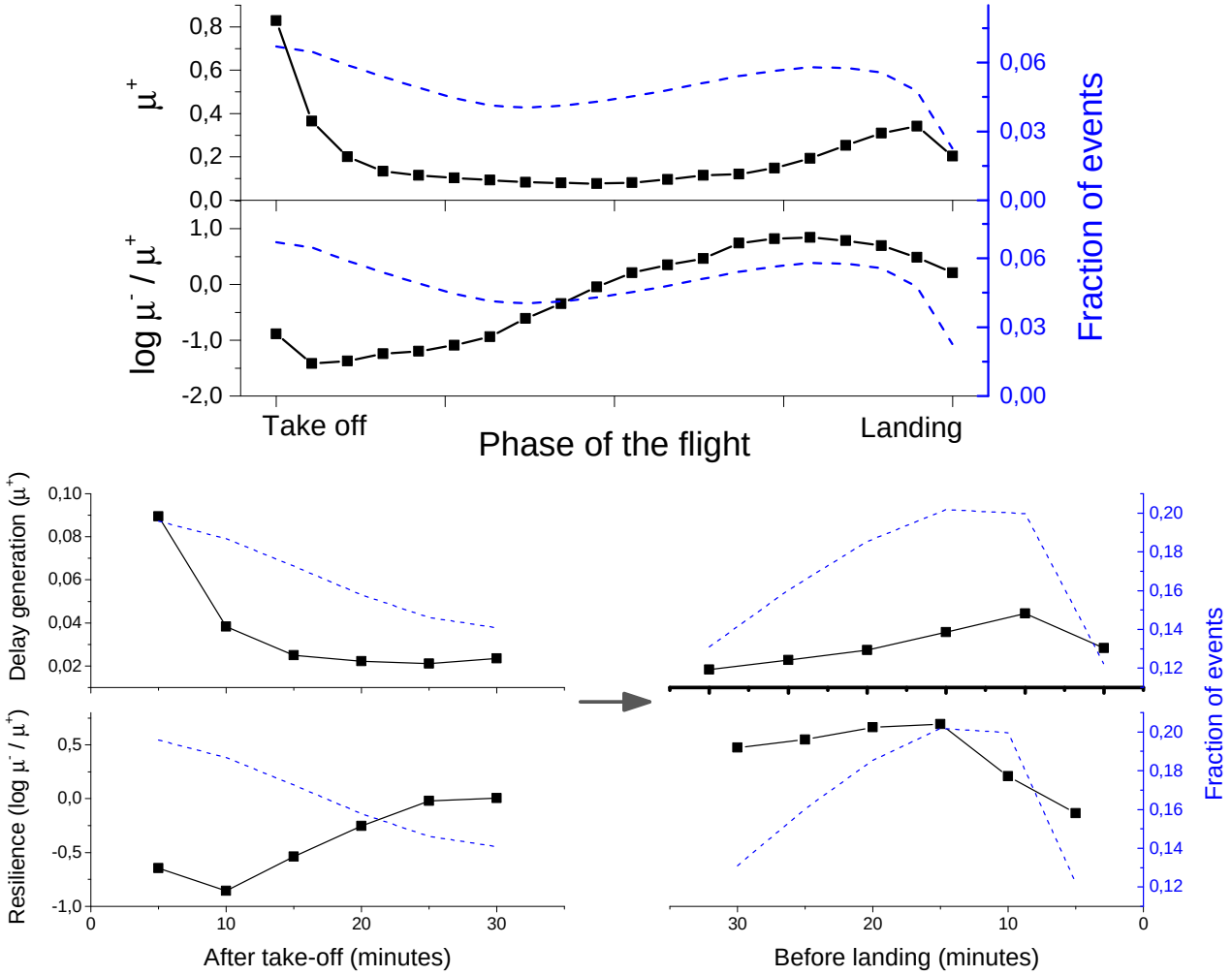

Figure 6.12: Characterisation of the system in different moments of the flight. (Top) Average positive events (μ^+) and average resilience ($\log \mu^-/\mu^+$) along the evolution of flights (solid black lines). (Bottom) Same two metrics for the 30 minutes after take-off and before landing. Dashed blue lines represent the fraction of events generated in each step of the flight. Reprinted with permission from [BPZ16].

We have roughly defined the take-off and landing phases as the first, respectively last, 25% of the flight in term of time. It does not correspond to the actual climbing and descending phase of a flight, as such a proxy might be erroneous for short trips or very long ones. However, as a first

approximation, its shortcomings can be accepted. Fig. 6.12 (upper panel) plots the evolution of the average amount of positive delays (μ^+) and the resilience of the system ($\log \mu^-/\mu^+$) as a function of the defined phases of a flight, highlighting how most positive delays are generated during the first phase of the flight. Moreover, it can be appreciated how the first phase and last phase of a flight have opposite behaviour in term of resilience, the first phase of a flight being less resilient, probably because of the elevated amplitude of the generated delays. The last phase, on the contrary, seems much more resilient.

To eliminated the bias of our first proxy - as flights of different length are being considered - the 30 minute after take-off and prior to landing of a flight has been further considered as the climbing and landing phase. Fig. 6.12 (bottom panel) confirms the previous results: the first phase of a flight is characterised by the appearance of delays of higher amplitude while last phase prior to landing is characterised by its higher resilience.

So far, a rough time approximation has been used to identify the phase of a flight. A scan of the data might allow to extract the real phase of the flight, *i.e.* climbing and descent segments, independently of time. To this end, we scan the radar altitude of the flight, looking for altitude stabilisation in the form of 5 consecutive points at the same altitude (end of climbing phase) and 5 consecutive descending point (beginning of landing phase). The scan is done iteratively as if it fails to encounter the phase changing points, the number of consecutive points looked for is decreased by 1 until the unitary lower bound (which may happen in short trips where no en-route phase will be encountered). For each of these phases, the spatial distribution of μ^+ and $\log \mu^-/\mu^+$ is depicted in Fig. 6.13. It can be appreciated that whilst positive delays are more prone to appear during take-off and landing phases, only the first one corresponds to a non-resilient state. This difference might be caused by the lack of flexibility of departure procedures, as they are both the most easily altered procedures in the name of safety (weather, congestion, etc.) and specifically designed to minimise the acoustic impact of operations in near-town airports [HV08]. If we go back to Fig. 6.10, the higher resilience at the end of the day might be explained in light of the decreasing number of take-offs, more prone to create higher delays.

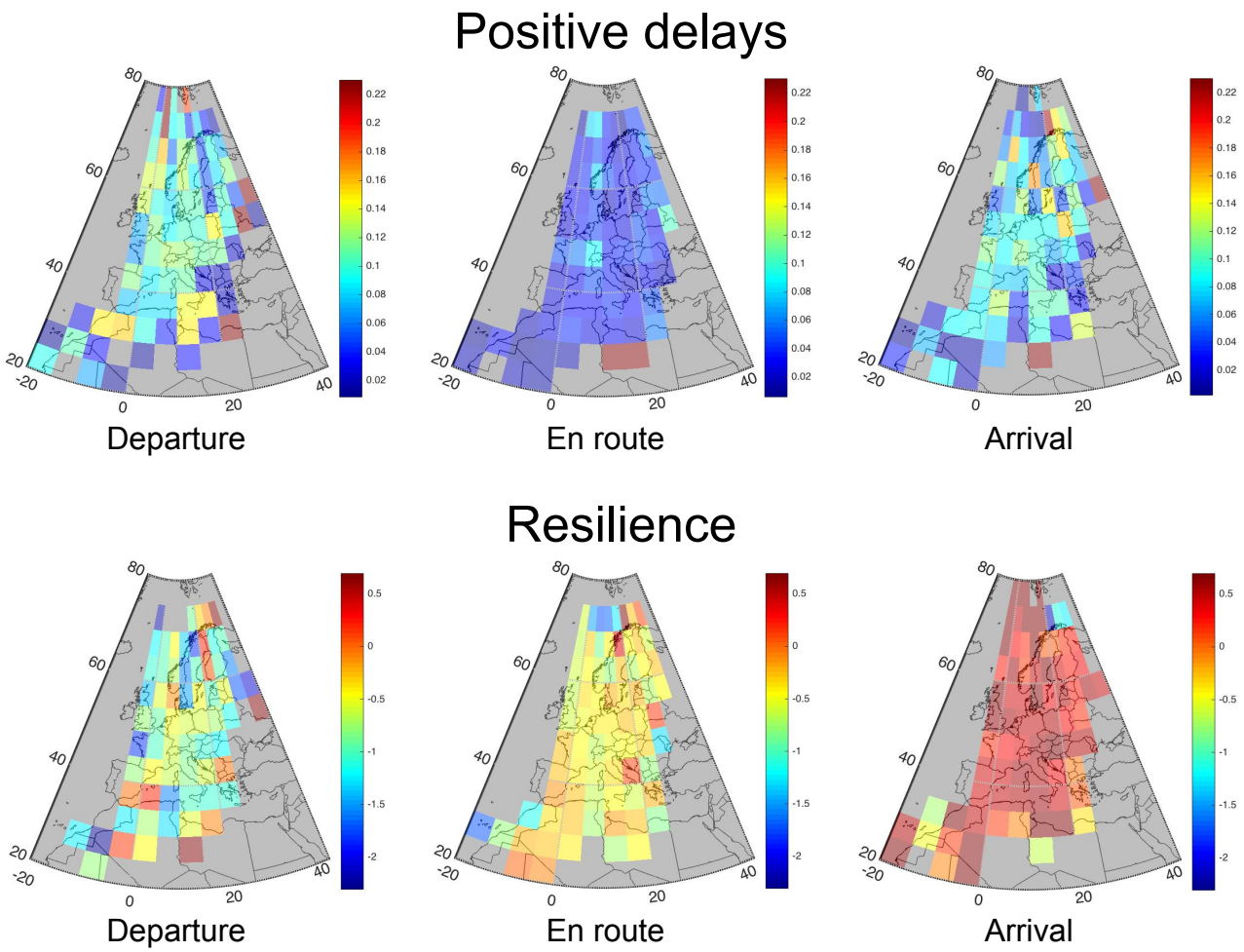

Figure 6.13: Average positive delays and resilience as a function of the phase of the flight. Top and bottom panels respectively represent the spatial distribution of positive delays and of resilience. Left, centre and right panels correspond to the departure, en route and arrival phases, respectively. Reprinted with permission from [BPZ16].

6.2.5 Vertical analysis

A similar study has been conducted for vertical deviations from ideal flight profile. Fig. 6.14 (left panel) shows how vertical deviations[8] - just as horizontal deviations - do not appear uniformly through the European airspace, but are more frequent in some regions; possibly because of the different flight management procedures adopted. Moreover, the right panel suggests a two-phased linear relationship between both kinds of deviations when the time dimension (*i.e.* the day of the week) is taken into account, such that week-end days are more prone to exhibit stronger deviations from planned trajectories, both vertically and horizontally. This behaviour is probably related to the lower level of traffic during weekends, allowing for

[8]When normalised according to the number of flights.

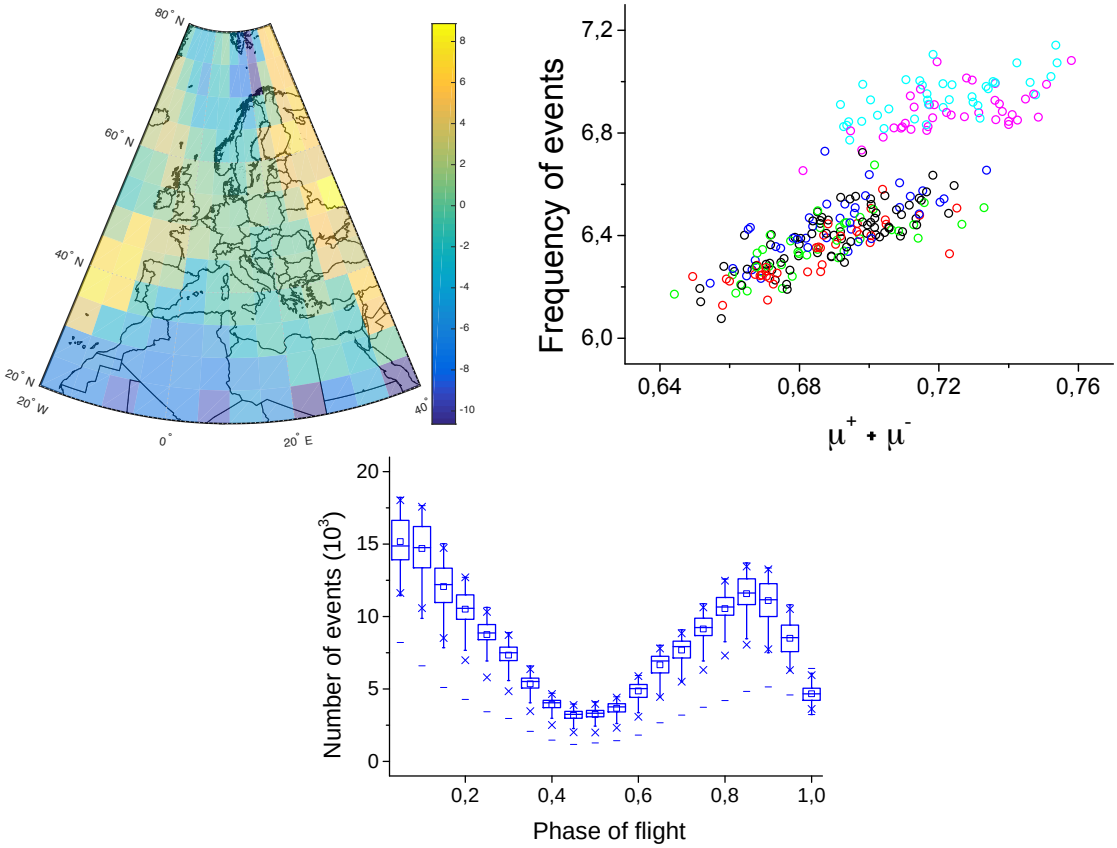

Figure 6.14: From left to right, top to bottom, graphs represent the spatial density of vertical deviations from ideal flight profile (log scale); the relationship between the frequency of vertical deviations with the system's delay generation; and finally the number of events as a function of the phase of the flights.

more vertical adjustments without significant risks. Finally, bottom panel highlights how the departure and the approach procedures are the ones most suffering from deviations, probably because of stair-like ascent and descent procedures.

6.3 Results and Discussion

The classical way to study a complex networks (like air transport system) usually focuses on extracting a network from a system, and then proceeding with the study of its properties. However, such can be done with various levels of precision. In other words, the representation of a complex system depends on the level of detail used in describing its elements or their interactions. Whilst the macroscopic approach used in Chapter 5 was interested in assessing the information flow between airports, it contrasts with this Chapter's microscopic approach,

where only the characterisation of a unique element of the network (an airport) is reached, with no information about external elements. In other words, we varied the scale of observation, focusing our attention more precisely on one atomic element of the network and its behaviour.

First, we criticised the commonly used delay multiplier (DM) metric commonly used to describe the delay-wise behaviour of an airport. Specifically, we highlighted the limitations of DM: as a static global metric, it fails at fully characterising the non-linear dynamic of airport delays' propagation. We therefore proposed a more granular, complementary metric, shedding light on the non-linear behaviour of each airport; thus allowing for example the characterisation of the dynamics of airport under strong stresses. Results showed that airports can be classified into two groups: those whose participation into delay propagation tends to explode with large inbound delay, and those for whom the outbound delay saturates, making them independent of the magnitude of average inbound delays. These non-linear behaviour have been partly related to simple airport characteristics, suggesting that more complex phenomena (*e.g.* local idiosyncrasies or global regulations) are also responsible for an airport's delay propagation behaviour.

Yet the non-linearity of an airport has already been assessed in Chapter 5, where the channels of non-linear propagation have been explicitly identified. However, there are two main differences between the functional network there constructed and the approach adopted in this Chapter. First, the propagation channels connected airports pair-wise, while our proposed methodology considered inbound and outbound delays of an individual airport independently of any other airport. As such, the latter detects the expected (or average) behaviour of an airport: it is possible that an airport, being at the origin of few non-linear propagation arrows (according to Chapter 5 reconstructed network), nevertheless presents a linear average behaviour, as the non-linear channels have been smoothed away by other linear ones. Secondly, the relationships detected in Chapter 5 yield from a black-box model, in the sense that the function driving the non-linearity is not accessible. On the other hand, our proposed methodology results in an interpretable and explicit propagation function. As such, both results are complementary and by nature different. Additionally, the heterogeneous behaviour of airports against high magnitude delays further explains the difference between NNC and EE delay propagation networks:

high delays might be controlled and propagated linearly, depending on the airport internal resources.

This gain of information certainly comes at a cost. The higher precision of the information requires more granular data. When average inbound and outbound delays are sufficient to compute the DM, the metric still needs a large number of time windows in order to ensure of the presence of enough number of extreme operations. Such scenario might not happen in small airports that handles few flights per hour. Also, we have opted for a quadratic fit of the data in order to extract the non-linear behaviour of an airport. Such choice is supported by the easier interpretability of the coefficients (as opposed to a higher polynomial order) and by the upper limit of average inbound delays (as average delays above 30 minutes are extremely infrequent), which yield 30 bins (see the methodology of the metric Section 6.1.1) and a risk of overfitting if more coefficients are introduced into the model.

Secondly, the common usage of the ground program in Europe has made airborne delay studies under-appreciated. We nevertheless pinpointed their importance in term of environmental consequences, for example in light of the future Single European Sky (SES) open-route project. However, the lack of studies about airborne delay generating events might also be partly explained by (a) the lack of coherent and complete report data and (b) the high computational cost of high granularity (*i.e.* such information is only accessible following an aircraft course). We proposed a linear, parallelisable algorithm that compares aircraft's en-route trajectories with their planned ones to extract positive and negative delay generating events. The linear complexity of the algorithm makes its implementation as a parallel processing possible, and therefore the analysis of a very large amount of flight tractable. Would additional data be available (*e.g.* controllers and pilots reports), the proposed algorithm might be adapted to detect specific types of events, like course changes due to adverse weather or traffic congestion. Nevertheless, even with no explanatory factors about the nature of the cause behind the appearance or recovery of delay, this high precision algorithm allows for the extraction of spatial and temporal intelligence about the system that would not be accessible with a macro-level approach. Specifically, the results showed a globally resilient system in its en-route phase, *i.e.* that it is able on average to recover (given a long time window) as much time through rerouting

(or other procedure) than the positive suffered delays. Such resilient behaviour seems to be even enhanced by some favourable situations, like a low numbers of flight (i.e. low sky congestion) or a high delays at take-off (implying a incentive for delay recovery). Additionally, the study yields geographical and temporal information about the system. Of particular relevance, horizontal and vertical deviations (thus delays) do not appear uniformly through the European airspace, but are more frequent in some regions - possibly due to different flight management procedures.

The algorithm might be however sensitive to the resolution of the trajectories. As said before, a higher resolution should not be a problem whenever parallel processing is possible. We have considered a temporal resolution of 2 minutes, which resulted in 4 GB of data per month after the interpolation. While this rather low resolution does not hinder the obtention of meaningful results, an increased one would allow more sensitivity to small changes, like small-deviations due to bursting strong wind. This higher sensitivity can also be perceived as noise, given that the nature of the delay introduced are independent of the pilots' or air traffic controllers' will. The lower resolution therefore allowed for the averaging of such events (over larger time windows), such that results are easier to interpret.

Chapter 7

Conclusions

7.1 Review of the Thesis objectives

In this document we have presented the main results obtained within this PhD Thesis. As demonstrated along the presented work, data mining and complex network techniques can be fruitfully merged together, enabling the improvement of our comprehension of complex systems, as might be the air transport system and more precisely its delay propagation process. The complex networks' capability for synthesising large structures of interactions in simple matrices with no superfluous information naturally blends with the ability of data mining for managing large sets of data. As a result, knowledge can be extracted from complex systems: in this specific case here considered, knowledge can be extracted from delay time series, in order to understand delay propagation, in a way not possible before. Moreover, the presented results show that adopting an *information processing* perspective yields relevant results about the emergent dynamics of a system, when data are available. On the other hand, data mining techniques can also be used to enhance traditional individual analyses, reaching therefore a deeper understanding of the system's behaviour.

The delay propagation problem has here been tackled in tree steps:

Reconstruct the physical structure of the system. While the customary approach in com-

plex systems analysis recommends mapping all the constituting elements into nodes of a network, this presents some important disadvantages. Besides of an increased computational cost, the inclusion of noisy nodes, *i.e.* of nodes codifying irrelevant information (as might be airports with few flights a year), or the mere availability of data may condition any analysis of the resulting network.

As shown in Chapter 4, the topology observed in the air transport system strongly depends on the way the network representation is created. Specifically, it is possible to see that some sampling procedures (*e.g.* selecting only the busiest airports) strongly (and negatively) affect the topology of the representation. Even more, based on a simplistic toy-model, it has been shown how this sampling can even perturb the assessment of the dynamics happening on top of the network, including the characterisation of the delay propagation process.

Network reconstruction through data mining techniques. The reconstruction of network representations of a system is either done considering the presence of some clear physical connections, or, in the case of functional networks, by assessing the abstract exchanges of information between the elements of the system. To that end, algorithms must extract the desired information from the available data, which is then mapped into a network to be studied through the complex networks framework.

To that end, Chapter 5 combined three different causality metrics, to detect some specific interactions between the different elements (*e.g.* airports) of the systems. Specifically, in Section 5.1 we presented two existing causality metrics, allowing the representation of two delay propagation scenarios: a) the propagation links present on average, expected to be observed at any time; and b) the exceptional connections that appear when the system is strongly disrupted. In Section 5.2 we introduced a new data mining algorithm specifically designed to extract non-linear causation from time series, therefore allowing to complete the characterisation of the propagation process looking at its more non-linear (thus less controllable) connections.

Use data mining to improve transportation studies. While the relevance of a macro-

scopic perspective have been tackled in Chapters 4 and 5, the network representations clearly disregard the specific dynamics of the elementary items of the system. Specifically, it is the information flow within the system, considering all the elements, that matters. We complemented this insightful approach by a more traditional one. Chapter 6 focuses on the two main elements of air transport systems: airports and airplanes.

As such, the incoming and outgoing flow of airports are studied independently of the others; and individual flights are followed along their trajectories. Specifically, Section 6.1 introduced a new metric, characterising and quantifying the general non-linear behaviour of individual airports; and Section 6.2 compared the planned and executed trajectory of flights to extract delay-generating events, whose temporal and spatial distributions yield valuable information about the strengths and shortcomings of the air traffic management system.

In conclusion, the novel contributions of this PhD thesis to the existing Literature are:

- A first quantitative comparison of the European, US and Chinese networks, including the description of both their structural and dynamical properties.
- An ad-hoc methodology to optimise the sampling strategy for a network without harming its topology and dynamics, or in other words, its *representativeness* of the real system.
- A new algorithm for the detection of non-linear causality relationships between two time series, especially designed to highlight the non-linear delay propagation channels of the system.
- A methodology aiming at complementing the poor information used to characterise airports role in delay propagation.
- A methodology able to detect delay-generating events, both temporally and spatially.

7.2 Future lines of research

The preliminary questions have been, all considered, successfully answered. However, as it is usually the case with any type of problem, its solution (or the mere path used to reach the solution) sheds light on other potential problems. We here list some issues that we trust need to be tackled in the future, along with some improvements of the presented solutions:

Improve sampling strategy. In order to mitigate the proven effect of a sampling strategy on a network representation, we have proposed a method in Section 4.2.4 based on the stabilisation of the degree distribution entropy, which represents an optimal strategy to safely eliminate nodes without loosing the representativeness of the network. However, such methodology presents the severe drawback of being *a posteriory*, in the sense that the final value of E_{dd} must be known beforehand, while such scenario might not always be the case. Thus the need for the development of a new technique mitigating the topological bias resulting from sampling nodes, in the line of some recently proposed algorithms [RC17].

Delay propagation mitigation strategies. The results presented in Section 5.1 suggest that normal and extreme perturbations propagate with different dynamics, which, in turn, imply that different strategies must be used for their mitigation. Specifically, several questions might be asked: what are the economical consequences of both propagations? Is it more efficient to search for resources or procedures aiming at reducing the amount of normal delays, abnormal ones, or to increase the threshold of the system?

Along with these questions, other aspects of the study might be further improved. First, we have mentioned that the system does not switch entirely from one phase to the other, but that links of the extreme event network might substitute some expected propagation channels; therefore yielding a whole distribution of possible combinations. The study of the range of possibilities and their respective frequencies may lead to additional knowledge about the characteristics of the system. Additionally, such characterisation might be complemented by the extracted critical additional delay for an airport to change its

propagation pattern, that is tantamount to the assessment of the resilience of the system.

Causality network pruning. Causality metrics have the difficult task to segregate simple correlations from real causation. The Granger's metric has been proven well-able to successfully distinguish between both. However, it can still be fooled when a third element controls the dynamics of two elements with different lags τ_1 and τ_1. Therefore, the two subsequent elements will appear to have a causality relationships with a lag of $\tau_1 - \tau_2$. Such inefficiency engenders the appearance of artefacts in the functional networks representing delay propagation that has been reconstructed through the use of Granger's causality test. It is interesting to investigate how such artefacts can be detected and eliminated, to have a even more real representation of the delay dynamics.

Enhance microscopic analysis. Beyond what has been presented, the methodology for studying en-route delays on a micro-scale perspective opens new doors towards the understanding and measurement of where and how delays are generated and absorbed. The fusion of such results with an event report database would yield valuable knowledge to the community. While this requires for data that are hard to obtain, a properly performed study might answer some additional issues. For example, the temporal relationships between positive and negative delays might be studied: are negative delays triggering positive ones? Are they independent? A deeper characterisation of these micro-delays will allow for more applied decisions, as the creation of a procedure aimed at improving the efficiency of air traffic. Also, applying the analysis to the US open-route airspace, where airborne delay are expected to already play a more central node into the delay propagation dynamics of the system, would be very insightful.

Appendices

Appendix A

Neural Network Causality

A.1 NNC metric performance

In this section we present some results about the behaviour and capabilities of the Neural Network Causality (NNC) model, by comparing its results with those of a standard Granger Causality (GC) - implemented through the Python StatsModels library [SP10].

A.1.1 Linear relationships

As a first example, we here consider two time series X and Y, with the former causing linearly the latter; this example is of relevance, as the linear relationship is within the capabilities of the Granger Causality. In a mathematical form, the two time series are defined as follows:

$$x_t = \epsilon_t \tag{A.1}$$

$$y_t = \alpha x_{t-1} + sin(t) + \epsilon_t, \tag{A.2}$$

ϵ representing random numbers drawn from a normal distribution and α the coupling constant.

Results for several analyses are synthesised in Fig.A.1.

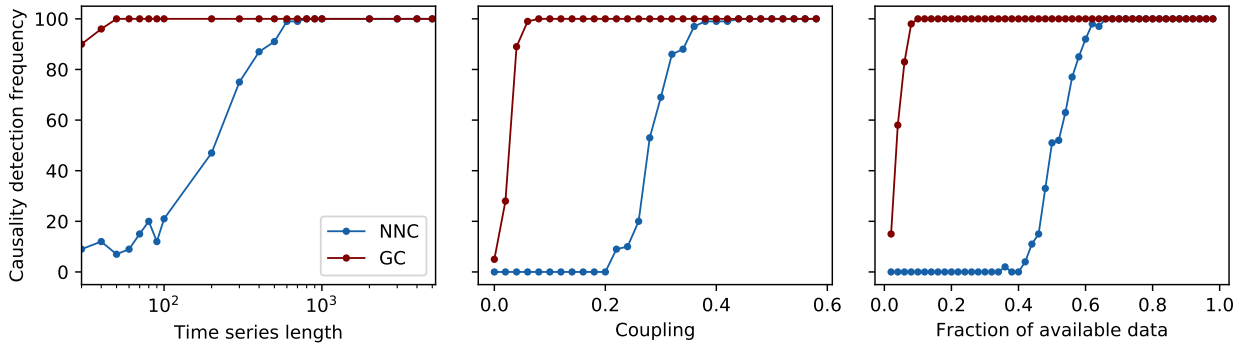

Figure A.1: Evolution of the causality detection rate in function of the time series length (left), the strength of the causal relationship (center) and the number of deactivated causal points (right) for both GC and NNC tests in case of a linear coupling.

We firstly analyse the dependency of the results on the time series length for both metrics - see Fig. A.1 (left panel), for a constant coupling of $\alpha = 0.8$. Each point corresponds to the percentage of times a statistically significance relationship is detected (significance level of 0.05), for 500 independent realisations. First observation is the near perfect behaviour of GC as it seldom misses the causal relation even with low number of points, which contrasts with the bad handling NNC makes of time series with less than 1000 points. Above that umbral, NNC is also able to spot 100% of the causal relations. That brings forward the first important requirement of NNC metric: large enough time series are needed for the neural network to extract efficiently intelligence from data.

Secondly, it is interesting to analyse the influence of the coupling parameter α. Fig. A.1 (middle panel) depicts the behaviour of both causality tests for time series of sufficient length (5000 points) as a function of the coupling strength. The NNC test presents a sharp transition, with a reliable detection of causality only when the coupling exceeds a value of 0.3. GC remains however more accurate with low coupling as the proportion of detected causal relationships tops at 100% for a coupling of only 0.06. Nevertheless, it can be appreciated how NNC metric seems to be as reliable as GC for intermediate and high coupling values.

Finally, real-world time series are seldom perfect, real causation might be activated and deactivated under specific conditions, hence transcribed in data only in a sporadic way. We therefore analyse the behaviour of both causality tests in the presence of missing causal values,

encoded in the causal time series as zeros[1]. Fig. A.1 (right panel) depicts the percentage of correctly detected causalities, for a coupling of $\alpha = 0.8$, as a function of the proportion of non-zero values present in X. Results indicate that the NNC test behaves well, being able to correctly spot causality relationships with just $30 - 40\%$ of the data. Here again, it seems that NNC is being out-performed by the Granger Causality that is able to handle such situations when only 10% of the data.

Even if NNC seems to be out-performed by GC for the linear case, it is important to highlight two determining facts. First, note in Fig. A.1 (middle panel) how GC starts (for a coupling of 0 - that represents the absence of causation) at 5% detection rate while NNC rightfully starts at 0%. For the significance level of both tests being set to 0.05, it would be normal to expect a 5% error. However it seems that NNC either rejects causation completely ($\alpha < 0.2$) or seldom misses it ($\alpha > 0.3$). This makes NNC almost categorical - that is that either it detects a causation or not, leaving few situations where the detection is random (*i.e.* the transition between the two phases, here between 0.2 and 0.3). Secondly, it must be specified that NNC is based on the training of various Neural Networks (the main model and the control ones). Such training is controlled by various internal parameters. The results presented in Fig. A.1 have been indeed obtained by the parameters implemented by default in the library. However, those results depends on how the algorithm is tuned. Section A.3.3 will approach how NNC detection rate is affected by some parameter changes.

A.1.2 Non-linear relationships

Let us now consider the case in which the causal relation between X and Y is non-linear; as the GC test is based on linear models, the GC should only be able to detect the linear part of the relationship, thus potentially leading to an underestimation of the causation. In mathematical terms, we here consider the following relationship:

[1]The baseline, *i.e.* here, the sinusoidal function $sin(t)$ and the random noise ϵ, are not set to zero, only x_{t-1} is.

$$x_t = \epsilon_t \tag{A.3}$$

$$y_t = \alpha g(x_{t-1}) + sin(t) + \epsilon_t, \tag{A.4}$$

where ϵ represents random numbers drawn from a normal distribution $\mathcal{N}(0, 1)$, and g a non-linear function. More specifically, the latter is the sinusoidal function (sin).

The main results are represented in Fig.A.2. Blue and red lines respectively representing the NNC and GC behaviours. It is quite remarkable how the conclusions subsequent to the linear case are now inverted. Whilst NNC still needs more than 1000 points to detect a 0.8 coupling non-linear relationship, it appears that GC is now struggling to detect the causation even in long time series (Fig. A.2, left panel). This is further confirmed in Fig. A.2 (middle and right panels) that represents the causal detection frequency of both metrics in function of, respectively, the strength of the relationship and the proportion of de-activated causal points. The previously observed characteristics of NNC are still present: it does not detect true negatives when the coupling is set to 0; neither it returns false positives when the coupling surpasses 0.3; the transition between no-detection and detection is sharp (from alpha between 0.2 to 0.3) making its output almost categorical; and finally it is able to detect causation when more than 40% of the causal data is provided. On the other hand, GC is struggling with the non-linearity task as it slowly increases its accuracy from the initial error-due 5% to a 95% detection frequency

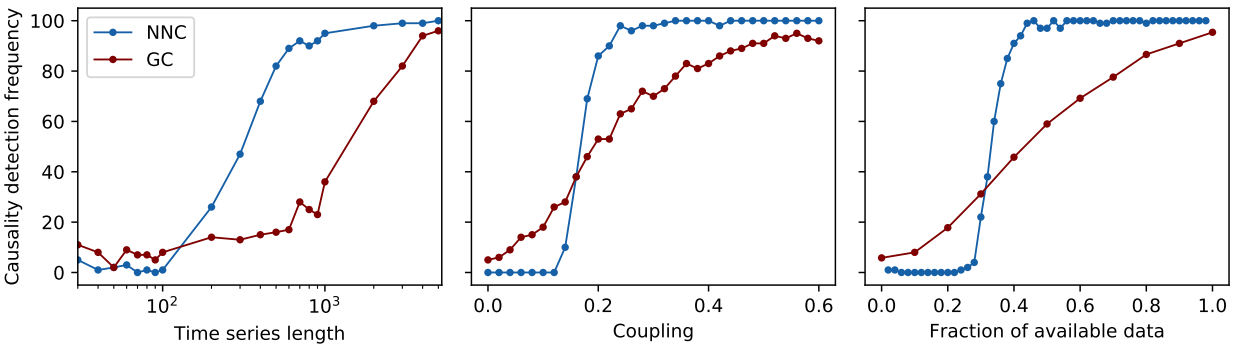

Figure A.2: Evolution of the causality detection rate in function of the time series length (left), the strength of the causal relationship (center) and the number of deactivated causal points (right) for both GC and NNC tests in case of a simple non-linear coupling.

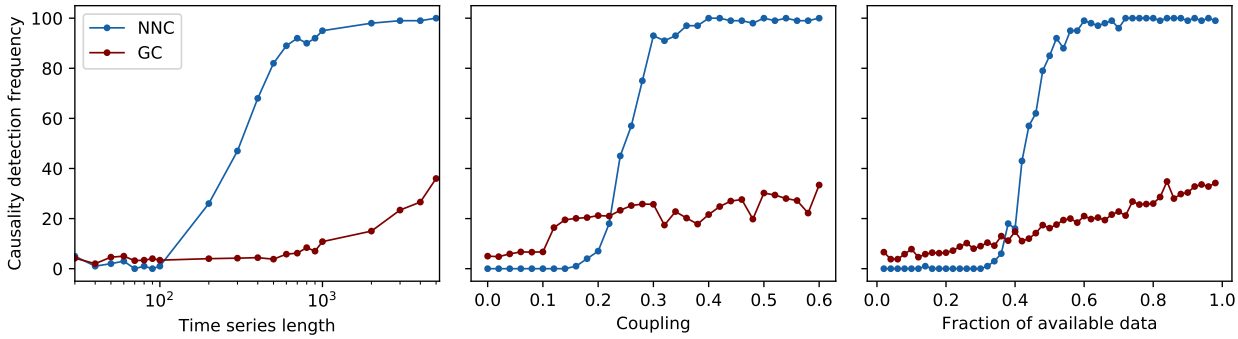

Figure A.3: Evolution of the causality detection rate in function of the time series length (left), the strength of the causal relationship (center) and the number of deactivated causal points (right) for both GC and NNC tests in case of a complex non-linear coupling.

for high couplings[2] ($\alpha > 0.6$). Note that contrarily to the linear case, GC do not reach a full detection rate 100%, precisely because of statistical fluctuations. Its ability to detect causality with only 10% of the data in the linear case has also vanished.

Yet, the differences between both metrics might appear not significant to the reader as GC is still able to detect a sinusoidal causal relationship with high enough couplings. Let us therefore change the non-linear function g in Eq. A.4. Specifically, let us consider its cubic form:

$$g = sin^3 \tag{A.5}$$

As presented in Fig.A.3, for complex enough functions, GC is unable to detect the causation even for high coupling values, while the behaviour of NNC is, in substance, similar to the one presented in Fig. A.2. Indeed, GC tops a 50% detection rate. NNC, on the other side, reaches the 100% accuracy with the same input size requisite (*i.e.* time series length) needed to spot a linear causal relation. Specifically, Fig. A.2 shows that the coupling must be slightly stronger (0.3, *vs.* 0.2 of the sinusoidal case) for it to fully detect causality. On the negative side, we observe that the transition towards a full detection is slower (*i.e.* the slope of the curve is smaller) when compared with the sinusoidal case. That result can be improved tuning the parameters of the NNC function and we encourage the reader to look at Section A.3.3 for more

[2]Note that this is possible because sinusoidal function can be approximated by a linear function when the variable values are low.

details.

A.1.3 Computational time

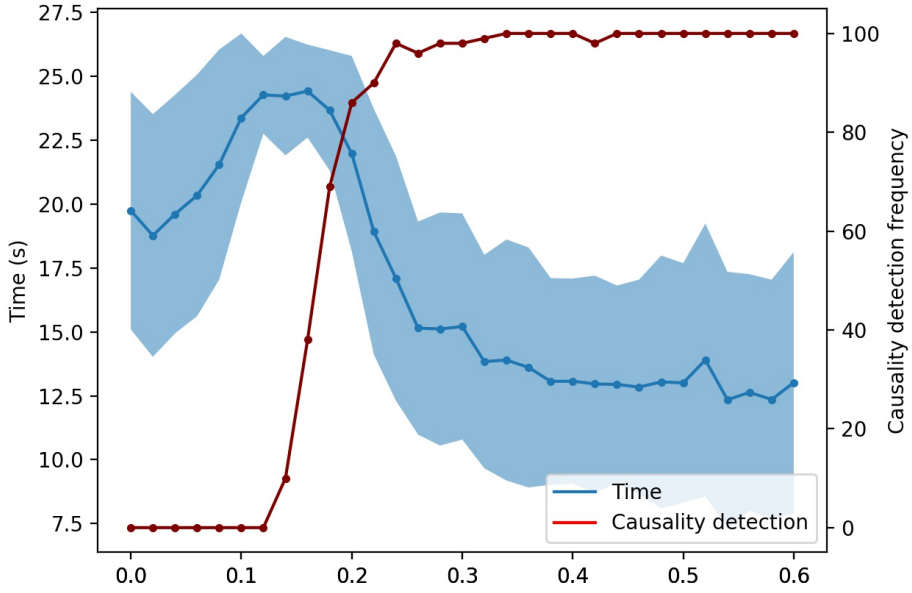

Figure A.4: NNC computational cost.

The non-linear capabilities of NNC certainly comes with a price, in terms of a higher computational cost. In this Section we are going to discuss this aspect. The reader should nevertheless note that the results here presented completely depend on the configuration of the computer - including available memory, CPU cores and speed, *etc.* As such, what is here presented should be understood as the qualitative, generic, time-wise behaviour of the algorithm.

Fig. A.4 depicts the behaviour of NNC when the coupling varies from 0 to 0.6 (red curve), for the sinusoidal relationship presented in Fig. A.2, along with the respective computational time (blue curve) and its standard deviation (blue area). NNC is clearly faster when the causality is strong. On the other hand, it is significantly slower for a coupling between 0.1 and 0.2, which corresponds to the transition window; in other words, when the algorithm doubts, it takes more time to reach a conclusion.

This is further analysed in Fig. A.5: for six coupling constants, from 0 to 0.5, we plot the histogram of the number of training loops used by the algorithm. It can be observed that at 0.1 or 0.2, within the transition window, the algorithm tends to use all allowed loops. When

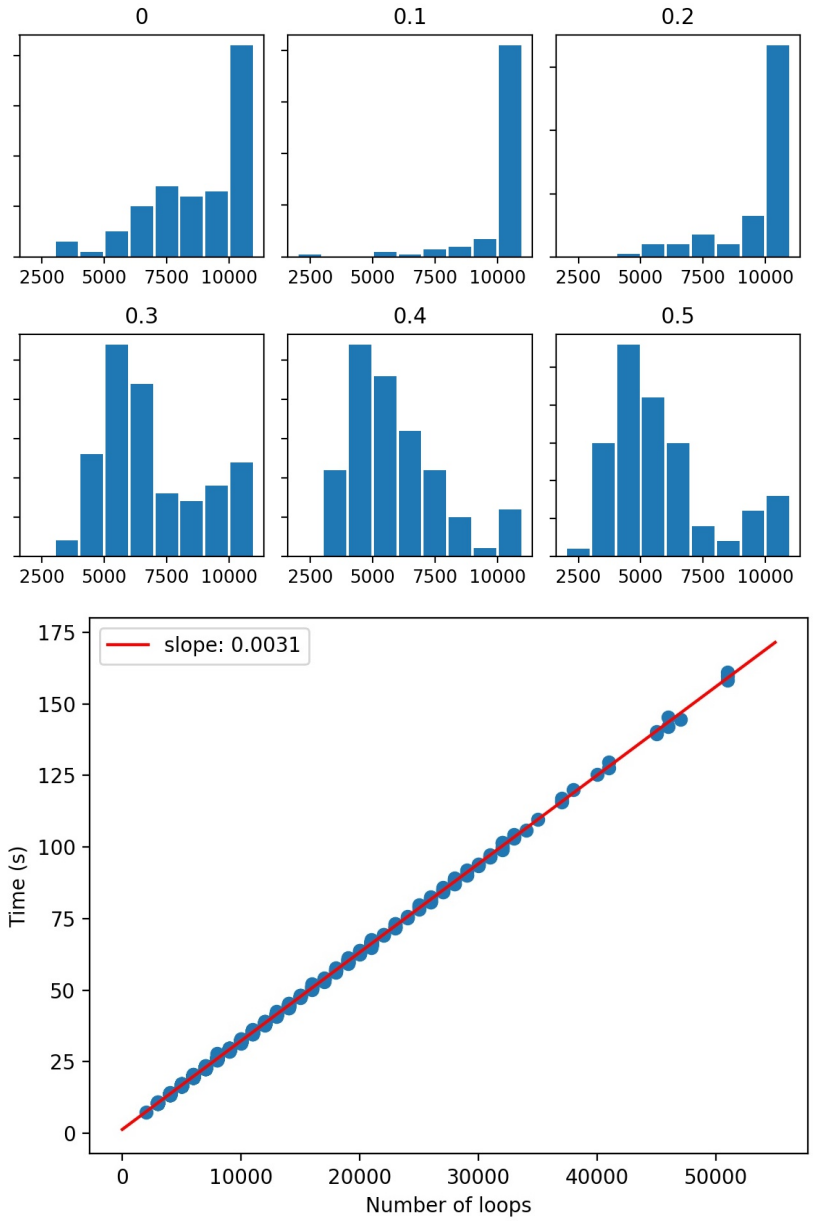

Figure A.5: (Top) Histograms presenting the distribution of the number of epochs used by the test for a non-linear causal relationships of coupling value ranging from 0 to 0.5. (Bottom) a linear fit presenting the relation between the number of epochs effectuated by the algorithm and how long it lasts to compute.

the coupling is at 0.0, such that there is no causality, it is sometimes able to cut the learning process earlier, as it is clear that more iterations do not lead to better results[3]. Finally, when the causality is stronger (above 0.3), it appears that the histogram shifts to the left, as the algorithms needs less and less loops - *i.e.* a relationship is readily detected. In synthesis, the more the algorithms doubts, the more loops are needed to reach a stable solution, thus the

[3]Such behaviour is subsequent to the stochastic nature of Neural Network as some times the convergence is quicker than others.

more time it takes[4].

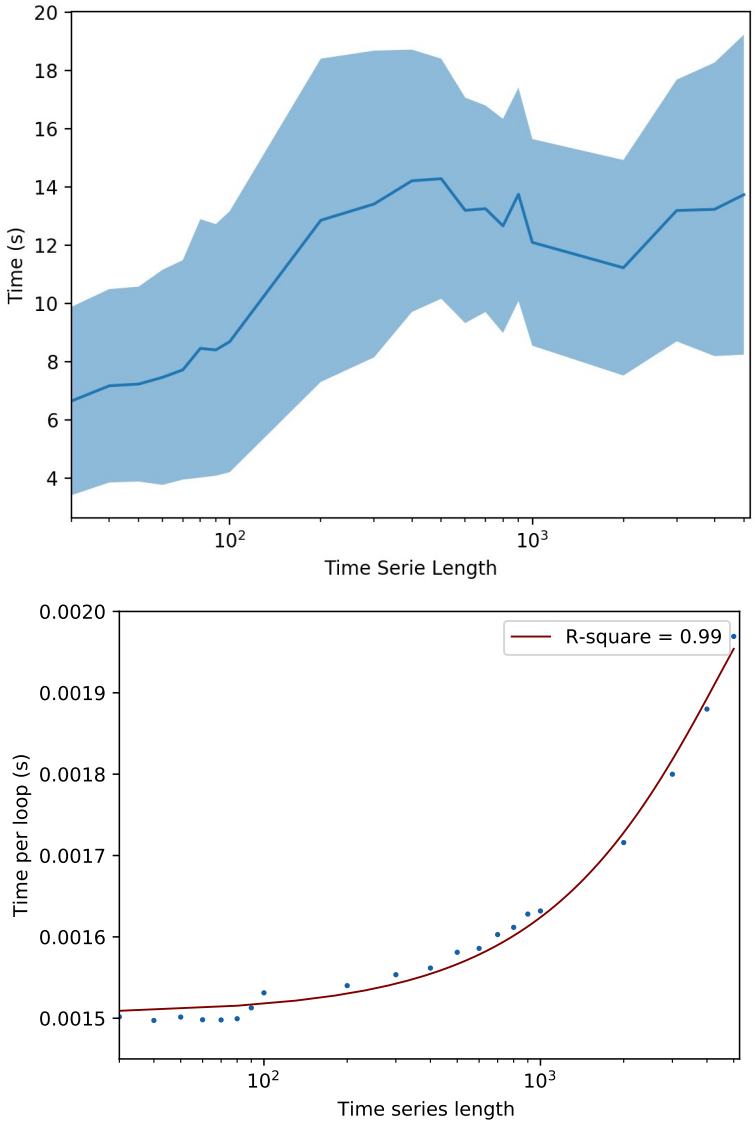

Figure A.6: (Top) Evolution of the mean computational time (and its standard deviation in shaded blue area) of the algorithm for a fixed non-linear causal relationships of coupling 0.8 as a function of the time series length. (Bottom) A quadratic fit of the duration of an epoch in function of the time serie length.

The computational time of the algorithm is then linearly related to the number of loops used by the algorithm, as shown in Fig. A.5 (bottom). It is also relevant to observe that the time serie length has little effect on the computational time, as depicted in Fig. A.6: the computation time relates to the number of training loops through a non-linear equation[5].

[4]The user might wonder why the algorithm would still use high number of loops in a case of strong coupling. The answer lies with the offset detection. As the algorithm also returns what is the shift in time between X and Y that drives the causality relation, it sometimes needs supplementary loops to efficiently derive it.

[5]The full equation is $-7e-12.L^2+1e-7.L+1.e-3$, with L being the time serie length and an R-squared of $3e-9$

A.2 Installation

NNC requires Python 3.3+. The module is freely available on Github and requires the additional installation of TensorFlow Python API r1.0. [6]

TensorFlow can be installed either from the provided binary packages at `http://www.tensorflow.org` or from the Github source. The algorithm also depends on the following Python standard libraries:

- Scipy 0.18.1 (to perform statistical tests)
- Numpy 1.11.2 (to handle the time series)
- statsmodels 0.6.1 (to create the lag time series)
- sklearn 0.17.1 (to preprocess data)

A.3 Using the library

A.3.1 An initial example

The library consist in a unique command-line to facilitate its use. Basically, for X and Y respectively the causal and effect arrays, the command would be:

```
In[0]: import NNC
In[1]: NNC.NNC(X,Y)
```

An example[7] of the output of such command is provided here below:

```
Neural Network Causality
p-value = 1.55361836297e-28
offset = 1.0
```

[6] In principle, NNC would be compatible with Python 2.7 and the related TensorFlow version after slight changes in the code.

[7] The number provided are the result of one random test performed and is of no relevance here.

However, the aforementioned command-line only specified the tested time series therefore leaving all the internal parameter of the NNC function at their default values.

A.3.2 The learning parameters

```
NNC(x, y, maxlag = 1, test_size = 0.2, num_epoch = 2000, num_batch = [], incr
    ↪ = 1000, max_epoch = 10000, h = 10, learn_rate = 1e-3, learning_stop = 0,
    ↪  pv_stop=0.01, false_stop = 0.2, verbose = False)
```

As illustrated by the previous line of code, the NNC function has several parameters that can be tuned accordingly to the problem faced by the user. Let us review them.

X: X must be a numpy array representing the "causing" time serie. It is of size $(n_{obs}, 1)$, where n_{obs} is the number of observations or instances available. X will be tested to understand whether it is causing the time serie Y.

Y: The second numpy array, representing the "effect" time serie, must be of size $(n_{obs}, 1)$ too.

maxlag: The test will be calculated for all lags up to *maxlag*. If not given, *maxlag* will be equal to 1 (default). Its value also indicates the number of corresponding control models created to identify if one of the shifted version of X causes Y.

test_size: Float number between 0 and 1 that specifies the portion of the dataset used for testing the Neural Network models. The remained (1 - *test_size*) is used for training.

level Precise the significance level of the test.

num_epoch: Specifies the initial number of epochs, *i.e.* the number of iteration of the training. For more information refer to Section A.3.3.

num_batch: Specifies the size of the random shuffled batches (*i.e.* samples) of the training data that are used at each epoch to train the model. If not specified, *num_batch* is set to the tenth of the test dataset size.

incr: Defines the number of additional epochs (iterations) when convergence is not reached yet. For more information refer to Section A.3.3.

max_epoch: Specifies the maximum number of epochs or iterations authorised without convergence. Reaching such point will stop the algorithm. For more information refer to Section A.3.3.

h: Defines the number of neurons of the neural network's hidden layer.

learn_rate: Floating point defining the learning rate that is fed to the AdamOptimiser gradient descend algorithm.

learning_stop: Floating number between 0 and 1 describing when the training can stop based on the gain of information of the main model. The gain of information is inversely proportional to the evolution of the main model's error of prediction (see Section A.3.3 for the definition of the gain). Thus, a *learning_stop* of 0 will stop the algorithm at the absence of learning (default value). A higher value of *learning_stop* (let us denote it $\triangle$) would trigger a stop when the error decreases of less than $\triangle\%$.

pv_stop: Floating number describing the significance level of the p-values the test comparing the main and control models must reach to trigger an early stop of the algorithm. That is - even if the main model is still learning - when one and only one of the control models is suggesting a strong enough causation (*i.e.* a p-value inferior to *pv_stop*).

false_stop: Floating number describing the minimum gain of information of the control models to allow an early stop of the algorithm. If an early stop of the algorithm has been triggered by *pv_stop*, the gain of information of the corresponding control model must be lower than *false_stop* to avoid stopping the algorithm too early, that is before the algorithm converges. It is set to 0.2 by default.

verbose: Boolean triggering information printing during the execution of the algorithm.

A.3.3 Controlling the convergence

In the previous subsection, we have seen that various parameters of the NNC function are linked with the convergence of the Neural Network model. It is important for the user to understand how the NNC manages the convergence of the models, as this allows a better parameter calibration.

The NNC algorithm has to train $maxlag + 1$ models (one main model with X value, and $maxlag$ control models with the shuffled lagged versions of X). The different requisites to stop the training are: (a) no more gain of information is obtained (or that the gain is less than specified by *learning_stop* parameter); (b) that the maximum number of authorised loops (max_epoch) has been met; or (c) that a causality has been encountred (*pv_stop*) with a gain of information of the corresponding control model lower than *false_stop*. To explain each of these requisites, we must first define the notion of gain of information.

Information gain

NNC uses an iterative training, in the sense that it starts training the algorithms (*num_epoch* loops) to increment the training by *incr* loops if none of the three stopping requisites has been met.

At each training loop of a Neural Network, the "proximity" of the model's output to the real time serie is assessed using a sample of the data; in other words, at each loop, the model's accuracy is tested on a randomly picked subsample of the test dataset defined by its length *num_batch*. As such, some noise is introduced in the validation, as two consecutive accuracy assessments are computed over distinct subsets and the comparison might be biased [8]. To ensure an unbiased comparison between the performances of the models, its average accuracy over *num_batch* consecutive loops is considered - allowing a better cover of the whole test dataset. Fig. A.7 illustrates the process of the information gain calculation. The accuracy of a model is assessed through the prediction error on a random subset of the test dataset. As such,

[8] Note that this may be solved by considering a fixed evaluation subset; nevertheless, if such subset contains unusual data (*e.g.* artefacts, errors, *etc.*, the whole validation procedure may yield wrong results.

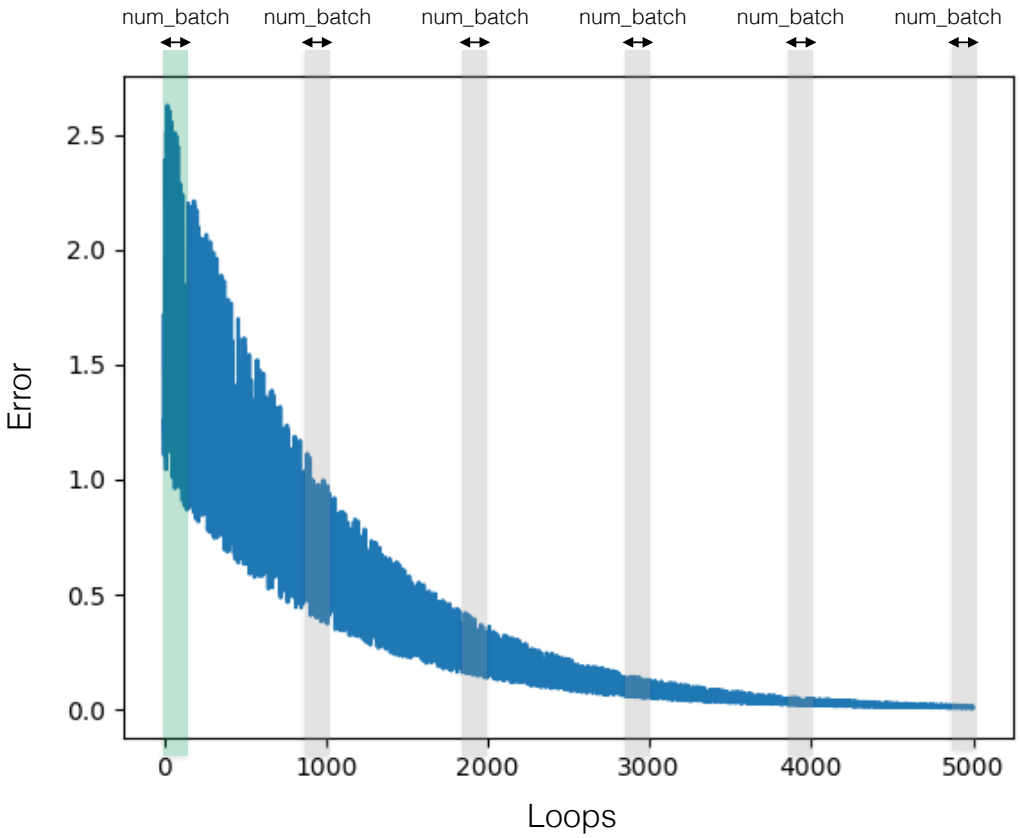

Figure A.7: Neural Network gain extraction.

the algorithm computes an initial average error e_0 as the average error of the first *num_batch* training loop (depicted in green); and the average error at the end of each iteration (*i.e. incr* loops). Those e_i are depicted in grey in Fig. A.7 and considers the *num_batch* loops anterior to each increment step (*incr* parameter). The gain g_i is then simply:

$$g_i = \frac{e_i}{e_{i-1}}, \qquad i > 0 \tag{A.6}$$

Convergence

As we said, the algorithm is programmed to train iteratively:

1. Train the $maxlag + 1$ models with *num_epoch* loops. If $g_1 < 1 - learning\_stop$ then:

2. Continue the $maxlag + 1$ models' training for *incr* loops:

(a) If requisite (a) or (c) is met then stop. That is, respectively if the gain of the main model is still inferior to our threshold or if a causality is detected.

(b) Otherwise, return to step 2.

3. If the total number of iterations reaches *max_epoch* (requisite (b)). In these situations, it may be useful to increase the *max_epoch* parameter, in order to allow the algorithm to reach convergence.

Three different scenarios are then possible (corresponding to requisites a, b and c) and are respectively represented in Fig. A.8. (Left panel) First case: *max_epoch* is not reached but the gain reaches $1 - learning\_stop$ - meaning that no more significant information is extracted from the data; therefore the algorithm stops and yields an absence of causality[9]. In the second case (Middle panel) the model is still learning but *max_epoch* has been set too low. In that case, increasing *max_epoch* is recommended, although there is no guarantee of the final outcome, as the algorithm can either reject the causality (like in the first scenario) or behave like the third case (right panel). In the latter case, a causality is detected after some loops (inferior to *max_epoch*) and with the gain of one of the control model inferior to $1 - false\_stop$.

Setting a high *max_epoch* would not hinder on scenarios (a) and (c). Specifically, if a causality is detected or rejected, the algorithm will stop as soon as possible. On the other hand, its usefulness in scenario)b) depends entirely on the dataset. A higher *max_epoch* might sharpen the transition between non-detection to detection, as pictured in Fig. A.9 (Top panel), or just increase the computation time.

[9]Note that it is possible because main and control models are trained simultaneously.

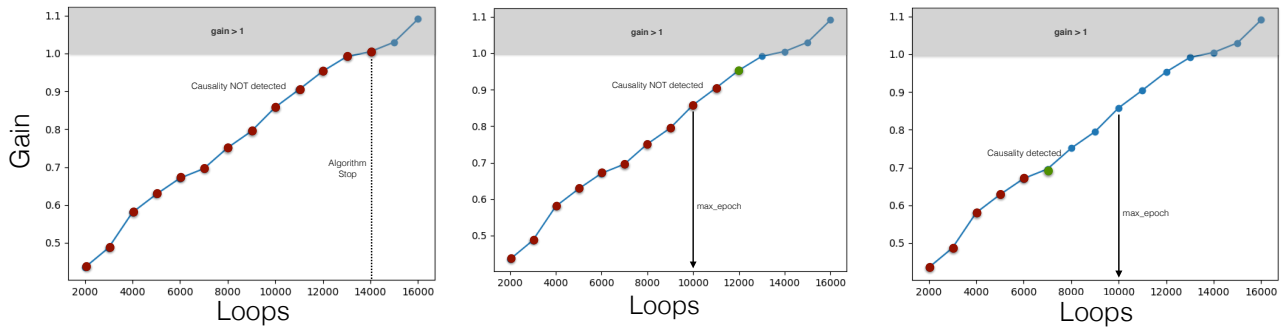

Figure A.8: Possible NNC scenarios.

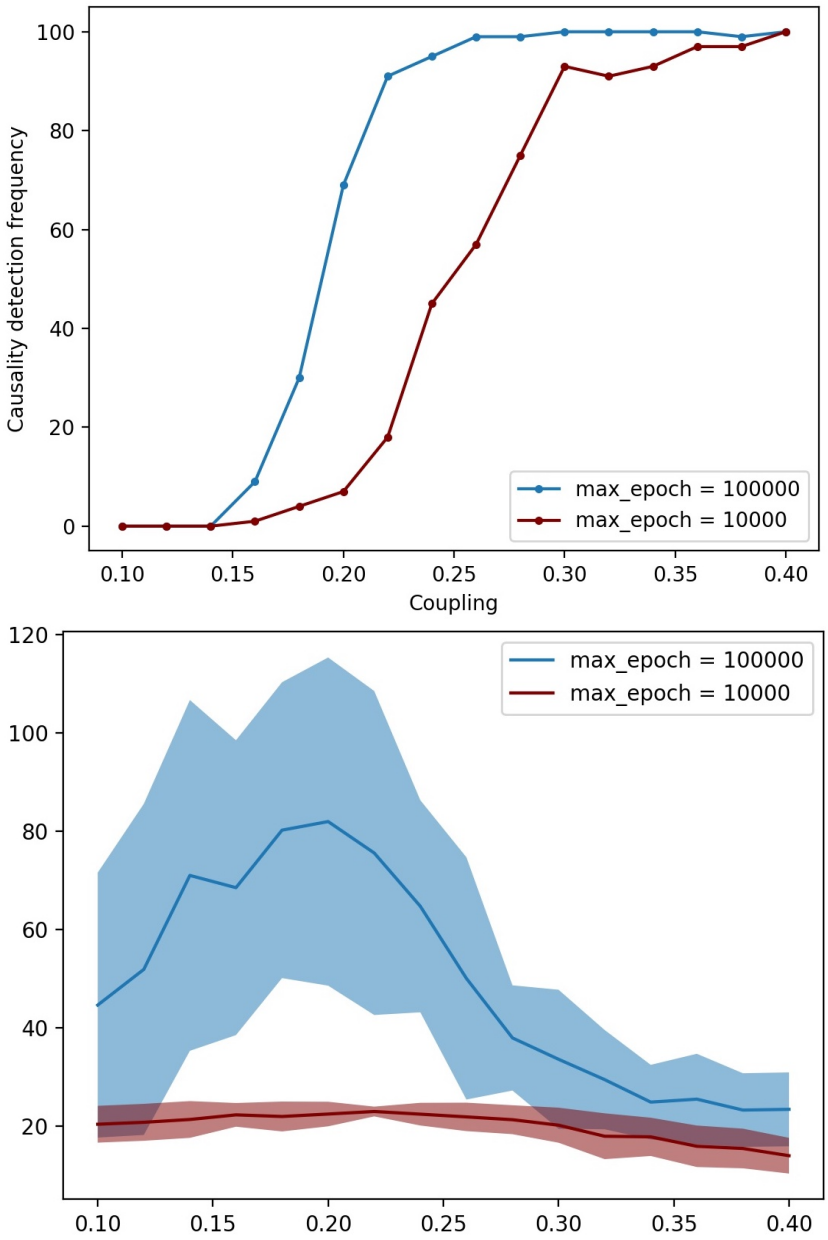

Figure A.9: (Top) Evolution of the causality detection rate for a non-linear causality relationships of coupling 0.8 and (Bottom) the evolution of the computation time; as a function of the maximum number of epochs authorised.

Fig. A.9 shows that a higher *max_epoch* can allow sharper results as it let more time for the learning - that is, for reaching the convergence. This is illustrated by the fact that a non-linear causality (spotted in Fig. A.3 at 0.3) is detected earlier (0.2) when the parameter tuning the maximum number of permitted epochs is increased.

As may be expected, this comes at a certain computational cost (see Fig. A.9, bottom panel): while the time needed by the algorithm does not change much when the coupling is higher (that

is, when the algorithm is sure of the results and the convergence is easier), important differences can be observed when the coupling is low and the algorithm struggles to learn. In the latter case, the algorithm tries to learn during more time, but still does not catch the causality.

Tuning parameters

As it have been listed, various parameters are susceptible to alter the output of NNC algorithm. This section focuses on providing qualitative description of their effect in order to guide future users in their task. Some of them are directly linked with the convergence of the ANN models (*e.g.* h and *learning_rate*). In this section, we will rather focus on parameters affecting the decision making of the NNC algorithm - except for *max_epoch* parameter which case have already been approached in past section.

First, the *learning_stop* is set to 0.0. This means that the algorithm does not stop until extracting the last small piece of information from data. There is no common rule to choose this parameter mostly because of the stochastic nature of ANNs - the learning can be slow as likely as it can be fast. Choosing a larger value is up to the reader as the information gained above a certain threshold might be considered negligible and therefore trigger an earlier stop of the algorithm. A too high value for this parameter might trigger erroneous results as the training is truncated.

Second, the *pv_stop* algorithm allows for the early detection of causal relations. At each increment of the training, the predictions of the main model with the control ones are compared yielding a p-value for each of the lags tested. If a unique control model (representing the prediction with a lagged-and-shuffled X) is statistically significant (inferior to *pv_stop*), then there is a unique lag which shuffling seems to significantly destroy the information contained in the time serie, therefore suggesting causation.

Finally, let us consider the *false_stop* parameter which is related to the previously mentioned *pv_stop*. Indeed, an early stop triggered by a unique control model satisfying the *pv_stop* requirement might only be subsequent to the stochastic nature of ANN (*i.e.* that the conver-

gence is slower, for some reason). Therefore, to ensure that the algorithm does not jump into a erroneous conclusion, it is wise to ensure that the control model is converging or at least, learning less than *false_stop*% at each increment training step.

To tuning of these parameter is not automatic. We encourage the reader to look at the loss function (see Section A.3.4 for more detail) to monitor the behaviour of the training in function of the data and adapt the parameter.

A.3.4 Output

The algorithm returns several information, listed below:

min_p Returns the p-value of the test.

offset Returns the shift between X and Y driving the causal relation, if $min\_p < level$.

epoch Returns the number of epochs used for the training. If $epoch = max\_epoch$ then it might be recommendable to higher *max_epoch* parameter to allow for convergence. A too low epoch mean for an early stop of the algorithm and depends on the presence of causality or not, and on the parameters controlling the decision making process of the algorithm.

time Returns the time elapsed.

loss Returns an array representing the evolution of the loss function of the main model. Its analysis might yield valuable information about the behaviour of the Neural Networks. Whether the convergence is slow or quick might encourage the user to tune the algorithm for better results.

loss_shuffle Returns a list of arrays representing the evolution of the loss function of the control models.

Appendix B

Fostering interpretability of data mining models through data perturbation

In this Annex, we will present an additional work that has not been yet applied to ATM for lack of compatible data - which explain its imcorporation as Annex and not in the main corp of the PhD Thesis. Let us start with the rationale behind the work, then proceed to clarify the value it could bring to diverse ATM situations.

The scope of this work started with the will to use deep learning based on complex non-linear models - or what we called non explanatory algorithms (see Section 2.1.2) - in ATM studies. As we said, delay propagation process is a non-linear process and can probably benefit from a more complex data mining approach. However, the lack of interpretability of the former have strongly limited its usage. The interpretability and comprehensibility of deep learning algorithms has been the focus of many recent studies [Fre14, GSB+16]. Yet, the proper definition of what is and how to measure the interpretability of a model is still not clear [BF16]. [Lip16] highlights two main questions that characterise a model: (a) how the model works? Which would be related to a transparency property of the model; and more interestedly (b) what additional information can be extracted from the model? Which is directly linked to post-hoc

interpretability of the model. The purpose of the latter can be easily understood considering a medicinal example. Physicians often needs to combine various analysis inputs (*e.g.* x-ray, biopsies, MRI, etc.) in order to assert the diagnosis [UAB+10]. To help them in this labor is the project of a widespread Medical Diagnosis Decision Support system (MDDS) which will basically be a large and complete medical database on top of which are run deep learning algorithms to automatically generate diagnosis [Mil94, Ber07, MMG14]. However, from a scientific and medical point of view, the rationale of the algorithm might be as important as the proper treatment suggested. In case of discrepancy between both the clinician and the MDDS-based treatments, a reasoned disambiguation of the support tool might help to settle the conflict. Also, one can see the benefits that such improvement would have on training, as young trainee might be tested on different situation to be compared to the system [CFT97, YVPN08, SHH09]. A similar problem can be imagined in an ATM-centered system where the detection of some safety properties might be linked to complex and variate variables.

As mentioned in Section 2.1.2, white-box models like decision trees are not acceptable solutions as their interpretability comes at the cost of accuracy (which can result in live losses in the cases of medicine or ATM safety). Therefore the need to foster interpretability from black-box models like ANNs or SVMs in a model-agnostic way [RSG16b, RSG16a]. Our idea consists in introduce perturbations in a trained black-box model to extract post-hoc intelligence from its behaviour. The assessment of the variations that trigger a change in the output classification contain implicitly interpretability knowledge about the rules of the models. Not only this method allows to specify the reasons behind a classification in particular (as opposed to the common average feature importance extracted from black-box models), but it is also customisable as the extraction of the smallest changes (or the minimum number of features involved) to swap classes can be implemented. The purpose of the latter can be understood when simpler changes (lower number of variations) are preferred over precise ones (smaller variations) in order to optimise a cost function. Our methodology present the significant advantage of being independent of the used black-box model.

This methodology have been tested on two different dataset: one describing breast cancer records and the other about wine characteristics. We remember here that such methodology

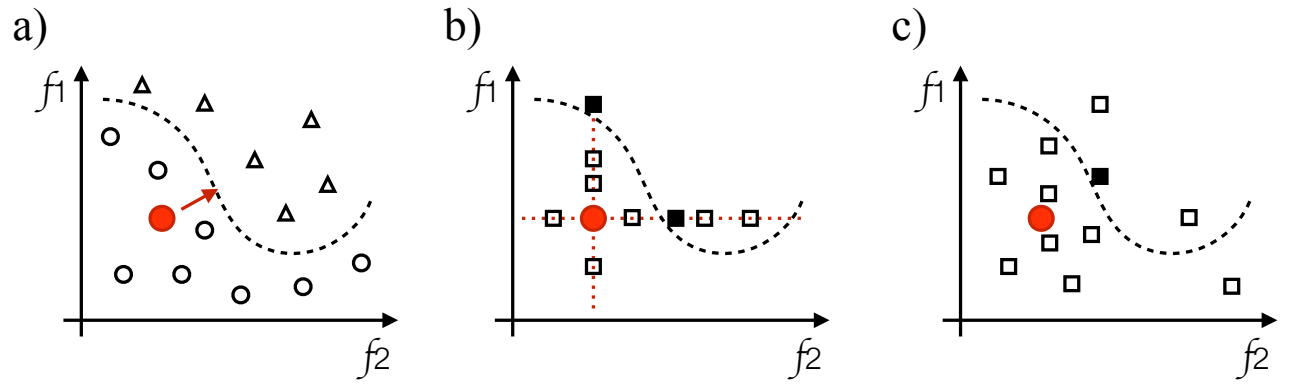

Figure B.1: Graphical representation of the process for creating rationales from a black-box classification model. Panel *a*) depicts the initial situation, while *b*) and *c*) respectively represent the search in one and two dimensions. See main text for details.

have not been implemented in an ATM context. However, we pursue the goal of accessing more complex and complete data that would allow its application. An idea might be the monitoring of an aircraft characteristics to understand what changes should be done to avoid risky unstable approaches.

B.1 Creating rationales from classification models

Without loosing its generality, the problem can be narrowed down two a two dimensional classification task. Instances are then described by two features denoted f_1 and f_2, and the space of possible variations is limited to a plane. Fig. B.1 (Left panel) schematises the initial problem where the a new instance (red circle) have been classified by a black-box model in the "circle" category. The previously trained black-box model has generated an invisible (in the sense that it is implicit and not accessible) delimitation of the space between both categories. The aim of the methodology is to explicit locally this delimitation. For that we aim for the smallest distance (*i.e.* the minimal variation) necessary for the new instance to switch its forecasted label to the "triangles" class (red arrow pointing toward the delimitation frontier). The information would therefore contain information about a local portion of this frontier and help understand a piece of the black-box algorithm's unknown rationale.

The first intuitive and non optimal solution would be exhaustive scanning of the space. That

is abruptly create a large set of *virtual* instances (*i.e.* not resulting from an actual observation, but rather synthetically created for the purpose of the method) uniformly spaced all over the feature space, to then study their forecasted class. This would allow for the description of the whole frontier function. However, such brute force approach becomes intractable when more dimensions are considered. Indeed, the computational cost of this first solution is in the order of $O(n^d c)$, n being the sampling resolution for each dimension, d the number of dimensions, and c the cost of performing one single classification. Furthermore, such method guarantees the extraction of the whole frontier, but not the precise changes necessary for our red circle to change of label. In our scenario, any many others, it is more interesting to have the least number of changes, even of higher magnitudes, to maintain the interpretability.

So, we propose a more adapted solution than the complete sampling of the feature space consists in an progressive increase of the number of independent features allowed to changes (denoted n_f) until finding the least dimensional solution. The process steps are the following:

1. Initially set n_f to 1.
2. Create a vector C with the $\binom{d}{n_f}$ possible combinations of n_f elements out of d - d being the dimension of the problem. Note that, when $n_f = 1$, C consists is merely the list of all the features in the problem.
3. Generate n_p new virtual instances for every feature combination c in C. The new point is created by replacing in the target observation (see red dot in Fig. B.1) the value of the features contained in c by randomly drawn values from the empirical distribution observed for such features within the original (training) data set. When $n_f = 1$ the task basically consists in randomly changing one feature at the time, and save each result as a new virtual instance.
4. Sort the ensemble of generated virtual instances according to their distance to the target instance.
5. Sequentially apply the classification model until one of the virtual instances falls into the desired class. If no forecasted class complies, increment n_f and go back to step (2).

Fig. B.1 middle and right panels schematise the afore-explained process. Specifically, the process starts with setting n_f to 1 and modify only one feature at a time (see middle panel; squares represent the set of virtual created instances.). Two potential solutions (black squares) are found along the two axis. The algorithm would have stopped at the closest one, then saved the corresponding virtual instance. If no solutions would have been found, n_f would have been incremented to 2 (right panel) where virtual elements are sampled from the complete $f_1 - f_2$ plane space. Again, the closest solution (black square) would have been saved by the algorithm.

It is important to specify three customisable aspects of the process:

a The priority is given to low dimensional solution following Occam's Razor principle. High dimensional solutions are only explored when no simpler solution has been found. However, such propriety might be customised by the user of the system in order to save multiple feasible solutions for each set of modified set of feature (resulting in a Pareto front), or prioritising large dimensional changes. The former situation might be of interest in our MDDS example where the system might propose a different treatment in both cases where the patient had a different age, or the same age and a different physical condition. Therefore, the saving of various possible solutions might help improving the understanding of the rationale of the system.

b The sampling strategy in step (3) have been designed to avoid creating virtual points in irrelevant, sparsely populated regions of the feature space therefore optimising the search. Here again the strategy might be adapted to suit more complex situation (*e.g.* Simulated Annealing, Genetic Algorithm, etc.).

c A simple Euclidean distance have been computed between the virtual points and the target instance to perform the sorting of step (4). This definition of the distance suggest an uniform importance of the features. However, some situation might call for a weighting scheme where all feature importance differs in function of a given characteristic. For example, some feature might be difficult to change (or very costly) and therefore the user might want to minimise its contribution to the best solution by affecting it with a low weight.

This approach, whilst still computationally intensive, is still substantially quicker than the first intuitive brute force analysis. The resulting complexity is dominated by the maximum dimension explored, therefore scales as $O(n_f^{\max} \cdot n_p \cdot c)$.

B.2 Case study: breast cancer analysis

Our first application example shows how the proposed methodology complement medical breast cancer diagnosis when the task consists in classifying the cancer as malign or benign. We applied it to the publicly available [WBC16] Breast Cancer Wisconsin Data Set (described in [SWM93, MSW95]) which resumed the digitalised image of a fine needle aspirate (FNA) [WB04] of the breast mass into ten cell nucleus features for 569 patients: radius, texture, perimeter, area, smoothness, compactness, concavity, number of concave points, symmetry, and fractal dimension. Each features is characterised by the average value over all cell nucleus, its standard error and its average over the three largest values, yielding a total of 30 features per patient.

The classification have been performed using the previously introduced (Section 2.1.3) Random Forest and ANN algorithms respectively implemented through the Scikit-learn Python library [PVG+11] and Google's TensorFlow API [AAB+16, RG16]. ANN model was composed of one hidden layer with three neurons and was trained for 1000 epochs. Random Forest has been implemented with 1000 estimators and a minimum number of samples in each split of 10. The parametrisation is not relevant as the objective here is not to reach the better classification, but to illustrate the capabilities of the methodology. However, we report in Tab. B.1 the classification score obtained with and without cross-validation (see Section 2.1.4) along with

Classification model	**Accuracy No CV**	**Accuracy CV**	**Avg. training time (sec.)**
Random Forest	99.29%	94.31%	1.54
ANN	97.89%	94.83%	2.15

Table B.1: Classification scores and average training time for the breast cancer data set, for the two considered classification algorithms.

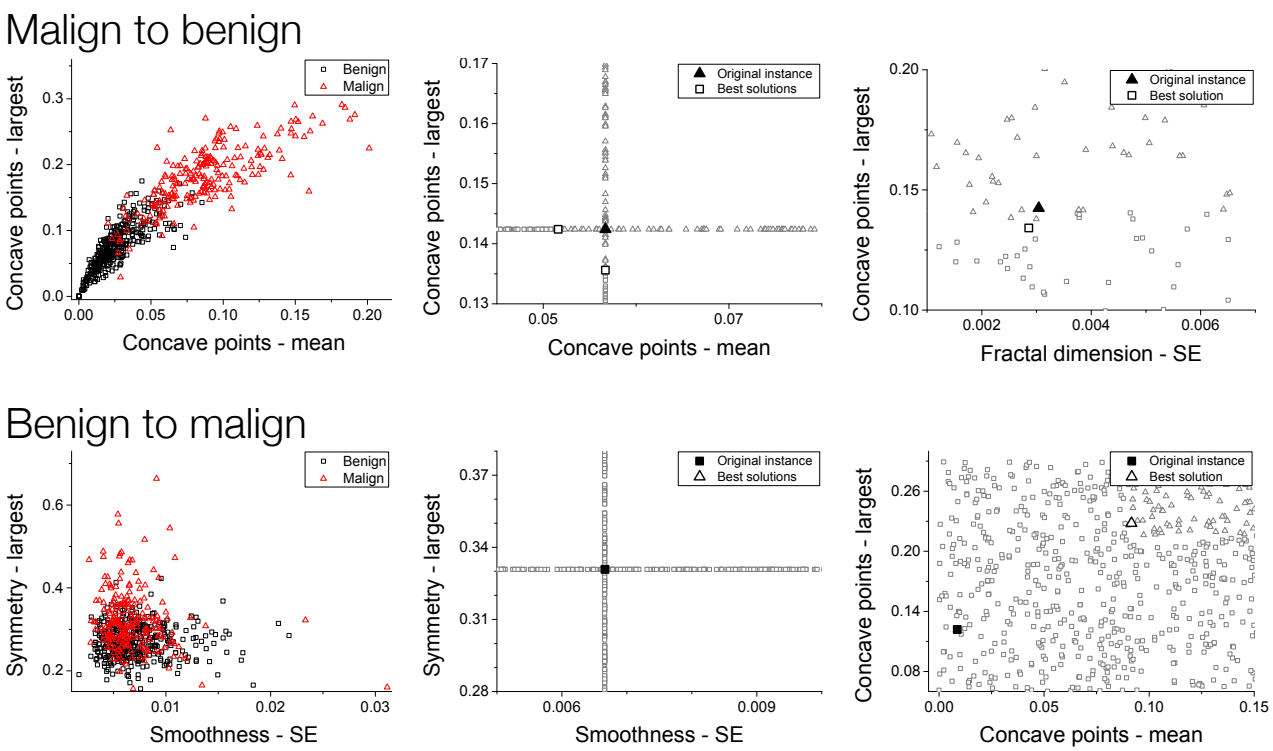

Figure B.2: Graphical representation of the proposed methodology applied to the breast cancer data set. Upper panels correspond to a malign cancer, lower panels to a benign one. Left, centre and right panels respectively depict the original data set, the search by varying one single feature, and the full search in the plane. See main text for details.

the training time. We remember that the methodology is agnostic as to the used black-box algorithm so that Random Forest and ANN can be substituted by any other classification model.

For the sake of clarity, we presented the results in Fig. B.2 like we did in Fig. B.1. The upper row correspond to a target image that have been classified as malign, and for which we want to find the closest benign virtual instance. The lower row depicts the opposite example, as the target observation have been classified as benign and we search for the virtual malign closest instance. The left panel of both row present the projection of all instances in the two dimensional plane composed by the two features for which a change in class have been easily found. In both, squares represent benign instances and red triangles malign ones. The central and right panel shows the distribution of the created virtual instances when only one feature at a time can be modified (middle panel) or when combination of two features are perturbed (right panel). The solid black symbol representing the original instance (whether it be malign - upper

row - or benign - lower row) and the the hollow ones represent the best solutions. Note that in the second case (*i.e.* benign to malign) no one-dimension solution have been encountered; the variation of two feature at the same tine was necessary to trigger a switch of class.

This exemplify how the methodology can be of help in real life situations. The diagnostic of the physician looking at the fine needle aspirate output can conflict with the MDDS diagnostic of the nature of the cancer. In spite of the black-box nature of the system, running the proposed methodology on top of it would yield sufficient information about its rationale to help the physician gain additional knowledge about the reason behind the machine's classification and to decide in a fully informed way which way to go.

B.3 Case study: Portuguese wine quality

The second case study proposes to understand the relationship between quantitative characteristics of 4898 Portuguese white wines and their qualitative rank. The data set if described in [CCA+09] and available at [WQd16]. Each wine is described by 12 physiochemical features: fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, alcohol. The qualitative rank of a wine originally ranges from 0 (bad) to 10 (excellent) based on experts sensory taste test; it have been narrowed to a bivariate situation, for the sake of clarity, grouping together wines of quality inferior to 6 in class 1 (lower quality) and the others in class 2 (higher quality).

This dataset will allow to present how the proposed methodology not only can help at understanding the rationale behind the black-box algorithm but also to pinpoint which feasible modifications (or corrections) must be administered to trigger a change of class of an instance.

Classification model	**Accuracy No CV**	**Accuracy LOOCV**	**Avg. training time (sec.)**
Random Forest	97.48%	76.77%	3.62
ANN	72.64%	72.30%	2.01

Table B.2: Classification scores and average training time for the wine data set, for the two considered classification algorithms.

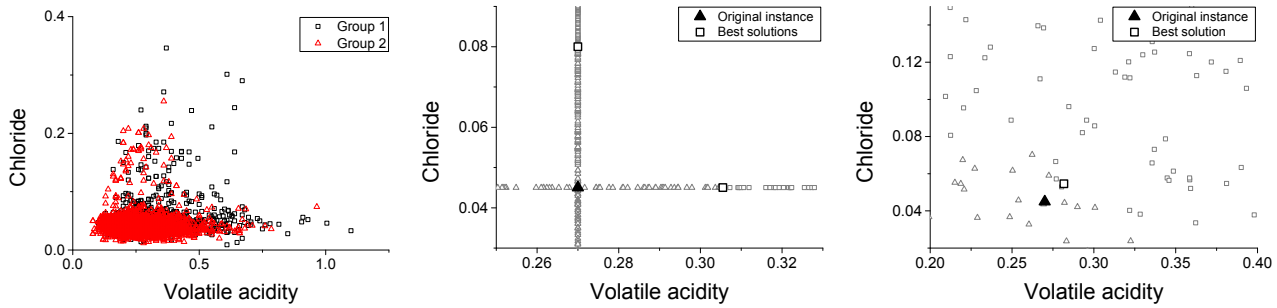

Figure B.3: Graphical representation of the proposed methodology applied to the wine data set. Left, centre and right panels respectively depict the original data set, the search by varying one single feature, and the full search in the plane. See main text for details.

In this particular case, a producer will want to know what features of his low quality labelled wine to change in order to make it of higher quality. By altering some characteristics of the production chain, the producer might be able to reach a upgrade of quality with a controlled change in some features. The producer might certainly appreciate a solution entailing the fewer (thus less costly) changes in its production process, therefore low-dimensional solutions are privileged.

Similarly to the previous case, Tab. B.2 reports the classification performance achieved and Fig. B.3 present the search procedure to pass from a lower to a higher quality instance. Both groups of wines are projected in left panel along the two most important features in swapping class (*i.e.* volatile acidity and chloride concentration) - low quality wine as black squares and higher quality ones as red triangles). In the central and right panels the original instance is presented as a black triangle, the virtual instances are hollow triangles or squares depending on their forecasted label. Big squares represent therefore the closest instance of the higher quality group to the original wine. It can be appreciated that the bidimensional solution is much closer to the target wine than the two unidimensional solutions, suggesting that the producer will have to choose between a aggressive, yet easier to implement, modification or a softer but unwieldy two-parameter change for his/her product.

It is then of interest to look at the general distance between the best solution and the targeted instance. Fig. B.4 specifically plots the histograms of the distances to best solution for ANN and Random Forest in both uni- and two-dinensional case. In general, most of the solutions

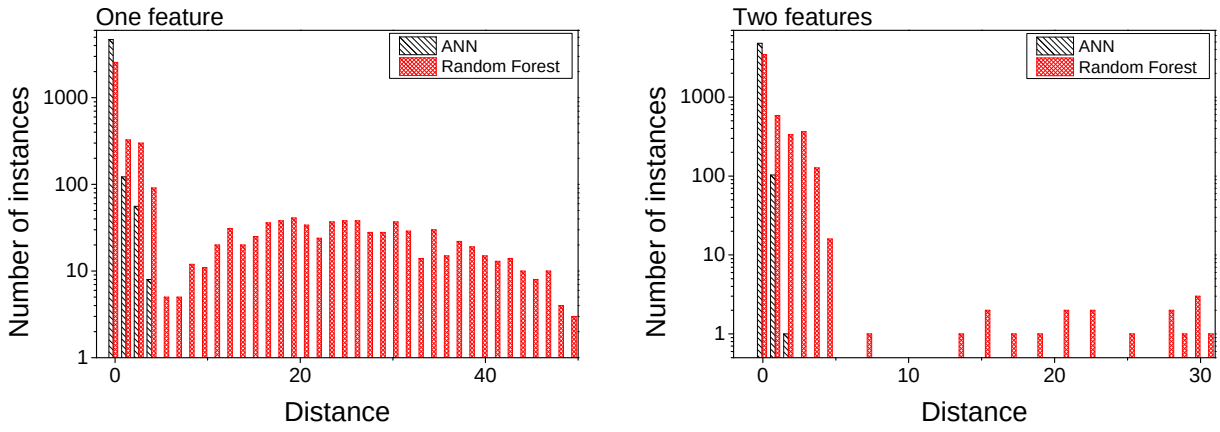

Figure B.4: Histograms of the distance between the target and the best virtual instance, for changes in one (left panel) and two (right panel) features.

are encountered close to the targeted instance (note the vertical $\log_{10}$ axis). Whilst ANN derived solutions are always close to target, Random Forest seem to need the introduction of a supplementary feature to transform a far away 1D solution into a closer 2D solution. This difference between the two algorithms is due to their structural characteristics. ANN being a more complex non-linear classification model is therefore more sensitive to small changes and subsequently less resilient to any noise present in the data.

B.4 Solution optimality and computation cost

However good (and computationally tractable) seems to work the proposed algorithm, a doubt subsists. We mentioned before a brute force research algorithm capable of scanning the entire solution space. What if a better solution exists providing no dimension limitations? What if the proposed methodology yield a local-minima and therefore a sub-optimal solution? This might not be a problem *per se* though it may certainly introduces a bias to the model extracted rationale.

Let us then compare the results with brute force extracted ones. For this, the Simulated Annealing (SA) algorithm [Hwa88] is an optimisation technique, inspired from metallurgy, aimed at encountering a global optimum solution across the solution space in a limited fixed amount of time. The SA stochastically tests solutions around the best result - in the Euclidean

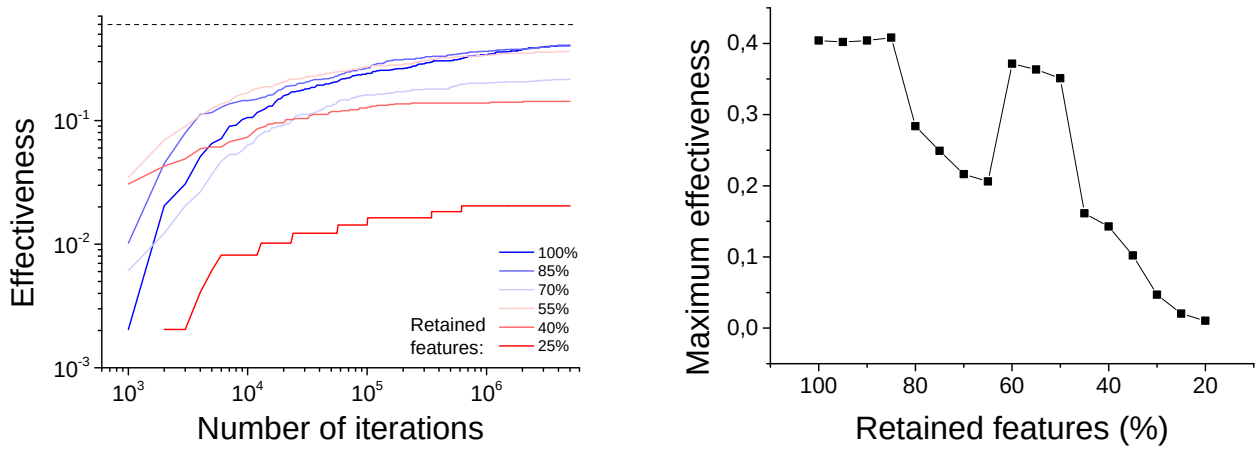

Figure B.5: Efficiency of a Simulated Annealing optimisation. The six curves in the left panel represent the evolution of the SA success (*i.e.* the fraction of times it finds a solution better than the proposed methodology) as a function both of the percentage of features included in the search, and of the number of iterations. Additionally, the right panel depicts the maximum achieved success as a function of the percentage of features included in the search.

distance sense - while progressively narrowing the search radius. We have complemented the standard SA algorithm introducing a parameter controlling the number of features of the search, amplifying the spectrum of results: from the unrestricted standard SA output to results focusing only on a subset of features (similarly to the proposed methodology). So, starting from multiple copies of the original target, the algorithm scans solution in its neighbourhood yielding a subset of solutions using the selected features.

Not only the proposed methodology yields as good results as those found through SA; but it does so with a significant reduction of computational time. Fig. B.5 (left panel) reports the evolution of the fraction of times SA led to better (*i.e.* closer to the target instance) result as a function of the number of iteration of the search and of the percentage of features allowed into the search space. The results have been condensed in right figure to only display the maximum effectiveness of the SA (*i.e.* the proportion of times it outperformed the proposed methodology) as a function of the fraction of retained features. Whilst SA seems to handle badly a low number of features (which can be expected as the chosen features might be irrelevant to the solution), results further suggests a similar behaviour of both methodologies when high number of features are considered as an effectiveness of 0.5 is approached. Yet, left panel indicates that such situations is concomitant to a large number of iteration, implying a significant computational

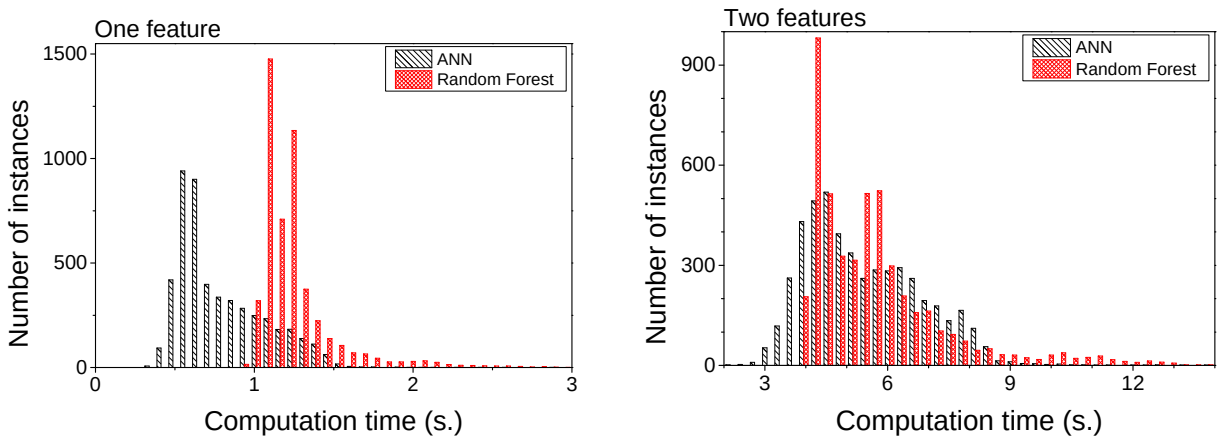

Figure B.6: Histogram of the computation time associated with the proposed methodology, for one (left panel) and two (right panel) features searches.

cost, while Fig. B.6 reports times of computation two order of magnitude faster for the proposed methodology. Left and right panels respectively displaying the histogram of time elapsed to found the best solution for both ANN and Random Forest methods and for both uni- and two-dimensional searches, suggesting that solutions are generally encountered in less than 10 seconds.

B.5 Conclusions and discussion

We have shown that, independently of the nature of the classification model, we have been able to extract post-hoc, low computational intelligence allowing to increase the interpretability of black-box complex classification models. The method has the additional benefit of being modular, thus being adaptive to the analysed problem. Also, there is no practical reason to limit this method to case similar to those presented: the same approach can be imagined for an un-supervised classification task in order to improve the interpretability of the clusters.

Two important points must be discussed. How does the methodology react to noisy or wrong classification models? and secondly, how does the *a priori* feature selection condition the resulting solution?

Whilst the presented case study presented rather strong classification score, real-world situation

might be much more noisy, as the classification model might only be able to grasp a fraction of the complexity of the reality. These limitations might be due to the intrinsic limitations of the model, or to the presence of noise in the training data, situation commonly referred to as learning with noisy labels [NDRT13]. This case is not specific to the proposed methodology as even the best Decision Support System (DSS) can be wrong. However, this methodology might help handling such situations as the closeness of a solution might be used as a proxy of the "confidence" of the classification model. Therefore, the gain in understanding of the model's rationale combined with one's own knowledge might yield to a better awareness of the situation, even under noisy situations.

Secondly, Fig. B.7 highlight a linear relationship between the features more relevant to the methodology and the ones relevant to the global classification model (represented by the dashed red line with a slope of 1.478 ± 0.33, $R^2 = 0,830$). The feature importance of the model are extracted through their Gini importance [BFOS84]. While this relationship is not perfect, it suggests that the features used for by the model can be used to derive corrections for the system. Specifically, let us imagine an airborne system monitoring the flight's characteristics and its surroundings to assess the probability of it suffering an unstable approach. Therefore, the same feature used in the assessment might be used to derive correction that the flight might

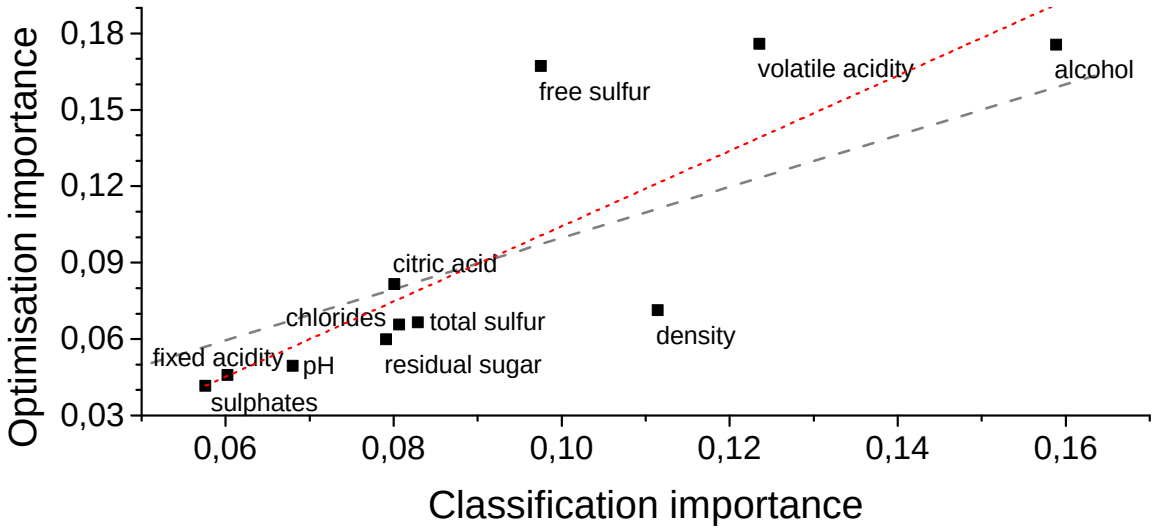

Figure B.7: Importance of features in the classification as a function of their importance in the proposed methodology, for the whine dataset - see main text for definitions. The grey dashed line represents the identity function, while the red one the best linear fit.

apply in order to avoid such landing problems.

Bibliography

[AAB+16] Martın Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

[ABCX09] Sergio Arianos, E Bompard, A Carbone, and Fei Xue. Power grid vulnerability: A complex network approach. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 19(1):013119, 2009.

[ABEG01] SS Allan, JA Beesley, JE Evans, and SG Gaddy. Analysis of delay causality at newark international airport. In *4th USA/Europe Air Traffic Management R&D Seminar*, 2001.

[ABL06] Soufian Ben Amor, Marc Bui, and Ivan Lavallée. A complex systems approach in atm modeling. In *Doctoral Symposium–International Conference on Research in Air Transportation*, pages 24–28, 2006.

[ACGB08] Shervin AhmadBeygi, Amy Cohn, Yihan Guan, and Peter Belobaba. Analysis of the potential for delay propagation in passenger airline networks. *Journal of air transport management*, 14(5):221–236, 2008.

[ACL+11] Juan A Almendral, Regino Criado, Inmaculada Leyva, Javier M Buldú, and Irene Sendina-Nadal. Introduction to focus issue: Mesoscales in complex networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 21(1):016101, 2011.

[ACNR07] Marco Alderighi, Alessandro Cento, Peter Nijkamp, and Piet Rietveld. Assessment of new hub-and-spoke and point-to-point airline network configurations. *Transport Reviews*, 27(5):529–549, 2007.

[AGE01] SS Allan, SG Gaddy, and JE Evans. Delay causality and reduction at the new york city airports using terminal weather information systems. Technical report, Lincoln Laboratory, Massachusetts Institute of Technology Cambridge, Mass, USA, 2001.

[AJB00] Réka Albert, Hawoong Jeong, and Albert-László Barabási. Error and attack tolerance of complex networks. *nature*, 406(6794):378–382, 2000.

[AMMIL12] Yaser S Abu-Mostafa, Malik Magdon-Ismail, and Hsuan-Tien Lin. *Learning from data*, volume 4. AMLBook New York, NY, USA:, 2012.

[AMS04] Nicola Ancona, Daniele Marinazzo, and Sebastiano Stramaglia. Radial basis function approach to nonlinear granger causality of time series. *Physical Review E*, 70(5):056221, 2004.

[ASRA04] Khaled F Abdelghany, Sharmila S Shah, Sidhartha Raina, and Ahmed F Abdelghany. A model for projecting flight delays during irregular operation conditions. *Journal of Air Transport Management*, 10(6):385–394, 2004.

[ASZ12] Gautam Ahuja, Giuseppe Soda, and Akbar Zaheer. The genesis and dynamics of organizational networks. *Organization Science*, 23(2):434–448, 2012.

[Bag08] Ganesh Bagler. Analysis of the airport network of india as a complex weighted network. *Physica A: Statistical Mechanics and its Applications*, 387(12):2972–2980, 2008.

[BCPZ16] Seddik Belkoura, Andrew Cook, José Maria Peña, and Massimiliano Zanin. On the multi-dimensionality and sampling of air transport networks. *Transportation Research Part E: Logistics and Transportation Review*, 94:95–109, 2016.

[BDL+04] Andrea Brovelli, Mingzhou Ding, Anders Ledberg, Yonghong Chen, Richard Nakamura, and Steven L Bressler. Beta oscillations in a large-scale sensorimotor cortical network: directional influences revealed by granger causality. *Proceedings of the National Academy of Sciences of the United States of America*, 101(26):9849–9854, 2004.

[Bel57] Richard Bellman. Dynamic programming, 1957.

[Ber07] Eta S Berner. *Clinical decision support systems*. Springer, 2007.

[Ber08] Ralf Berghof. Prr 2007-an assessment of air traffic management in europe during the calendar year 2007. performance review report 2007. *PRR 2007*, 2008.

[BF16] Adrien Bibal and Benoît Frenay. Interpretability of machine learning models and representations: an introduction. In *24th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pages 77–82, 2016.

[BFOS84] Leo Breiman, Jerome H Friedman, Richard A Olshen, and Charles J Stone. Classification and regression trees. wadsworth & brooks. *Monterey, CA*, 1984.

[BHBR99] Roger Beatty, Rose Hsu, Lee Berry, and James Rome. Preliminary evaluation of flight delay propagation through an airline schedule. *Air Traffic Control Quarterly*, 7(4):259–270, 1999.

[BLM+06] Stefano Boccaletti, Vito Latora, Yamir Moreno, Martin Chavez, and D-U Hwang. Complex networks: Structure and dynamics. *Physics reports*, 424(4):175–308, 2006.

[BLW76] Norman Biggs, E Keith Lloyd, and Robin J Wilson. *Graph Theory, 1736-1936*. Oxford University Press, 1976.

[Bon07] Phillip Bonacich. Some unique properties of eigenvector centrality. *Social Networks*, 29(4):555–564, 2007.

[BPZ16] Seddik Belkoura, José Maria Peña, and Massimiliano Zanin. Generation and recovery of airborne delays in air transport. *Transportation Research Part C: Emerging Technologies*, 69:436–450, 2016.

[BPZ17] Seddik Belkoura, José Maria Peña, and Massimiliano Zanin. Beyond linear delay multipliers in air transport. *Journal of Advanced Transportation*, 2017, 2017.

[Bre01] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[BS09] Ed Bullmore and Olaf Sporns. Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience*, 10(3):186–198, 2009.

[BS11] Steven L Bressler and Anil K Seth. Wiener–granger causality: a well established methodology. *Neuroimage*, 58(2):323–329, 2011.

[BY13] Kenneth Button and Junyang Yuan. Airfreight transport and economic development: an examination of causality. *Urban Studies*, 50(2):329–340, 2013.

[BZ16] Seddik Belkoura and Massimiliano Zanin. Phase changes in delay propagation networks. *arXiv preprint arXiv:1611.00639*, 2016.

[CBZ16] Andrew Cook, Seddik Belkoura, and Massimiliano Zanin. ATM performance measurement in Europe, the US and China. *Chinese Journal of Aeronautics*, in press, 2016.

[CC04] LP Chi and X Cai. Structural changes caused by error and attack tolerance in us airport network. *International Journal of Modern Physics B*, 18(17n19):2394–2400, 2004.

[CC09] Yu-Hern Chang and Yu-Wei Chang. Air cargo expansion and economic growth: Finding the empirical link. *Journal of Air Transport Management*, 15(5):264–265, 2009.

[CCA+09] Paulo Cortez, António Cerdeira, Fernando Almeida, Telmo Matos, and José Reis. Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47(4):547–553, 2009.

[CCGJ02] Qian Chen, Hyunseok Chang, Ramesh Govindan, and Sugih Jamin. The origin of power laws in internet topologies revisited. In *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 608–617. IEEE, 2002.

[CDLHJ07] Sandrine Carlier, Ivan De Lépinay, Jean-Claude Hustache, and Frank Jelinek. Environmental impact of air traffic flow management delays. In *7th USA/Europe air traffic management research and development seminar (ATM2007)*, volume 2, page 16, 2007.

[CF12] Clement Kong Wing Chow and Michael Ka Yiu Fung. Measuring the effects of china's airline mergers on the productivity of state-owned carriers. *Journal of Air Transport Management*, 25:1–4, 2012.

[CFT97] Marvin S Cohen, Jared T Freeman, and Bryan B Thompson. Integrated critical thinking training and decision support for tactical anti-air warfare. In *Proceedings of the 1997 Command and Control Research and Technology Symposium*, 1997.

[CGGZ+13] Alessio Cardillo, Jesús Gómez-Gardenes, Massimiliano Zanin, Miguel Romance, David Papo, Francisco del Pozo, and Stefano Boccaletti. Emergence of network features from multiplexity. *Scientific reports*, 3, 2013.

[CH93] Mark C Cross and Pierre C Hohenberg. Pattern formation outside of equilibrium. *Reviews of modern physics*, 65(3):851, 1993.

[CLMR03] Paolo Crucitti, Vito Latora, Massimo Marchiori, and Andrea Rapisarda. Efficiency of scale-free networks: error and attack tolerance. *Physica A: Statistical Mechanics and its Applications*, 320:622–642, 2003.

[CLZ15] Qian Cao, Jinfeng Lv, and Jun Zhang. Productivity efficiency analysis of the airlines in china after deregulation. *Journal of Air Transport Management*, 42:135–140, 2015.

[CMS99] Robert Cooley, Bamshad Mobasher, and Jaideep Srivastava. Data preparation for mining world wide web browsing patterns. *Knowledge and information systems*, 1(1):5–32, 1999.

[COD14] CODA. Delays to air transport in europe - annual 2013, 2014.

[COJT+11] Luciano da Fontoura Costa, Osvaldo N Oliveira Jr, Gonzalo Travieso, Francisco Aparecido Rodrigues, Paulino Ribeiro Villas Boas, Lucas Antiqueira, Matheus Palhares Viana, and Luis Enrique Correa Rocha. Analyzing and modeling real-world phenomena with complex networks: a survey of applications. *Advances in Physics*, 60(3):329–412, 2011.

[Com15] Performance Review Commission. Performance review report 2014 – an assessment of air traffic management in europe during the calendar year 2014. Technical report, Brussels (Belgium): EUROCONTROL, Performance Review Commission, 2015.

[CRTVB07] L da F Costa, Francisco A Rodrigues, Gonzalo Travieso, and Paulino Ribeiro Villas Boas. Characterization of complex networks: A survey of measurements. *Advances in physics*, 56(1):167–242, 2007.

[CT11] Andrew J Cook and Graham Tanner. European airline delay cost reference values. 2011.

[CTA04] Andrew J Cook, Graham Tanner, and Stephen Anderson. Evaluating the true cost to airlines of one minute of airborne or ground delay: final report. 2004.

[CTCZ15] A Cook, G Tanner, S Cristóbal, and M Zanin. Delay propagation–new metrics, new insights. In *11th USA/Europe Air Traffic Management Research and Development Seminar*, 2015.

[CTJL09] Andrew J Cook, Graham Tanner, Radosav Jovanovic, and Adrian Lawes. The cost of delay to air transport in europe: quantification and management. In *Air Transport Research Society (ATRS) World Conference, Abu Dhabi*, 2009.

[CTZ13] Andrew Cook, Graham Tanner, and Massimiliano Zanin. Towards superior air transport performance metrics–imperatives and methods. *Journal of Aerospace Operations*, 2(1-2):3–19, 2013.

[CV95] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[CWS+03] Li-Ping Chi, Ru Wang, Hang Su, Xin-Ping Xu, Jin-Song Zhao, Wei Li, and Xu Cai. Structural properties of us flight network. *Chinese Physics Letters*, 20(8):1393, 2003.

[CZGG+13] Alessio Cardillo, Massimiliano Zanin, Jesús Gómez-Gardenes, Miguel Romance, Alejandro J García del Amo, and Stefano Boccaletti. Modeling the multi-layer nature of the european air transport network: Resilience and passengers re-scheduling under random failures. *The European Physical Journal Special Topics*, 215(1):23–33, 2013.

[DDSRC+13] Manlio De Domenico, Albert Solé-Ribalta, Emanuele Cozzo, Mikko Kivelä, Yamir Moreno, Mason A Porter, Sergio Gómez, and Alex Arenas. Mathematical formulation of multilayer networks. *Physical Review X*, 3(4):041022, 2013.

[dig15] CODA digest. all-causes delay and cancellations to air transport in europe – 2014. Technical report, Brussels (Belgium): EUROCONTROL, Central Office for Delay Analysis, 2015.

[DP09] Luis Delgado and Xavier Prats. Fuel consumption assessment for speed variation concepts during the cruise phase. In *Proceedings of the Conference on Air Traffic Management (ATM) Economics*, 2009.

[DP12] Luis Delgado and Xavier Prats. En route speed reduction concept for absorbing air traffic flow management delays. *Journal of Aircraft*, 49(1):214–224, 2012.

[Duy93] DIRK Duytschaever. The development and implementation of the eurocontrol central air traffic flow management unit (cfmu). *Journal of Navigation*, 46(03):343–352, 1993.

[DW16] Sarah Dunn and Sean M Wilkinson. Increasing the resilience of air traffic networks using a network graph theory approach. *Transportation Research Part E: Logistics and Transportation Review*, 90:39–50, 2016.

[DYB03] Gerald F Davis, Mina Yoo, and Wayne E Baker. The small world of the american corporate elite, 1982-2001. *Strategic organization*, 1(3):301–326, 2003.

[DZL+16] Wen-Bo Du, Xing-Lian Zhou, Oriol Lordan, Zhen Wang, Chen Zhao, and Yan-Bo Zhu. Analysis of the chinese airline network as multi-layer networks. *Transportation Research Part E: Logistics and Transportation Review*, 89:108–116, 2016.

[ET95] Bradley Efron and Robert J Tibshirani. *Cross-validation and the bootstrap: Estimating the error rate of a prediction rule*. Division of Biostatistics, Stanford University, 1995.

[Eur07] Eurocontrol Performance Review Commission. Performance review report 2007: An assessment of air traffic management in europe during the calendar year 2007, 2007.

[EUR09] EUROCONTROL. A white paper on resilience engineering for atm, 2009.

[EUR14] Federal Aviation Administration EUROCONTROL. Comparison of air traffic management-related operational performance: U.s./ europe – 2013. Technical report, Brussels, Belgium and Washington DC, USA: EUROCONTROL and Federal Aviation Administration, 2014.

[EUR15] EUROCONTROL. Standard inputs for eurocontrol cost benefit analyses. Technical report, Eurocontrol, Brussels, Belgium,, 2015.

[FKHS13] John Ferguson, Abdul Qadar Kara, Karla Hoffman, and Lance Sherry. Estimating domestic us airline cost of delay based on european model. *Transportation Research Part C: Emerging Technologies*, 33:311–323, 2013.

[For10] Santo Fortunato. Community detection in graphs. *Physics reports*, 486(3):75–174, 2010.

[FRB83] Wolfgang Fichtner, Donald J Rose, and Randolph E Bank. Semiconductor device simulation. *SIAM Journal on Scientific and Statistical Computing*, 4(3):391–415, 1983.

[Fre91] David A Freedman. Statistical models and shoe leather. *Sociological methodology*, pages 291–313, 1991.

[FRE13] Pablo Fleurquin, José J Ramasco, and Victor M Eguiluz. Systemic delay propagation in the us airport network. *arXiv preprint arXiv:1301.1136*, 2013.

[Fre14] Alex A. Freitas. Comprehensible classification models: A position paper. *SIGKDD Explor. Newsl.*, 15(1):1–10, March 2014.

[FVB+99] Winrich A Freiwald, Pedro Valdes, Jorge Bosch, Rolando Biscay, Juan Carlos Jimenez, Luis Manuel Rodriguez, Valia Rodriguez, Andreas K Kreiter, and Wolf Singer. Testing non-linearity and directedness of interactions between neural groups in the macaque inferotemporal cortex. *Journal of neuroscience methods*, 94(1):105–119, 1999.

[GB08] Thilo Gross and Bernd Blasius. Adaptive coevolutionary networks: a review. *Journal of The Royal Society Interface*, 5(20):259–271, 2008.

[GCLR06] Jody Hoffer Gittell, Kim Cameron, Sandy Lim, and Victor Rivas. Relationships, layoffs, and organizational resilience airline industry responses to

september 11. *The Journal of Applied Behavioral Science*, 42(3):300–329, 2006.

[GE03] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182, 2003.

[GHW79] Gene H Golub, Michael Heath, and Grace Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.

[GI95] Jerrold W Grossman and Patrick DF Ion. On a portion of the well-known collaboration graph. *Congressus Numerantium*, pages 129–132, 1995.

[Glo96] Gregory Glockner. Effects of air traffic congestion delays under several flow-management policies. *Transportation Research Record: Journal of the Transportation Research Board*, (1517):29–36, 1996.

[Glu12] O. Gluchshenko. Definitions of disturbance, resilience and robustness in atm context, 2012.

[GM95] Murray Gell-Mann. *The Quark and the Jaguar: Adventures in the Simple and the Complex*. Macmillan, 1995.

[GM07] Michele Guida and Funaro Maria. Topology of the italian airport network: A scale-free small-world network with a fractal structure? *Chaos, Solitons & Fractals*, 31(3):527–536, 2007.

[Gof74] Erving Goffman. *Frame analysis: An essay on the organization of experience.* Harvard University Press, 1974.

[Goo68] Nelson Goodman. *Languages of art: An approach to a theory of symbols.* Hackett publishing, 1968.

[GP96] Donald J Grout and Claude V Palisca. A history of western music, 5 ‘“ed, 1996.

[Gra80] Clive WJ Granger. Testing for causality: a personal viewpoint. *Journal of Economic Dynamics and control*, 2:329–352, 1980.

[Gra88a] Clive WJ Granger. Causality, cointegration, and control. *Journal of Economic Dynamics and Control*, 12(2-3):551–559, 1988.

[Gra88b] Clive WJ Granger. Some recent development in a concept of causality. *Journal of econometrics*, 39(1-2):199–211, 1988.

[GSB+16] Jan Gerretzen, Ewa Szymańska, Jacob Bart, Antony N. Davies, Henk-Jan van Manen, Edwin R. van den Heuvel, Jeroen J. Jansen, and Lutgarde M.C. Buydens. Boosting model performance and interpretation by entangling preprocessing selection and variable selection. *Analytica Chimica Acta*, 938:44 – 52, 2016.

[Gun00] Lance H Gunderson. Ecological resilience–in theory and application. *Annual review of ecology and systematics*, pages 425–439, 2000.

[GVC+14] Gérald Gurtner, Stefania Vitali, Marco Cipolla, Fabrizio Lillo, Rosario Nunzio Mantegna, Salvatore Miccichè, and Simone Pozzi. Multi-scale analysis of the european airspace using network community detection. *PloS one*, 9(5):e94414, 2014.

[Han02] Mark Hansen. Micro-level analysis of airport delay externalities using deterministic queuing models: a case study. *Journal of Air Transport Management*, 8(2):73–87, 2002.

[HARA13] Murad Hossain, Sameer Alam, Tim Rees, and Hussein Abbass. Australian airport network robustness analysis: a complex network approach. In *Proceeding of the 36th Australasian Transport Research Forum, Brisbane, Australia*, 2013.

[HDB+96] Martin T Hagan, Howard B Demuth, Mark H Beale, et al. Neural network design, pws pub. *Co., Boston*, 3632, 1996.

[HH13] Lu Hao and Mark Hansen. How airlines set scheduled block times. In *10th USA/Europe Air Traffic Management Research and Development Seminar, Chicago IL*, 2013.

[HMAS03] Wolfram Hesse, Eva Möller, Matthias Arnold, and Bärbel Schack. The use of time-variant eeg granger causality for inspecting directed interdependencies of neural assemblies. *Journal of neuroscience methods*, 124(1):27–44, 2003.

[Hol05] Petter Holme. Core-periphery organization of complex networks. *Physical Review E*, 72(4):046111, 2005.

[Hoo01] Kevin D Hoover. *Causality in macroeconomics*. Cambridge University Press, 2001.

[HS12] Petter Holme and Jari Saramaki. Temporal networks. *Physics reports*, 519(3):97–125, 2012.

[HS13] Petter Holme and Jari Saramaki. Temporal networks. understanding complex systems, 2013.

[HT90] Trevor J Hastie and Robert J Tibshirani. *Generalized additive models*, volume 43. CRC press, 1990.

[HV08] Sander J Hebly and Hendrikus G Visser. *Advanced noise abatement departure procedures: custom optimized departure profiles*. American Institute of Aeronautics and Astronautics AIAA, 2008.

[Hwa88] Chii-Ruey Hwang. Simulated annealing: theory and applications. *Acta Applicandae Mathematicae*, 12(1):108–111, 1988.

[HWL07] Erik Hollnagel, David D Woods, and Nancy Leveson. *Resilience engineering: concepts and precepts*. Ashgate Publishing, Ltd., 2007.

[J+08] Matthew O Jackson et al. *Social and economic networks*, volume 3. Princeton university press Princeton, 2008.

[Jan05] Milan Janić. Modeling the large scale disruptions of an airline network. *Journal of transportation engineering*, 131(4):249–260, 2005.

[Jan15] Milan Janić. Modelling the resilience, friability and costs of an air transport network affected by a large-scale disruptive event. *Transportation Research Part A: Policy and Practice*, 71:1–16, 2015.

[Jet09] Martina Jetzki. The propagation of air transport delays in europe. *Master's thesis, RWTH Aachen University, Airport and Air Transportation Research*, 2009.

[JJ12] Tao Jia and Bin Jiang. Building and analyzing the us airport network based on en-route location information. *Physica A: Statistical Mechanics and its Applications*, 391(15):4031–4042, 2012.

[JZ97] Anil Jain and Douglas Zongker. Feature selection: Evaluation, application, and small sample performance. *IEEE transactions on pattern analysis and machine intelligence*, 19(2):153–158, 1997.

[K+95] Ron Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Stanford, CA, 1995.

[KDTB01] Maciej Kamiński, Mingzhou Ding, Wilson A Truccolo, and Steven L Bressler. Evaluating causal relations in neural systems: Granger causality, directed transfer function and statistical assessment of significance. *Biological cybernetics*, 85(2):145–157, 2001.

[KQJWBXB12] Cai Kai-Quan, Zhang Jun, Du Wen-Bo, and Cao Xian-Bin. Analysis of the chinese air route network as a complex network. *Chinese Physics B*, 21(2):028903, 2012.

[KR09] Mark J Koetse and Piet Rietveld. The impact of climate change and weather on transport: An overview of empirical findings. *Transportation Research Part D: Transport and Environment*, 14(3):205–221, 2009.

[KSJ+00] Eric R Kandel, James H Schwartz, Thomas M Jessell, Steven A Siegelbaum, and AJ Hudspeth. *Principles of neural science*, volume 4. McGraw-hill New York, 2000.

[LC04] Wei Li and Xu Cai. Statistical analysis of airport network of china. *Physical Review E*, 69(4):046106, 2004.

[Lip16] Zachary C. Lipton. The mythos of model interpretability. In *2016 ICML Workshop on Human Interpretability in Machine Learning (WHI 2016)*, pages 96–100, 2016.

[LKJ06] Sang Hoon Lee, Pan-Jun Kim, and Hawoong Jeong. Statistical properties of sampled networks. *Physical Review E*, 73(1):016102, 2006.

[LKLG14] Kyu-Min Lee, Jung Yeol Kim, Sangchul Lee, and K-I Goh. Multiplex networks. In *Networks of networks: The last frontier of complexity*, pages 53–72. Springer, 2014.

[LM01] Vito Latora and Massimo Marchiori. Efficient behavior of small-world networks. *Physical review letters*, 87(19):198701, 2001.

[LSS14] Oriol Lordan, Jose M Sallan, and Pep Simo. Study of the topology and robustness of airline route networks from the complex network approach: a survey and research agenda. *Journal of Transport Geography*, 37:112–120, 2014.

[LWG93] Tae-Hwy Lee, Halbert White, and Clive WJ Granger. Testing for neglected nonlinearity in time series models: A comparison of neural network methods and alternative tests. *Journal of Econometrics*, 56(3):269–290, 1993.

[Mil94] Randolph A Miller. Medical diagnostic decision support systems—past, present, and future. *Journal of the American Medical Informatics Association*, 1(1):8–27, 1994.

[ML01] Robert M May and Alun L Lloyd. Infection dynamics on scale-free networks. *Physical Review E*, 64(6):066112, 2001.

[MMG14] Mark A Musen, Blackford Middleton, and Robert A Greenes. Clinical decision-support systems. In *Biomedical informatics*, pages 643–674. Springer, 2014.

[MNP04] Yamir Moreno, Maziar Nekovee, and Amalio F Pacheco. Dynamics of rumor spreading in complex networks. *Physical Review E*, 69(6):066130, 2004.

[MS99] Rosario N Mantegna and H Eugene Stanley. *Introduction to econophysics: correlations and complexity in finance*. Cambridge university press, 1999.

[MSF10] Marcial Marazzo, Rafael Scherre, and Elton Fernandes. Air transport demand and economic growth in brazil: A time series analysis. *Transportation Research Part E: Logistics and Transportation Review*, 46(2):261–269, 2010.

[MSOI+02] Ron Milo, Shai Shen-Orr, Shalev Itzkovitz, Nadav Kashtan, Dmitri Chklovskii, and Uri Alon. Network motifs: simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002.

[MSW95] Olvi L Mangasarian, W Nick Street, and William H Wolberg. Breast cancer diagnosis and prognosis via linear programming. *Operations Research*, 43(4):570–577, 1995.

[MT13] Kirsi Mukkala and Hannu Tervo. Air transportation and regional growth: which way does the causality run? *Environment and Planning A*, 45(6):1508–1520, 2013.

[NDRT13] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. In *Advances in neural information processing systems*, pages 1196–1204, 2013.

[Nea14] Zachary Neal. The devil is in the details: Differences in air traffic networks by scale, species, and season. *Social Networks*, 38:63–73, 2014.

[New01a] Mark EJ Newman. Scientific collaboration networks. i. network construction and fundamental results. *Physical review E*, 64(1):016131, 2001.

[New01b] Mark EJ Newman. The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences*, 98(2):404–409, 2001.

[New02] Mark EJ Newman. Spread of epidemic disease on networks. *Physical Review E*, 66(1):016128, 2002.

[NG04] Mark EJ Newman and Michelle Girvan. Finding and evaluating community structure in networks. *Physical review E*, 69(2):026113, 2004.

[NMEI09] Robert Nisbet, Gary Miner, and John Elder IV. *Handbook of statistical analysis and data mining applications*. Academic Press, 2009.

[NPRN14] Jenaro Nosedal, Miquel A Piera, Sergio Ruiz, and Alvaro Nosedal. An efficient algorithm for smoothing airspace congestion by fine-tuning take-off times. *Transportation Research Part C: Emerging Technologies*, 44:171–184, 2014.

[oC16] Civil Aviation Administration of China. Statistical bulletin of civil aviation industry development in 2014. Technical report, Beijing (China): Civil Aviation Administration of China, 2016.

[PFT+99] Anne Péguin-Feissolle, Timo Teräsvirta, et al. A general framework for testing the granger noncausality hypothesis. *Stockholm School of Economics Working Paper Series in Economics and Finance*, (343):287–290, 1999.

[PMO13] Nikolas Pyrgiotis, Kerry M Malone, and Amedeo Odoni. Modelling delay propagation within an airport network. *Transportation Research Part C: Emerging Technologies*, 27:60–75, 2013.

[PVG+11] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830, 2011.

[PWNT09] Tamara Pejovic, Victoria Williams, Robert Noland, and Ralf Toumi. Factors affecting the frequency and severity of airport weather delays and the impli-

cations of climate change for future delays. *Transportation Research Record: Journal of the Transportation Research Board*, (2139):97–106, 2009.

[PZMB14] David Papo, Massimiliano Zanin, and Javier Martin Buldú. Reconstructing functional brain networks: have we got the basics right? *Frontiers in human neuroscience*, 8:107, 2014.

[RB14] Juan Jose Rebollo and Hamsa Balakrishnan. Characterization and prediction of air traffic delays. *Transportation research part C: Emerging technologies*, 44:231–241, 2014.

[RC17] Filippo Radicchi and Claudio Castellano. Maximum entropy sampling in complex networks. *arXiv preprint arXiv:1703.03858*, 2017.

[RCC+04] Filippo Radicchi, Claudio Castellano, Federico Cecconi, Vittorio Loreto, and Domenico Parisi. Defining and identifying communities in networks. *Proceedings of the National Academy of Sciences of the United States of America*, 101(9):2658–2663, 2004.

[RFG05] Alard Roebroeck, Elia Formisano, and Rainer Goebel. Mapping directed influence over the brain using granger causality and fmri. *Neuroimage*, 25(1):230–242, 2005.

[RG01] Carl Edward Rasmussen and Zoubin Ghahramani. Occam's razor. *Advances in neural information processing systems*, pages 294–300, 2001.

[RG16] Ladislav Rampasek and Anna Goldenberg. Tensorflow: Biology's gateway to deep learning? *Cell systems*, 2(1):12–14, 2016.

[RPL10] Juan D Rodriguez, Aritz Perez, and Jose A Lozano. Sensitivity analysis of k-fold cross validation in prediction error estimation. *IEEE transactions on pattern analysis and machine intelligence*, 32(3):569–575, 2010.

[RSG16a] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Model-agnostic interpretability of machine learning. In *2016 ICML Workshop on Human Interpretability in Machine Learning (WHI 2016)*, pages 91–95, 2016.

[RSG16b] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 1135–1144, New York, NY, USA, 2016. ACM.

[RSNC09] Aura Reggiani, Sara Signoretti, Peter Nijkamp, and Alessandro Cento. Network measures in civil air transport: A case study of lufthansa. In AhmadK. Naimzada, Silvana Stefani, and Anna Torriero, editors, *Networks, Topology and Dynamics*, volume 613 of *Lecture Notes in Economics and Mathematical Systems*, pages 257–282. Springer Berlin Heidelberg, 2009.

[Rus97] John Rust. Using randomization to break the curse of dimensionality. *Econometrica: Journal of the Econometric Society*, pages 487–516, 1997.

[Rus01] Bertrand Russell. *The problems of philosophy*. OUP Oxford, 2001.

[SBG+11] Mirelle Gettler Summa, Léon Bottou, Bernard Goldfarb, Flonn Murtagh, Catherine Pardoux, and Myriam Touati. *Statistical learning and data science*. Chapman and Hall/CRC, 2011.

[Sch00] Thomas Schreiber. Measuring information transfer. *Physical review letters*, 85(2):461, 2000.

[SFS+09] Frank Schweitzer, Giorgio Fagiolo, Didier Sornette, Fernando Vega-Redondo, Alessandro Vespignani, and Douglas R White. Economic networks: The new challenges. *science*, 325(5939):422–425, 2009.

[SHH09] Mark Stevenson, Yuan Huang, and Linda C Hendry. The development and application of an interactive end-user training tool: part of an implementation strategy for workload control. *Production Planning and Control*, 20(7):622–635, 2009.

[Sim51] Edward H Simpson. The interpretation of interaction in contingency tables. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 238–241, 1951.

[Sin84] Roger W Sinnott. Virtues of the haversine. *Sky and telescope*, 68(2):159, 1984.

[SL08] Matthäus Staniek and Klaus Lehnertz. Symbolic transfer entropy. *Physical Review Letters*, 100(15):158101, 2008.

[SP10] Skipper Seabold and Josef Perktold. Statsmodels: Econometric and statistical modeling with python. In *Proceedings of the 9th Python in Science Conference*, volume 57, page 61, 2010.

[SR10] Georgina Santos and Maël Robin. Determinants of delays at european airports. *Transportation Research Part B: Methodological*, 44(3):392–403, 2010.

[SSGM12] Janina Scheelhaase, Martin Schaefer, Wolfgang Grimme, and Sven Maertens. Cost impacts of the inclusion of air transport into the european emissions trading scheme in the time. *European Journal of Transport and Infrastructure Research*, 12(3), 2012.

[SWL15] Xiaoqian Sun, Sebastian Wandelt, and Florian Linke. Temporal evolution analysis of the european air transportation system: air navigation route network and airport network. *Transportmetrica B: Transport Dynamics*, 3(2):153–168, 2015.

[SWM93] W Nick Street, William H Wolberg, and Olvi L Mangasarian. Nuclear feature extraction for breast tumor diagnosis. In *IS&T/SPIE's Symposium on Electronic Imaging: Science and Technology*, pages 861–870. International Society for Optics and Photonics, 1993.

[SWM05] Michael PH Stumpf, Carsten Wiuf, and Robert M May. Subnets of scale-free networks are not scale-free: sampling properties of networks. *Proceedings of the National Academy of Sciences of the United States of America*, 102(12):4221–4224, 2005.

[Tib96] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[TLW+89] Donald E Thomas, Elizabeth D Lagnese, Robert A Walker, Jayanth V Rajan, Robert L Blackburn, and John A Nestor. *Algorithmic and Register-Transfer Level Synthesis: The System Architect's Workbench: The System Architect's Workbench*, volume 85. Springer Science & Business Media, 1989.

[UAB+10] A Urruticoechea, R Alemany, J Balart, A Villanueva, F Vinals, and G Capella. Recent advances in cancer therapy: an overview. *Current pharmaceutical design*, 16(1):3–10, 2010.

[Ver05] PF Verdes. Assessing causality from multivariate time series. *Physical Review E*, 72(2):026222, 2005.

[VR07] Fernando Vega-Redondo. *Complex social networks*. Number 44. Cambridge University Press, 2007.

[WB04] Maoxin Wu and David E Burstein. Fine needle aspiration. *Cancer investigation*, 22(4):620–628, 2004.

[WBC16] Wisconsin Breat Cancer data set. `http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29`, 2016.

[WC02] Cheng-Lung Wu and Robert E Caves. Towards the optimisation of the schedule reliability of aircraft rotations. *Journal of Air Transport Management*, 8(6):419–426, 2002.

[WCL01] Keith D Wichman, Göran Carlsson, and Lars GV Lindberg. Flight trials: "runway-to-runway" required time of arrival evaluations for time-based atm environment. In *Digital Avionics Systems, 2001. DASC. 20th Conference*, volume 2, pages 7F6–1. IEEE, 2001.

[WDM12] Sean M Wilkinson, Sarah Dunn, and Shu Ma. The vulnerability of the european air traffic network to spatial hazards. *Natural hazards*, 60(3):1027–1036, 2012.

[WH05] Ian Wilson and Florian Hafner. Benefit assessment of using continuous descent approaches at atlanta. In *24th Digital Avionics Systems Conference*, volume 1, pages 2–B. IEEE, 2005.

[Whi00] Halbert White. A reality check for data snooping. *Econometrica*, 68(5):1097–1126, 2000.

[Wie56] Norbert Wiener. The theory of prediction. *Modern mathematics for engineers*, 1:125–139, 1956.

[WMWJ11] Jiaoe Wang, Huihui Mo, Fahui Wang, and Fengjun Jin. Exploring the network structure and nodal centrality of china's air transport network: A complex network approach. *Journal of Transport Geography*, 19(4):712–721, 2011.

[WQd16] Wine Quality data set. `https://archive.ics.uci.edu/ml/datasets/Wine+Quality`, 2016.

[WSC15] Sebastian Wandelt, Xiaoqian Sun, and Xianbin Cao. Computationally efficient attack design for robustness analysis of air transportation networks. *Transportmetrica A: Transport Science*, 11(10):939–966, 2015.

[WT12] Jinn-Tsai Wong and Shy-Chang Tsai. A survival model for flight delay propagation. *Journal of Air Transport Management*, 23:5–11, 2012.

[WTGX06] Bing Wang, Huanwen Tang, Chonghui Guo, and Zhilong Xiu. Entropy optimization of scale-free networks' robustness to random failures. *Physica A: Statistical Mechanics and its Applications*, 363(2):591–596, 2006.

[Wu05] Cheng-Lung Wu. Inherent delays and operational reliability of airline schedules. *Journal of Air Transport Management*, 11(4):273–282, 2005.

[WW12] Hongyong Wang and Ruiying Wen. Analysis of air traffic network of china. In *Control and Decision Conference (CCDC), 2012 24th Chinese*, pages 2400–2403. IEEE, 2012.

[XH08] Zengwang Xu and Robert Harriss. Exploring the structure of the us intercity passenger air transportation network: a weighted complex network approach. *GeoJournal*, 73(2):87–102, 2008.

[Yul03] G Udny Yule. Notes on the theory of association of attributes in statistics. *Biometrika*, 2(2):121–134, 1903.

[YVPN08] Sang Won Yoon, Juan D Velasquez, BK Partridge, and Shimon Y Nof. Transportation security decision support system for emergency response: A training prototype. *Decision Support Systems*, 46(1):139–148, 2008.

[Zan15] Massimiliano Zanin. Can we neglect the multi-layer structure of functional networks? *Physica A: Statistical Mechanics and its Applications*, 430:184–192, 2015.

[Zan16] Massimiliano Zanin. On causality of extreme events. *PeerJ*, 4:e2111, 2016.

[ZBCB08] Massimiliano Zanin, Javier M Buldú, P Cano, and S Boccaletti. Disorder and decision cost in spatial networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 18(2):023103, 2008.

[ZBY16] Massimiliano Zanin, Seddik Belkoura, and Zhu Yanbo. Network analysis of chinese air transport delay propagation. *Chinese Journal of Aeronautics*, in press, 2016.

[ZCDC10] Jun Zhang, Xian-Bin Cao, Wen-Bo Du, and Kai-Quan Cai. Evolution of chinese airport network. *Physica A: Statistical Mechanics and its Applications*, 389(18):3922–3931, 2010.

[ZL13a] Massimiliano Zanin and Fabrizio Lillo. Modelling the air transport with complex networks: A short review. *The European Physical Journal Special Topics*, 215(1):5–21, 2013.

[ZL13b] Massimiliano Zanin and Fabrizio Lillo. Modelling the air transport with complex networks: A short review. *The European Physical Journal Special Topics*, 215(1):5–21, 2013.

[ZM06] Konstantinos G Zografos and Michael A Madas. Development and demonstration of an integrated decision support system for airport performance analysis. *Transportation Research Part C: Emerging Technologies*, 14(1):1–17, 2006.

[ZMW05] Etay Ziv, Manuel Middendorf, and Chris H Wiggins. Information-theoretic approach to network modularity. *Physical Review E*, 71(4):046117, 2005.

[ZN10] Yu Zhang and Nagesh Nayak. Macroscopic tool for measuring delay performance in national airspace system. *Transportation Research Record: Journal of the Transportation Research Board*, (2177):88–97, 2010.

[ZR09] Yahua Zhang and David K Round. The effects of china's airline mergers on prices. *Journal of air Transport management*, 15(6):315–323, 2009.

[ZSM14] Massimiliano Zanin, Pedro A Sousa, and Ernestina Menasalvas. Information content: Assessing meso-scale structures in complex networks. *EPL (Europhysics letters)*, 106(3):30001, 2014.

[Zur92] Jacek M Zurada. *Introduction to artificial neural systems*, volume 8. West St. Paul, 1992.

[ZZRP12] Massimiliano Zanin, Luciano Zunino, Osvaldo A Rosso, and David Papo. Permutation entropy and its main biomedical and econophysics applications: a review. *Entropy*, 14(8):1553–1577, 2012.

[ZZY03] Shichao Zhang, Chengqi Zhang, and Qiang Yang. Data preparation for data mining. *Applied Artificial Intelligence*, 17(5-6):375–381, 2003.