

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/309041951>

Optimal Routing and Assignment for Commercial Formation Flight

Thesis · July 2015

DOI: 10.13140/RG.2.2.27277.74721

CITATIONS

0

READS

91

1 author:



[Thomas E Kent](#)

University of Bristol

5 PUBLICATIONS 29 CITATIONS

SEE PROFILE

Optimal Routing and Assignment for Commercial Formation Flight

Thomas Eliot Kent

A dissertation submitted to the University of Bristol in accordance with the requirements of the degree of Doctor of Philosophy in the Faculty of Engineering,
April 2015

Word count: Approximately 47,000

Abstract

This thesis investigates the notion of fuel-reduction through formation flight for commercial aircraft, addressing the problems of global routing and assignment. A two stage centralised approach is presented, firstly, assuming a reduction in observed cost by flying in formation the routes, including rendezvous and break points, are calculated to minimise a total cost. The interconnected assignment problem then takes a set of flights, their possible formations and corresponding costs and optimally allocates them into a cost-minimising formation fleet.

An analytic geometric approach is used to develop a scalable methodology for the formation routing problem enabling the quick calculation of costs. The rapid evaluation allows the large scale fleet assignment problem to be solved via a Mixed Integer Linear Program in reasonable time. A Transatlantic case study shows possible formation fuel savings against solo flight of around 8.7% and 13.1% for formations up to size two and three respectively. Further case studies of three distinct sets of flights show that encouraging levels of saving can still be achieved by flights with varied distances, geographical locations and formation drag-reduction levels.

For the more complex task of routing through wind, results show that the analytic approach can act as a reasonable estimate to the assignment problem, allowing higher-fidelity and computationally more intensive routing methods to be introduced via a post-process, significantly reducing solve time.

Methods for mitigating the impact of uncertainty in aircraft take-off times are explored, where a state-space approach, solved using value iteration, can provide optimal speed-policies for aircraft to follow for any possible realisation of delay. Additionally portfolio optimisation provides a method for formations to be assigned to simultaneously maximise reward and minimise the associated risk. Finally the calculation of efficient frontiers allows matching of reward to desired levels of risk-aversion.

This thesis is dedicated to Lisa,
for her unwavering love, care and motivation.

Acknowledgements

I would like to begin by expressing my profound gratitude and appreciation to my PhD supervisor Arthur Richards. Without his sustained support, guidance and motivation the work of this thesis would not have been achievable.

This work has been made possible from the generous funding of EPSRC and Airbus, for which I am hugely grateful. The continued level of help, involvement and assistance I have enjoyed from my Industrial supervisor Andy Williams, has greatly enhanced my PhD experience, something I will forever appreciate.

During my PhD I have had the good fortune of working alongside a number of great people, through a combination of mentoring, camaraderie and regular coffee breaks they have helped me through this difficult process.

An immeasurable amount of thanks goes to my mother and father, not only for their enduring love and support, but for making me the person I am today.

Finally and most importantly to Lisa, my best friend and partner, for your consistent belief and encouragement I thank you.

Author's Declaration

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

Signed

Date

Nomenclature

\hat{S}	Stochastic state space
cont_w	Contribution of the wind
A, B, C, D	Points representing airport locations
$C(s, u)$	Cost of applying control u at state s
C_D	Coefficient of drag
C_L	Coefficient of lift
C_t	Thrust specific fuel consumption factor
C_{D_0}	Zero-lift drag coefficient
c_j	Cost of formation j
F_1, F_2	Flights
$J(s)$	Cost-to-go function from state s
j_t	Time-step
K	Induced drag correction factor
m	Gradient of aircraft speeds
n	Size of formation fleet
N_a	Number of aircraft
N_f	Number of favourable formations
n_r	Negative binomial distribution number of successes

p	Negative binomial distribution probability of success
P, Q	Formation rendezvous and break points
$p_{j,i}$	Binary variable if aircraft i is included in formation j
R	Range of aircraft
r	Distance flown by aircraft
R_r	Reachable region
r_A, r_B, r_C	Constrained radial distances of climbing and descending
S	Deterministic state space
s	State
S_F	Formation state space
t	Time
T_A	Thrust available
T_R	Thrust required
t_{co}	Cutoff time
t_{del}	Delay time
U	Set of possible controls
u	Control choice
V	Speed
V_∞	Stream-free velocity
V_G	Ground vector
V_H	Heading vector
V_W	Wind vector
W	Aircraft weight
W_0	Final aircraft weight

W_1	Initial aircraft weight
w_A, w_B, w_C	Solo arc weighting
X_i, Y_j	Back vertices of intercept loci
x_j	Binary decision variable of whether formation j is chosen
α	Wind angle
Δt	Maximum permitted schedule change
δ	Binary state of whether an aircraft is airborne
γ	Contribution of constant aerodynamic terms
$\hat{\lambda}_f$	Formation discount factor percentage
$\lambda_{f,n}$	Formation discount factor
Π	Set of all policies
π	Control policy
π^*	Optimal control policy
ρ_∞	Air density
σ	Standard deviation
τ	Risk tuning factor
$\theta_A, \theta_B, \theta_C$	Angles of intersection
r_σ	Risk level
$\mathbb{E}(s, s')$	Expected value to move between states
$\mathcal{F}(s, s')$	Probability that the other aircraft takes off at next state

Acronyms

ATAG	Air Transport Action Group
ATC	Air Traffic Control
BADA	Base of Aircraft Data
BL	Bottom Left
BR	Bottom Right
CI	Confidence Interval
CSV	Comma Separated Values
DP	Dynamic Programming
EAM	Estimated Assignment Method
GC	Geometric Cost
GR	Geometric Routing
GUI	Graphical User Interface
HGV	Heavy Goods Vehicle
HPC	High Performance Computer
JS	Jet Stream
KML	Keyhole Markup Language
MILP	Mixed Integer Linear Program
NAT	North Atlantic Tracks

NBD	Negative Binomial Distribution
NTP	No Takeoff Policy
OAG	Official Airline Guide
OSM	Open Street Map
PDF	Probability Density Function
SDP	Stochastic Dynamic Programming
TR	Top Right
TL	Top Left
VI	Value Iteration
WR	Wind Routing
WC	Wind Cost
ZFTOW	Zero Fuel Take Off Weight

Contents

1	Introduction	1
1.1	Formation Flight	2
1.1.1	Key Areas of Formation Flight	3
1.1.2	Thesis Objective	4
1.2	Review of Current Literature	5
1.2.1	Aerodynamics of Formation Flight	5
1.2.2	Guidance and Control	8
1.2.3	Routing and Assignment	9
1.2.4	Summary of Literature	10
1.2.5	Implementation of Commercial Formation Flight	10
1.3	Contribution Summary	12
1.4	Associated Publications	13
1.5	Thesis Outline	14
2	A Geometric Method for Optimal Formation Flight Routes	16
2.1	Introduction	16
2.2	The Geometric Method	17
2.2.1	Using Arc Weights to Represent Formation Flight	18
2.2.2	The Fermat Point Problem	19
2.2.3	Extending for Weighted Arcs	20
2.2.4	Loci of Possible Formation Join Points	21
2.2.5	Routes With Distinct Departure and Destination Nodes	24
2.2.6	Incorporating a Minimum Distance to Climb and Descend	25
2.2.7	Extension onto the Sphere	26
2.2.8	Verification	27
2.3	An Extension for Larger Fleet Sizes	28
2.3.1	The Decoupled Problem	28
2.3.2	An Example of Creating Fleets of Size 2 and 3	30

2.4	Modelling Aircraft Fuel-Burn	31
2.4.1	Differential Fuel-Burn Model	31
2.4.1.1	Alternative Forms of Fuel Burn Equation	35
2.4.2	Initial Weight Estimation	35
2.4.3	Differential Fuel-burn Arc-weightings	36
2.5	The Global Fleet Assignment Problem	38
2.5.1	Fleet Assignment Using a Mixed Integer Linear Program	38
2.5.2	The Combinatorial Impact	39
2.6	Case Study: Transatlantic Formation Flights	40
2.6.1	For Formations of up to Two Aircraft	41
2.6.2	For Formations of up to Three Aircraft	41
2.6.3	Comparison of Results	43
2.6.4	Analysis of Aircraft Scheduling on Formation Flight	44
2.6.5	Utilisation of Potential Saving	46
2.7	Summary	46
3	Wind-optimal Routing for Formation Flight	48
3.1	Introduction	48
3.2	Wind Routing Method	49
3.2.1	Generating Static Wind Fields	49
3.2.2	Routing Through Wind Fields	50
3.3	Example Formation	53
3.4	Comparison of Methods: Transatlantic Case Study	54
3.4.1	Method	54
3.4.2	Enumeration and Assignment Workflow	56
3.4.3	Jet-Stream West-to-East	57
3.4.4	Jet-Stream East-to-West	60
3.4.5	More-Volatile Wind	60
3.4.6	Comparison of Results	63
3.5	Improving the Geometric Assignment in the Presence of Wind	63
3.5.1	Estimated Assignment Method	65
3.5.2	Results of Using an Estimated Assignment Method	66
3.6	Summary	71
4	Mitigating the Impact of Ground Delay on Formation Flight	73
4.1	Introduction	73
4.2	Probability Density Functions of Airport Ground Delay	74

4.2.1	Negative Binomial Distributions	75
4.2.2	Modelling Airport Ground Delay	75
4.3	Holding Pattern Approach	76
4.3.1	Holding Pattern Cutoff Time	78
4.3.2	Example Formation	78
4.4	A State Space Approach	81
4.4.1	Formation and Non-Formation States	82
4.4.2	Moving Through the State Space	83
4.4.3	Deterministic and Stochastic Regions of the State Space	85
4.5	The Deterministic Problem: Both Flights Airbourne	86
4.5.1	Problem Formulation	87
4.5.2	Value Iteration	88
4.6	Sampling of the State Space	89
4.6.1	Grid-based Sampling	89
4.6.2	Interpolated Cost Function	90
4.6.3	Quadtree decomposition	92
4.6.4	Interpolation Between Quadtree Sampling	93
4.6.5	Climbing and Descending	96
4.7	Stochastic Dynamic Programming	97
4.7.1	Problem Formulation	98
4.7.2	Sampling of the Stochastic State Space	100
4.7.3	Combining the Deterministic and Stochastic State Spaces	102
4.8	Example Formation	103
4.9	Robust Planning	106
4.9.1	Robust Formation Assignment Using a MILP	107
4.9.2	Portfolio Optimization	109
4.10	Transatlantic Case Study	110
4.10.1	Holding Pattern Approach	110
4.10.2	State Space Approach	112
4.10.3	Comparison of Methods	113
4.10.4	Using a Nominal Speed Policy	114
4.11	Summary	115
5	Formation Flight Case Studies	118
5.1	Introduction	118
5.2	Case Study Data Sets	118

5.2.1	Transatlantic Flights	119
5.2.2	EasyJet European Flights	119
5.2.3	Singapore Airlines Flights	120
5.3	Comparing Airline Network Design	120
5.4	Results for Formation Discount Factor of 10%	123
5.4.1	Overall Average Formation Saving	124
5.4.2	Proportion of Flights Joining Formation	125
5.4.3	Flight Deviations	126
5.4.4	Utilisation	128
5.4.5	Summary	132
5.5	Spread of Results	132
5.5.1	Correlations	134
5.5.2	Cross-correlation	135
5.5.3	Results	135
5.6	Sensitivity to Formation Discount	138
5.6.1	Overall Average Formation Saving	139
5.6.2	Proportion of Flights Joining Formation	140
5.6.3	Flight Deviations	140
5.6.4	Utilisation	142
5.7	Summary	142
6	Door-to-Door Routing For Road and Air	144
6.1	Introduction	144
6.2	Developing a Road, Airport and Flight Network	145
6.2.1	The Road Data	145
6.2.2	The Flight and Airport Data	147
6.3	Routino Routing Software	147
6.3.1	Routino Base-Software	147
6.3.1.1	Routing Profiles	148
6.3.2	Adapting Routino for Road and Air	149
6.3.2.1	Design Considerations	149
6.3.2.2	New Highway Types	150
6.3.2.3	Airport Network Design	150
6.3.2.4	Combining Road and Air Data	152
6.3.3	The Router	153
6.3.3.1	PlanetSplitter	153

6.3.3.2	Routing	154
6.3.3.3	Example Usage	155
6.3.4	Final Route Solution	156
6.4	Workflow	158
6.4.1	Stage 1 - Precomputing	158
6.4.2	Stage 2 - Routing	159
6.4.3	Stage 3 - Input Output	161
6.4.3.1	Stage 3 (a) - GUI	161
6.4.3.2	Stage 3 (b) - Automation	162
6.4.4	Stage 4 - Output and Analysis	163
6.5	Results	164
6.5.1	Where Total Journey Time is Spent	164
6.5.2	Journey Distance Compared to Journey Duration	166
6.5.3	Proportion of Wasted Time	167
6.5.4	Isochrone Map	167
6.6	Summary	171
7	Conclusion	173
7.1	Thesis Summary	173
7.2	Summary of Contributions	177
7.3	Future work	178
7.3.1	Extension of Current Methods and Work	178
7.3.2	Future Applications	179
7.3.3	Extensions of Door-to-Door Routing	180
	Bibliography	182

List of Figures

1.1	Wing-tip vortices and regions of upwash and downwash	3
2.1	A Fermat-Torricelli geometric construction solution	20
2.2	Three point vectorial representation and corresponding angles	21
2.3	Possible solution points given an angle of interception	22
2.4	Join and break points for two distinct routes	23
2.5	Join point location regions	24
2.6	Ensuring radial distances from each airport	26
2.7	Deviations between geometric solution and exhaustive search	28
2.8	Join and break points for three distinct routes	29
2.9	Optimal join and break points for fleet size 2 and 3	32
2.10	Transatlantic formation routes	42
2.11	Deviation in route distance between solo and formation flight	43
2.12	The effect of maximum permitted takeoff changes	45
3.1	Random wind fields for different levels of volatility	50
3.2	Variable way points making up formation route	51
3.3	Wind-triangle and contributing vectors	51
3.4	Wind Route vs Geometric Route	55
3.5	Workflow of methods for wind route calculation and assignment	58
3.6	Wind Routes for Jet-Stream West-to-East	61
3.7	Wind Routes for Jet-Stream East-to-West	62
3.8	Wind Routes for volatile wind field	64
3.9	Estimated assignment method solutions	68
3.10	Estimated assignment method's addition of Wind Costs	70
4.1	Negative Binomial PDFs	76
4.2	Geographical representation of Hold Policy approach	77
4.3	Possible achievable savings for varying cutoff times	79

4.4	Savings and probabilities for LAX-LHR and PHX-LHR	80
4.5	Representation of State Space approach	81
4.6	The two state variables x_1 and x_2 of aircraft location	82
4.7	Regions within the State Space	83
4.8	Solution policy walk-through	84
4.9	Discretisation of State Space for grid-based method	91
4.10	Interpolation of cost-to-go J	91
4.11	Quadtree based sampling method	93
4.12	Node count grid-based vs. quadtree	93
4.13	Types of quadtree nodes needing evaluation - Non-formation	94
4.14	Types of quadtree nodes needing evaluation - Formation	94
4.15	Moving between formation and non-formation states	95
4.16	Including climb and descent within S	97
4.17	Stochastic State Space for the airbourne aircraft	98
4.18	Full SDP and DP policy solution path	100
4.19	Grid-based interpolation for stochastic State Space	101
4.20	The shared variables between S and the two \hat{S} state spaces	103
4.21	No-Takeoff Policy for each flight	105
4.22	Probability distribution of possible savings for example formation	106
4.23	Efficient frontier solution	109
4.24	Expected values for tuning τ	111
4.25	Efficient frontier of formation allocations	112
4.26	Tuning penalty and the corresponding expected savings	113
4.27	Efficient frontier of formation allocations	114
4.28	Efficient frontiers of SDP vs nominal speed policy	115
5.1	Representative graph structures of flight networks	122
5.2	Circular connectivity ordered by airport degrees	122
5.3	Flight network sparsity ordered by longitude	122
5.4	Total average saving	125
5.5	Distribution of formation saving	125
5.6	Proportion of aircraft assigned into a formation	126
5.7	Deviation in distance between solo and formation flight	129
5.8	Proportion of formations who share a common airport	129
5.9	Average proportion of cruise to non-cruise flight	130
5.10	Utilisation of potential formation saving	130

5.11	Spread of cruise proportion vs saving	133
5.12	Spread of solo distance vs saving	133
5.13	Spread of deviation proportion vs saving	133
5.14	Cross-correlation of Transatlantic Case Study	136
5.15	Cross-correlation of EasyJet Case Study	137
5.16	Cross-correlation of Singapore Airlines Case Study	137
5.17	Relation between discount factor and overall saving achieved	139
5.18	Proportion of aircraft assigned into a formation	140
5.19	Deviation in distance between solo and formation flight	141
5.20	Proportion of formations who share a common airport	141
5.21	Utilisation	142
6.1	Airport network and possible flows	152
6.9	City connectivity results	165
6.11	Duration vs distance of journey	168
6.12	Percentage of journey spent waiting vs journey distance	169
6.13	Isochrones of reachability	170

List of Tables

2.1	Proportional formation fuel-burn weight factors	18
2.2	Binomial coefficients for varying problem sizes	40
2.3	Fuel saving for formations of size 2 and 3	43
2.4	Deviation in route distance between solo and formation flight	43
3.1	Case study comparison of savings	63
3.2	Comparison of saving and cpu time for three assignment methods	67
3.3	Comparison of matching assignments for three assignment methods	67
4.1	NBD n_r and p values for four US airports	75
4.2	Percentage of values expected to lie within confidence intervals	107
5.1	Deviations for Transatlantic flights	127
5.2	Deviations for EasyJet flights	127
5.3	Deviations for Singapore Airlines flights	128
5.4	Cruise proportions for Transatlantic flights	131
5.5	Cruise proportions for EasyJet flights	131
5.6	Cruise proportions for Singapore Airlines flights	131
6.1	Definition of flight types	150
6.2	Flight network highway types	151
6.3	Airport network highway types	151
6.4	Example of Routino input flags	154
6.5	Sample of output for Bristol to Glasgow	157

Chapter 1

Introduction

With passenger numbers expected to reach 7.3 Billion by 2034 [1] commercial aviation is constantly looking for ways to cope with increases in demand [2–4] whilst simultaneously trying to mitigate the resulting impact on the environment [5]. Even with the economic recession of 2008-2009 heavily impacting the airline industry, more recent reports [1] indicate growth in passenger demand for air transport returning to the more historic levels of between 5-6% per year. Therefore the focus of much of modern aviation-related research aims to make air travel cheaper, more-efficient and significantly more environmentally friendly.

Fuel prices still remain a significant cost factor for the airline industry; in 2014, with prices averaging \$130 a barrel, jet fuel accounted for around 31% of airline costs. While alternatives to fossil fuels, such as electric or biofuels [6], are feasible, increases to aircraft efficiency are always needed. Even though between 2000-2013 the average fuel per passenger trip (and therefore also kg of CO_2), has decreased by 31% [7], annually, air transport still represents 2% of all man-made CO_2 emissions [1]. The Air Transport Action Group (ATAG) has set a number of key targets, to improve fleet fuel efficiencies by 1.5% per year until 2020, followed by stabilising net emissions from 2020 with carbon-neutral growth in an attempt to reduce net emissions from aviation by 50% by 2050, compared to the levels of 2005 [8].

Allowing growth in demand whilst continuing to reduce emissions will likely require a combination of operational, economic and technological mitigating strategies [9]. While new technologies offer great potential, the ability for them to quickly become part of the global aircraft fleet [10], would largely rely on the introduction of new aircraft. Therefore it is clear that in order to achieve such environmental goals, without radical changes to current aircraft fleets, breakthroughs of operational-concepts are required. The work of this thesis looks at one potential concept for

commercial aviation: formation flight.

While formation flight has long been used within the military, for providing an effective approach for safely moving large numbers of aircraft [11], in addition to communicative and defensive purposes, the act of flying in close-proximity to obtain drag-reduction is an added benefit. Therefore it is of great interest to see how formation flight can be utilised to reduce fuel-burn for commercial aviation.

One of the immediate benefits of formation flight, over other proposed fuel saving methods such as air-to-air refuelling or blended wing body aircraft [12–14], is the relatively minimal change to the current airframes. The majority of today’s commercial airliners can fundamentally observe a reduction in drag from formation flight [15]. Although the possibility of designing new aircraft in the future to take advantage of the aerodynamic benefits of this scenario would be a long term goal, in the short term it would not be a necessity.

1.1 Formation Flight

Scientists have long looked to nature for inspiration; the field of biomimicry is devoted to developing techniques to emulate nature’s strategies. One motivating example is how geese, and other migratory birds, fly in a ‘V’ formation [16–20] when travelling long distances. Key research shows that birds participating in such formations will have significant increases to their range [18,21] while exhibiting lower heart rates [19] during flight compared to flying alone.

The aerodynamic fundamentals behind formation flight for aircraft are fairly well studied [22–26]. As an aircraft flies, the pressure differentials created over the wings’ surface, generates lift. The wake left behind the lifting-wing induces downwash inboard, between the aircraft wingtips, and a corresponding upwash outboard as depicted in Figure 1.1. A trailing aircraft flying through this region of upwash can maintain its flight while operating at a lower apparent angle of attack, that is, a lower angle of attack relative to the horizon. Therefore the aircraft observes a significant reduction in induced drag and as a result a corresponding reduction in fuel burn.

While estimates regarding the level of drag reduction are mixed, they all remain extremely promising [23,24,26–32]. Some flight tests have recorded levels of induced-drag being reduced by as much as 40-60% for the trail aircraft in a formation of two, equating to roughly a 20-30% reduction in total drag. Alongside this, tests have also directly measured fuel-flow reductions, for an aircraft flying in the upwash of another

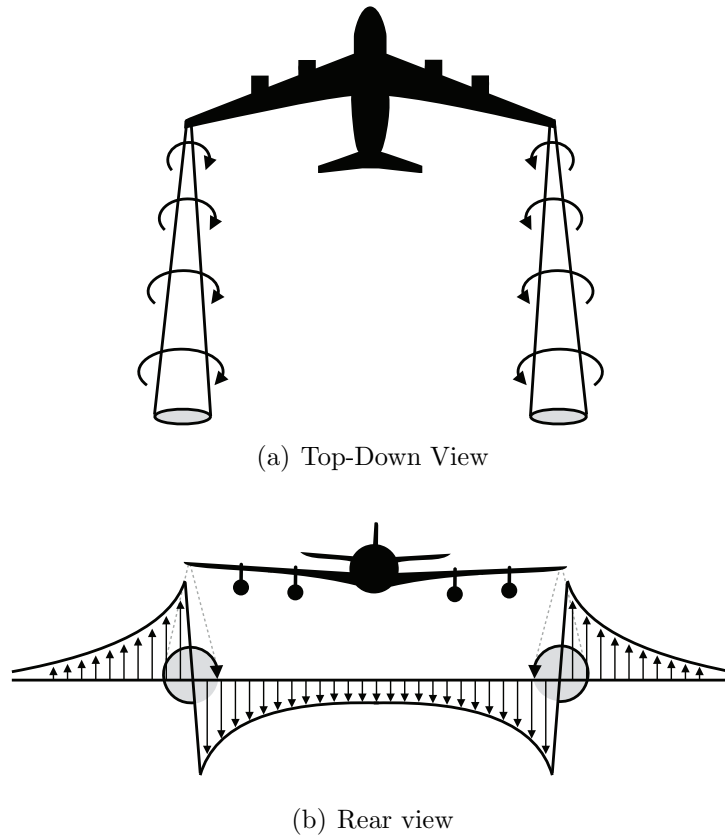


Figure 1.1: Wing-tip vortices and regions of upwash and downwash

aircraft, at levels of anywhere between 4-18%. Most importantly for commercial formation flight, Ning et al. [32] have shown that even flying as much as 10-40 wingspans behind the leading aircraft can still result in a 30% induced-drag reduction.

1.1.1 Key Areas of Formation Flight

The possibility of flying in formation to reduce fuel burn as an alternative to the way commercial flight operates today is an exciting prospect. However, implementation of formation flight is a huge challenge, spanning multiple disciplines and requiring a significant amount of research and investment. Before formation flight can become a realistic fuel-saving method for commercial aviation a number of key areas must first be addressed:

Aerodynamics and Simulation

Through experimentation, flight tests and modelling, a number of core aerodynamic aspects need to be covered. Observing, sensing and tracking wake-vortices is vital. Furthermore modelling and prediction of the aerodynamic

interaction between two (or more) aircraft will assist other areas of research.

Control and Automation

This includes modelling of the aircraft dynamics within a formation, creating suitable control laws to assist aircraft to utilise drag-reduction and determining which aircraft lead and which follow. A key problem will be maintaining optimal aircraft positioning within an aircraft's wake and automating the process to increase efficiency and minimise pilot fatigue.

Routing and Assignment

To take advantage of the fuel reduction flight benefits, aircraft routes need to be altered, calculating rendezvous locations between flights looking to join formation to minimise cost. Importantly the large scale problem of assigning aircraft into formations also needs addressing.

Operations and Regulations

A number of aviation regulations and standards will need to be adjusted, including the reduction of aircraft separation limits, in order to meet the need for the relatively close proximity required for formation flight. Changes to the operational side of commercial flight such as air traffic management and scheduling also need investigating.

1.1.2 Thesis Objective

With a number of studies assessing both the aerodynamic possibility [24, 32, 33] and the associated control problem [34–37] of flying in close proximity in order to reduce drag, coupled with flight tests [30, 31, 38] shows promise that flying in formation can be utilised to reduce fuel-burn. While each of the key areas holds an equal importance, and with all being ongoing research within the scientific community, the work of this thesis chooses to focus on the Routing and Assignment aspects. While some studies show a positive trade off between deviating routes, in order to fly in formation and the reduction in drag it produces [15, 21, 26, 27, 39–41], few have tackled the substantial fleet-assignment problem when routing for formation flight.

The routing side of the problem asks the question of how to route commercial aircraft to take advantage of formation flight. That is, assuming a reduction in observed 'cost' by flying as a formation and given a set of solo routes, how do we go about creating formations and then routing them to minimise a total cost. The interconnected assignment problem is then, given a set of possible solo flights, how

to go about assigning them to particular formation ‘fleets’. The problem in question is highly combinatorial and therefore as the number of flights or size of the fleets increase the possible ways of joining them together grows dramatically. The massively combinatorial nature of this task means that in order to assess sizeable problems a smart approach is needed. Therefore the core objective is broken down into two main, highly connected, questions:

Objective

Optimise current routes to fly in formation to minimise total fuel burn.

Question 1

Given a set of flights, where should the aircraft meet in order to fly in formation?

Question 2

Given the cost of each formation, which of these formations should be used to minimise the total cost?

Finding a globally optimal solution to the above objective is not an easy task, given the problem’s combinatorial nature. Therefore we approach the problem of large scale formation routing initially in a simplified manner where by using a fast geometric approach to gain analytic solutions allows the assignment problem tractability.

The next section outlines and summarises some of the work already carried out in areas relating to formation flight and optimal routing.

1.2 Review of Current Literature

The interest in utilising the formation-induced drag-reduction benefits has grown in popularity over the last ten years. This interest has been motivated by a number of different but interconnected research areas. The breadth of the necessary research to make commercial formation flight viable, means it is beyond the scope to this thesis to assess all these areas. Instead, the work of this thesis is focused on assessing the global routing and assignment problem. The following section outlines some of the key pieces of literature relating to formation flight in the three main areas of Aerodynamics, Guidance and Control and, Routing and Assignment.

1.2.1 Aerodynamics of Formation Flight

Although the aerodynamic aspects of formation flight are not covered by this thesis, underlying influential factors contribute to the practicality and effectiveness of flying

in formation. Results from a number of studies into the aerodynamic benefits of close and extended formation flight have been both inspirational to researchers and essential in developing an understanding of the potential drag-reductions achievable.

The aircraft considered within this thesis are from a set of pre-defined operational models outlined by the Base of Aircraft DAtabase (BADA) [42]. The BADA data set contains details of key operational and performance factors. The data outlines aircraft performance models for a wide range of the more common aircraft types and is broken down into three files. The Operations Aircraft Performance file contains all the thrust, drag and fuel coefficients to be used together with information on weights, speeds, maximum altitude, etc. The Airlines Procedure file defines values such as the default operational climb, cruise and descent speed schedule that is likely to be used by an airline. The Performance File presents the nominal performance of the aircraft model in the form of a look-up table. Having access to this data allows us to not only more accurately incorporate aspects such as speed, flight levels, climb, cruise and descent schedules but to also remain consistent between different aircraft.

The aerodynamic research concerning formation is broad, but a few key themes can be observed. Although formation flight has historically been a military endeavor, with concern being placed on avoiding the negative effects of wake vortices [43], more recent use of wake vortices for drag-reduction has grown in popularity. The 1998 work of Blake and Multhopp [27] highlights the interconnected aerodynamic and control problem. They showed that by flying in close-proximity a large reduction in induced drag could be realised by the trail aircraft; with simulations showing potential range increases of 60% for formations of five aircraft. Notably they emphasised the importance of the lateral-positioning of aircraft, with 50% of drag saving being lost if aircraft were unable to maintain it to within at least one tenth of a wingspan.

Blake and Gringas [26] conducted wind-tunnel tests, and outlined the difference between predicted drag reductions [27] of 40% and those realised at around 25%. These tests also reaffirmed the predicted importance of lateral positioning. Increases in drag occurred if the wing tips overlap by more than half a span, but saving was achieved at overlaps of about 15-20% compared to the 10% predicted.

Analytic modelling and simulation of drag reduction on T-38 aircraft outlined by Wagner et al. [28] showed 60% induced drag reductions at a longitudinal spacing of 0.9 of a wingspan from the leader, with a third aircraft realising a saving of 67%. While the work commended the fact that these savings can fundamentally be realised without any structural changes to the aircraft, attention was drawn to the need for a controller system to relieve the pilots' workload and stay in the optimal position.

This work was preliminary to an actual flight test presented by Wagner et al. [29], which showed that the positional predictions for fuel saving were successful, yielding fuel savings between 4-13%. However the results for a three-aircraft formations were inconclusive, citing the difficulty of flying in a stable position without additional sensing and control abilities.

A number of flight tests have been performed at the NASA Dryden Flight Research Centre to explore the fuel saving benefit of flying in a tight formation. Vachon et al. [30, 31] used two modified F/A-18 aircraft to control and monitor relative position and measure the effects on the performance of the trail aircraft. Total drag reductions of over 20% were calculated and average reductions in fuel flow of 18% were measured. This work also discussed the importance of a need for a control system to maintain position and throttle settings. More recent tests on drag reduction by Pahle et al. [23] on the much larger Boeing C-17 aircraft still show promising savings with fuel flow reductions of 7-8%.

The potential shown by preliminary estimates, backed up by real-world flight tests, is encouraging. Alongside these flight tests a number of analytical studies have been carried out to try to model and predict a number of the major aerodynamic processes at play. The importance of control and automation for positioning and throttle setting was conclusively demonstrated, however the process of designing and implementing the necessary sensing capabilities and control laws will be founded on the ability to model the aerodynamics.

It is essential to be able to predict and track both the evolution of the wake-vortices left by aircraft [33] and the aircraft itself [32], whilst maintaining a position relative to them [22, 44–46]. Furthermore at a design level there is potential [25, 27] for future aircraft to be altered to amplify these drag-saving effects.

One of the main factors of making commercial formation flight a reality is the ability to use the same principle observed in the flight tests of ‘close-formation flight’ and apply them for ‘extended formation flight’. The work of Ning, Flanzer and Kroo [32] shows that induced drag reductions upwards of 30% are possible even at longitudinal separations of 10-40 wing spans. Ning’s Doctoral thesis [24] provides one of the best in depth studies of the potential for formation flight within a commercial environment from an aerodynamic perspective. It covers the modelling of wake vortices for the purposes of extended formation flight with incorporation of the compressibility effects allowing for high-fidelity solutions. Formation topologies are also studied to examine the effect of different positioning within a formation fleet. However, one motivating conclusion from this work is the diminishing rate of formation benefit as

the formation grows in size and how it is coupled to an increasing control complexity. Therefore, for practicality and aerodynamic reasons it may not be feasible to operate large formation fleets within extended formation flight, indicating formations of two or three to be of greatest interest.

1.2.2 Guidance and Control

Many of the concluding remarks of the flight tests emphasised the importance of automation and control within formation flight. Not only to mitigate increases on pilot fatigue but also to realise more of the potential savings. Accordingly, there has been significant research undertaken over the past two decades to try an answer some of the fundamental questions affecting guidance and control within formation flight.

Through the use of decentralized controllers for a formation of five unmanned vehicles Wolfe, Chichka and Speyer [47] explored the idea of having aircraft follow the aircraft in front, analysing of the controller's sensitivity to disturbances in initial conditions. Further work by Chichka, Speyer and Park [48] showed the controller's ability to allow aircraft to quickly return to an optimal position to minimise drag. Similarly the use of distributed control for a formation fleet, studied by Zou et al. [49], allows for the easy addition or removal of aircraft along with an inherent scalability for changing formations sizes. Furthermore the controller includes the ability to adaptively account for an uncertain drag parameter coefficient, meaning changes to wake interaction can be accounted for.

Formation-hold autopilots have been introduced to enable formations to stay in position relative to each other [34, 36, 37, 50, 51]. Pachter, D'Azzo and Proud [36] outlined the use of a controller for the trail aircraft to maintain formation when faced with lead aircraft maneuvers. Furthermore results show that the inclusion of the aerodynamic coupling effects, present during close formation flight, may be unnecessary. Binetti et al. [37] adapts a formation-hold autopilot to incorporate an adaptive loop feedback system with the objective of maximum-efficiency flight while travelling through uncertain wake-induced velocity fields. The use of two extremum-seeking feedback loops controls both the optimal lateral and vertical separation between the aircraft.

DeVries and Paley [33] further explore the need be able to simultaneously track the wake of lead aircraft and control the follower aircraft's position, where applications to both air-to-air refuelling and formation flight are investigated. Optimal control is used to guide follower aircraft to desirable positions relative to the lead with a focus

on encouraging the observability of the parameter states associated with the wake. Furthermore extremum-seeking feedback is used by Brodecki and Subbarao [34] to automatically steer trailing aircraft towards the ‘sweet spot’. Showing that aerodynamic calculation of the optimal positioning of the aircraft may not be a prerequisite but instead ‘sensed’ during flight.

1.2.3 Routing and Assignment

Seminal work on the coupled formation routing and assignment problem was introduced by Ribichini and Frozzoli [41]. They assess the coordination of multiple independent aircraft, with distinct individual routes, to minimise some overall cost. Formation flight between the aircraft is non-mandatory, rather it is a result of the minimum-cost optimization problem. Optimal routes are comprised of great circle paths, ignoring wind, and assume per-unit distance fuel reductions of 10% and 15% when flying in a formation of two and three respectively. An example three flight study resulted in overall savings of 9.2% against solo flight. The work also implements a ‘proposal-marriage’ type greedy-algorithm for decentralised assignment of flights into formations, outlining their concerns with tractability of the larger globally-optimal problem. The centralised approach discussed has been motivational within this area of study and notably to the approach of this thesis.

The work of Bower, Flanzer and Kroo [15] improved upon the basic routing methods with a focus on aerodynamic performance and formation geometries. In addition to the aerodynamic approach the authors outline a very interesting case study for five FedEx flights, creating one formation of two and one of three equating to fuel savings of 7.8% and 12.5%. Notably, the formation induced drag reductions were simulated for each individual aircraft type and differing formation geometries. The in depth aerodynamic simulation within this study provides an excellent candidate for producing ‘higher-fidelity’ solutions for the formation flight problem. Furthermore the results of the case study show potential for even greater savings than the ‘simpler’ approach of Ribichini and Frozzoli.

Succeeding work by Xu, Ning, Bower and Kroo [39, 52] combines route optimization, an aerodynamic model by Ning [32] (previously discussed) and an assignment optimization. The approach taken is to take all possible combinations of missions and heuristically discard ‘bad’ formations, the remaining candidate solutions are then optimised for formation flight. For each route, a gradient-based optimisation is used to calculate values for a number of design variables, such as rendezvous locations

and altitudes and provides reasonably high-fidelity solutions which include a schedule optimisation. The results presented are extremely promising, with the Transatlantic Star-alliance case study consisting of 150 flights and producing a 7.7% overall saving, even with scheduling constraints. However, the extent of the computational time required, roughly 200 hours for 2500 formation combinations, shows the need for a faster approach if the larger global assignment problem is to be fully assessed. The contributions of these works, along with personal discussions between myself and the authors, have provided the basis for the approach taken within this thesis. The conclusion is that the higher-fidelity approach exists but suffers from significant computational requirements. To assess the massively combinatorial approach needed for the global assignment problem, it is clear that a faster method is required.

1.2.4 Summary of Literature

The literature discussed provides motivation for the core aim of this thesis: to address the global routing and assignment problem for commercial formation flight. The flight tests, aerodynamic simulations and analytical study help to not only show the potential for formation flight, but to also quantify it. Work by Blake et al. [26, 27], Wagner et al. [28, 29] and Vachon et al. [30, 31] all show significant drag reduction figures. These numbers help to estimate formation drag-reduction values used within this thesis. Furthermore the work of Ribichini and Frozzoli [41] and more recently Bower et al. [15, 39, 52] provide examples where using such drag-reduction numbers can produce promising results.

Importantly, while the work by Xu, Ning, Bower and Kroo [39, 52] provides an excellent solution methodology for the commercial formation flight problem, it cannot appropriately tackle the massively combinatorial global assignment problem.

1.2.5 Implementation of Commercial Formation Flight

Although each of the core research areas discussed in Section 1.2 will contribute to the overall success of commercial formation flight the work of this thesis is based within the Routing and Assignment category. In order to focus on this particular aspect of the formation flight problem, a number of assumptions have been made. While the flight tests and aerodynamics are a motivation, the process of modelling, predicting and measuring the benefits and practicalities are not explored. Instead these topics are assumed to be sufficiently solvable to allow for formation flight to occur. Aside from the modelling aspects of using the BADA data, estimates of the

potential drag-reduction are seen to most directly impact the work of this thesis.

Similar assumptions need to be made for the guidance and control side of formation flight. Utilisation of the drag-reduction will depend heavily on the ability to design and implement effective control-laws to automate the formation flight process. Again, the work of this thesis assumes that this is a feasible task.

Finally even with all the optimisation, mathematical and engineering problems solved, operations and regulations remain an obstacle. The underlying presumption is that, with enough realistic potential for significant reductions in fuel, emissions and costs coupled with sufficient safety and control measures, that this side of the problem will become solvable.

However, with these in place, it is still important to be able to outline (or at least hypothesise) where formation flight fits within the different phases of commercial flight today. For any flight to operate within a controlled airspace flight plans must first be filed, these detail a number of aspects of the flight such as route and fuel consumption and must comply to safety and Air Traffic Control (ATC) requirements. For airline companies these flight plans are most often submitted months in advance, as schedules are planned and known long before the flights are flown, allowing for Air Traffic Control (ATC) to best plan for busy and congested airspaces. The main three phases of flight planning within ATC are strategic, pre-tactical and tactical [53]. The strategic phase is used to plan for overall traffic predictions and occurs anywhere from months to two days before the flights. The strategic phase is then followed by a pre-tactical phase which occurs over the two days preceding the flight, this allows for some final adjustments to be made and the routes fine-tuned. The last phase is the tactical phase, on the day of the flight where adjustments for air traffic flow are made.

While the problem of formation flight would need to be addressed for each of these three phases, the work of this thesis mainly places itself within the strategic phase. That is, given our main focus of the routing and assignment of formation flights, this processes occurs at least a number of days before the intended execution of the flights. Therefore, given the huge number of flight plans submitted by airline companies, one could imagine that this process of routing and assignment, specifically for formation flight, might take place at an ATC level. Furthermore it is reasonable to imagine that the pre-tactical and tactical phases can be used to further improve the basic formation routes, taking into account a whole range of extra factors.

1.3 Contribution Summary

The work explored throughout this thesis contributes a number of novel ideas in areas of routing and assignment for commercial formation flight. The following is a summary of the key contributions:

A geometric routing methodology for commercial formation flight

A unique adaption of a Fermat-Torricelli mathematical problem to model the formation flight problem. Allowing fast, analytical routing for theoretically any sized formation, whilst including aspects such as minimum climb/descend distances constraining and differential-fuel burn. This approach helps to tackle the impact of the combinatorial nature of routing for all possible formation pairings.

Numerical method for formation routing through wind

Although wind-routing exists within single-aircraft routing and trajectory optimization the process has not yet been explored for commercial formation flight. This thesis uses an active-set approach with a numerical optimizer to find minimum-energy paths through a static wind-field.

An estimated assignment process for improving geometric allocation

A simple adapted assignment process to improve an initial geometric estimated MILP allocation of flights to formation fleets. A cost-estimating function, based on the geometric approach, is used to predict the potential cost for the significantly more complex wind-route.

Two-stage dynamic programming to mitigate effect of ground delay

Two previously unexplored methods are presented to try and mitigate the effect uncertainty in takeoff time has on flights looking to join in formation. The focus is a two-stage dynamic programming formulation to calculate optimal-speed control policies to follow for any possible realisation of delay.

Portfolio analysis for choosing risk-based formation assignments

Taking methods used extensively in economics and finance and applying them to the formation flight assignment problem. Creating formation assignments to incorporate aspects of risk and uncertainty. Efficient frontiers are calculated to directly compare different assignments for the multi-objective optimization of minimising both cost and risk-level.

Analysis of three distinct case studies with formation flight potential

The largest assessment of the global assignment problem currently available. Three distinct flight lists of Transatlantic, EasyJet and Singapore Airlines flights are routed and globally assignment to minimize total fuel burn.

Door-to-Door routing for a road, airport and flight Network

To assess the connectivity of major European cities door-to-door routing software was developed to calculate shortest paths between any two locations using a bespoke road, airport and flight network. This includes a number of key features, such as the ability to handle preferential treatment of road and flight types and also include aspects of airport waiting times such as check-in, taxiing and domestic and international flight transfers.

1.4 Associated Publications

The following papers, authored by myself and my supervisor Dr. Arthur Richards, have been published as result of the work in this thesis.

- T.E. Kent and A.G. Richards. Analytic Approach to Optimal Routing for Commercial Formation Flight. *Journal of Guidance, Control, and Dynamics*, published online March 2015, awaiting print publication.
- T.E. Kent and A.G. Richards. Accounting for the effect of ground delay on commercial formation flight. In 2014 UKACC International Conference on Control, July, 104-109, Loughborough, UK, 2014.
- T.E. Kent and A.G. Richards. On Optimal Routing For Commercial Formation Flight. In AIAA Guidance, Navigation, and Control (GNC) Conference, AIAA Paper 2013- 4889, 1-11, Boston, Massachusetts, August 2013. American Institute of Aeronautics and Astronautics.
- T.E. Kent and A.G. Richards. A Geometric Approach to Optimal Routing for Commercial Formation Flight. In AIAA Guidance, Navigation, and Control (GNC) Conference, AIAA Paper 2012-4769, 1-17, Minneapolis, Minnesota, August 2012. American Institute of Aeronautics and Astronautics.

1.5 Thesis Outline

Chapter 2 proposes a new method for evaluating large numbers of potential formations, based on a simplified geometric solution to the optimization of the rendezvous and break away points. The method outlined is fast, analytic and scalable, while being capable of theoretically solving for any sized formation. The simplified geometric approach allows for millions of formation combinations to be assessed in minutes. The ability to calculate all possible formation combinations in a reasonable time, makes the globally optimal assignment solution process attainable. A case study of 210 Transatlantic flights is first presented, illustrating the potential fuel saving results. Furthermore this case study will also be used consistently within Chapters 3, 4 and 5, allowing directly comparable results.

While the method of Chapter 2 deliberately omits wind effects and assumes great circle flight segments, the focus of Chapter 3 is to introduce formation routing in the presence of wind. The idea will be to compare, and ultimately benchmark, the use of the geometric approach and assignment to calculate an ‘approximate assignment’ to the wind problem. A method for calculating wind-optimal formation routes through a static, predictable wind-field is presented. It is shown that by using the geometric approach to first attain an initial global assignment of flights into formations the more complex routing can conceivably be left to a post-process. Furthermore an estimated assignment approach is used to improve an initial geometric-assignment, in the presence of wind, where results indicate comparable solutions whilst remaining significantly less computationally intensive.

The work of Chapter 4 then focuses on assessing the impact of uncertainty in takeoff time of aircraft looking to join formation. Airport-specific probability density functions quantify the level of probabilistic ground delay while two different methods are introduced to attempt to mitigate its effects. A naïve hold-approach, allowing delay to be easily absorbed at the rendezvous location, is first outlined. The problem is then modelled using a state space approach, whereby a two-stage dynamic programming problem is solved via value-iteration. Assigning optimal control speed-policies for aircraft to follow en-route for any realisable level of delay. Finally through the use of portfolio analysis, various formation assignment solutions are readily compared for the multi-objective problem of simultaneous minimising cost and the associated risk.

Three distinct case studies are then presented in Chapter 5 to try and assess the potential of commercial formation flight on a global scale for formations of size two. The Transatlantic case study used in Chapters 2, 3 and 4 will be compared against

a list of 1300 European EasyJet flights and a list of 230 Singapore Airlines flights. Analysis of results for each case study will be used to predict distinguishing features required for a formation flight to be successful.

The work undertaken out during a six-month industrial placement at Airbus is presented in Chapter 6, where a door-to-door journey planner for a road, airport and flight network is developed to calculate the shortest routes between any two locations, including the handling of airport waiting times such as check-in and taxiing. Open source data is used and combined with adapted open source software to develop a method capable of analysing a huge number of city pairs on a large global-network, with an aim of assessing the connectivity of cities within Europe in line with an 4 hour reachability target.

Finally Chapter 7 summarises the key findings and conclusions of this thesis, with discussion of future research questions and possible extensions to the current work.

Chapter 2

A Geometric Method for Optimal Formation Flight Routes

2.1 Introduction

This chapter focuses on the core motivation of this thesis and a key question in the deployment of formation flight: *which flights should join in formation?* To answer this question, two complex and interconnected problems must be solved:

1. The routing problem: estimate the rendezvous and breakaway points for all possible formation groupings and thus the fuel use for each;
2. The assignment problem: select a compatible set of formations from those considered to achieve minimum global fuel use.

Studies based on individual pairings, looking only at the routing problem, have shown potential positive trade-offs between the diversion to join formation and the reduction in drag formation flight produces [21, 26, 27, 39, 40, 52]. Formation routes for five aircraft studied by Bower et al. [15] focus on aerodynamic aspects such as wing-tip separation. This, along with a case study, shows significant fuel saving potential even with heuristically chosen routes. These prior works adopt numerical trajectory optimization techniques [54–56] to calculate high fidelity solutions to the routing problem. Once the routing problem is solved, evaluating the cost for each potential formation, the subsequent assignment problem is readily solved by a discrete optimization [52].

The challenge in considering larger numbers of aircraft is the growth in the number of potential formations to be evaluated. For example, the 210-flight Transatlantic case

study of Section 2.6 has roughly 22,000 possible pairings, and larger possible formations introduce even more combinations. Xu et al. [52] used a detailed trajectory optimization for a scenario involving 150 aircraft, but introduced a heuristic filtering stage to reduce the number of formations considered. An alternative approach, adopted in this chapter, is to simplify the routing problem. Frazzoli et al. [41] achieved this by assuming constant fuel-burn rates and modelling the problem as a graph search over possible rendezvous and breakaway points, showing potential savings for an example with three aircraft.

The main focus of this chapter is to propose a new method for evaluating large numbers of potential formations, based on a simplified geometric approach solution to the optimization of the rendezvous and break away points. To achieve rapid solutions, the method deliberately omits schedule and wind effects and assumes great circle flight segments. Once the most promising formations have been identified, these effects can be reintroduced in higher fidelity optimization [52,57]. While this chapter assumes fuel burn as a cost, Section 2.6 considers the impact on timing and introduces scheduling constraints.

Section 2.2 begins by outlining the geometric approach to finding time-free optimal routes with few constraints. An adaptation of the well known Breguet range equation [58] is introduced in Section 2.4 where it is used to calculate both the total fuel burnt and also the rate at which the fuel-burn changes during a flight. Section 2.3 describes the method for extending the geometric approach to generate optimal routes for larger formations. The costs of these formations can then be used within the ‘assignment problem’ of Section 2.5 to obtain a globally optimal allocation of aircraft into formations. Finally, a case study of 210 Transatlantic flights is presented in Section 2.6 to illustrate all the methods of this chapter and the results are compared.

2.2 The Geometric Method

This section introduces the route optimization method for a simplified case. Begin by assuming no airspace restrictions, constant altitude, and constant rate of fuel burn per unit distance. Two flights, Flight A and Flight B , fly from two distinct airports, A and B , to a common destination airport, C . Under the assumptions, the optimal flight will consist of two solo straight-line legs, from A and B respectively, to a common join point P , then a shared leg from P to the destination airport C . The formation routing problem is then defined as finding the point P joining A , B and C together, such that the sum of the fuel burnt is minimized. Extensions for distinct destinations,

a spherical Earth and differential fuel-burn will be addressed in Sections 2.2.5, 2.2.7 and 2.4 respectively. Also, although we have defined points A, B and C as airports, they could conceivably be defined as entry or exit points to airways or NAT tracks.

2.2.1 Using Arc Weights to Represent Formation Flight

A notion of ‘arc weighting’ will be used to incorporate the concept of the drag reduction benefits arising from flying in formation. Eurocontrol’s Base of Aircraft Data [42] (BADA) outlines detailed operational and performance factors. The data contains aircraft performance models for a wide range of common aircraft types. By only looking to create formations during cruise, the climb and descent section of the flight can be considered ‘sunk costs’, as they are carried out irrespective of any formation. A constant nominal fuel-burn rate can then be taken directly from BADA, for each particular aircraft, representing a per-distance fuel-burn rate. This fuel-burn constant is then used as the arc-weightings for each flight, to take into account distinct aircraft types and corresponding differing rates of fuel-burn.

The proportion of fuel used along the formation arc of the flight should however, be less than if the aircraft were not in formation. Studies by Ray et al. [31] and Bower et al. [15] expect very reasonable drag savings (and thus a relative reduction in fuel-burn) for aircraft flying in the up-wash of other formation members. The control and distribution of the formation, *e.g.* leader selection, is assumed to be determined separately, and only an aggregate fuel burn rate for the whole formation is used for route optimization. Ideally, this would be based on a detailed consideration of the aircraft types involved. However, this would be a significant piece of work in its own right and is beyond the scope of this thesis. For the purposes of this work, average formation fuel-burn factors are estimated as shown in Table 2.1 using results from Ref. [21, 26, 27, 40] for varying fleet sizes. For example, if the front aircraft receives no saving while the follower saves 20%, the resulting average is taken to be $(1 + 0.8)/2 = 0.9 = \lambda_{f,2}$ relative to both aircraft in solo flight. The method will readily extend to a more detailed determination of this factor based on formations of particular types of aircraft. While initially, the formation discount will apply directly to distances flown, in Section 2.4 the discount will be applied to the drag instead.

Table 2.1: Weight factors simulating proportional formation fuel-burn distance for size n fleets

Size of fleet (n)	1	2	3	4	5	6
Fuel use factor per fleet member ($\lambda_{f,n}$)	1	0.9	0.85	0.82	0.8	0.785

In terms of scalar arc weighting this means that at the formation stage of the flight, for n members in the fleet, each member contributes the proportion $\lambda_{f,n}$ (from Table 2.1) of their own weighting and the total estimated fuel-burn per unit distance on the formation arc is simply the sum of all these contributions.

For the two flights, Flight A and Flight B , leaving airports A and B , travelling to a common destination C and wanting to join in formation via some point P . Let the solo arcs AP and BP have arc-weightings of w_A and w_B respectively (taken from BADA). The fleet has a size of $n = 2$ and so the weight of the formation arc PC is

$$w_C = (w_A + w_B) \times \lambda_{f,2} = (w_A + w_B) \times 0.9.$$

With this in mind, the problem is then to find the optimal location for this point P . The following sections look to use an adaptation of the Fermat Point problem to solve this.

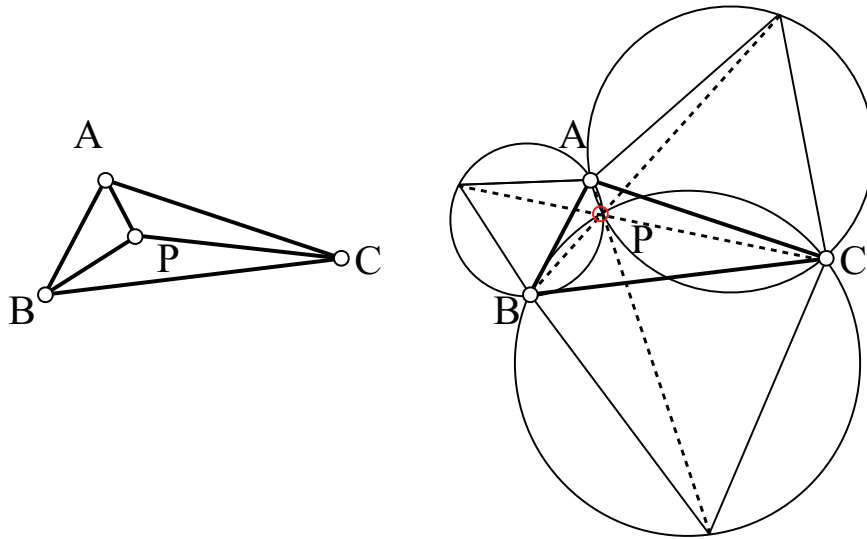
2.2.2 The Fermat Point Problem

The Fermat point problem [59, 60] is a classical mathematical problem posed in the late 17th century, it states:

For a given triangle, ABC , on the Euclidean plane, find a point P such that the sum of the distances $\|\mathbf{PA}\|$, $\|\mathbf{PB}\|$ and $\|\mathbf{PC}\|$ is minimized.

This is in fact equivalent to the formation problem if the weights w_A , w_B , w_C are all equal. Over the years, mathematicians have posed numerous ways of finding this point P , including derivative based methods, the use of mechanics and Fermat's elegant geometric solution. This section reviews an adaptation of the original approach, first proposed via a series of letters between the mathematicians Fermat and Torricelli [59, 60], creating a solution based on the geometric dualities of triangles and circles.

Take a triangle ABC and construct outwardly three equilateral triangles along, and with side lengths corresponding to, the arcs AB , BC and CA as in Figure 2.1. Then the lines from the outer vertex of each new triangle to its opposite vertex of the original intersect at a single point (Figure 2.1(b)). This intersection is the desired point P which minimizes the sum $\|\mathbf{PA}\| + \|\mathbf{PB}\| + \|\mathbf{PC}\|$ (sufficiencies ensuring certain types of solution are explored by Shen et al. [61]). An analogous result can also be observed by constructing the corresponding circumscribed circles of each of these three new equilateral triangles, creating a concurrency at the same optimal point P . Mathematical proofs for Fermat point problems of this type (both planar



(a) Triangle ABC with possible join point P (b) Circumscribed circles and subtending lines concurrent at an optimal point P

Figure 2.1: A Fermat-Torricelli geometric construction solution

and spherical) are fairly abundant: for a deeper understanding of these available methods the author invites you to read Refs. [62–66].

One notable observation is the angles at which these arcs intersect [67] $\angle APB$, $\angle BPC$ and $\angle CPA$ are all 120° . This result holds true with many studies of minimization observed in nature. For example the hexagonal structure of a honeycomb [68], minimal surfaces in soap film experiments [69, 70] and even molecular arrangements [71, 72] all exhibit 120° angles.

2.2.3 Extending for Weighted Arcs

With the notion of weighted arcs, representing differing costs per unit distance, the Fermat point problem can now be extended. For three vertices A , B and C and the join point P , the scalar weights w_A , w_B and w_C correspond to the arcs PA , PB and PC respectively. The problem is then minimizing the sum of the weighted distances:

$$f(P) = w_A \|\mathbf{PA}\| + w_B \|\mathbf{PB}\| + w_C \|\mathbf{PC}\|. \quad (2.1)$$

An analogy to this vectorial equation is to imagine a table with three holes, representing the locations of the points A , B and C . Then at each of the holes a massless, frictionless string is passed through and the corresponding weight is tied to one end. The remaining ends of these three strings are tied into a single knot. This system has

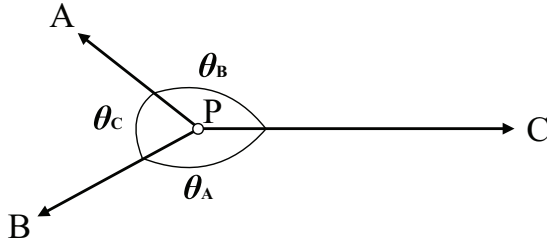


Figure 2.2: Three point vectorial representation and corresponding angles

a natural mechanical equilibrium and this analogy coupled with the minimal energy principle [59] implies that the location of the knot on the table at the mechanical equilibrium is identical to that which minimizes Equation (2.1).

Therefore adapting the ABC triangle of Figure 2.1(a) for weighted arcs leads to a vectorial equilibrium about the point P as in Figure 2.2 such that

$$w_A \frac{\mathbf{PA}}{\|\mathbf{PA}\|} + w_B \frac{\mathbf{PB}}{\|\mathbf{PB}\|} + w_C \frac{\mathbf{PC}}{\|\mathbf{PC}\|} = 0. \quad (2.2)$$

The law of cosines applied to Equation (2.2) leads to expressions θ_A , θ_B and θ_C for the intersection angles $\angle BPC$, $\angle APC$ and $\angle APB$ respectively.

$$\begin{aligned} \theta_A &= \cos^{-1} \left(\frac{-w_B^2 - w_C^2 + w_A^2}{2w_B w_C} \right), \\ \theta_B &= \cos^{-1} \left(\frac{-w_A^2 - w_C^2 + w_B^2}{2w_A w_C} \right), \\ \theta_C &= \cos^{-1} \left(\frac{-w_A^2 - w_B^2 + w_C^2}{2w_A w_B} \right). \end{aligned} \quad (2.3)$$

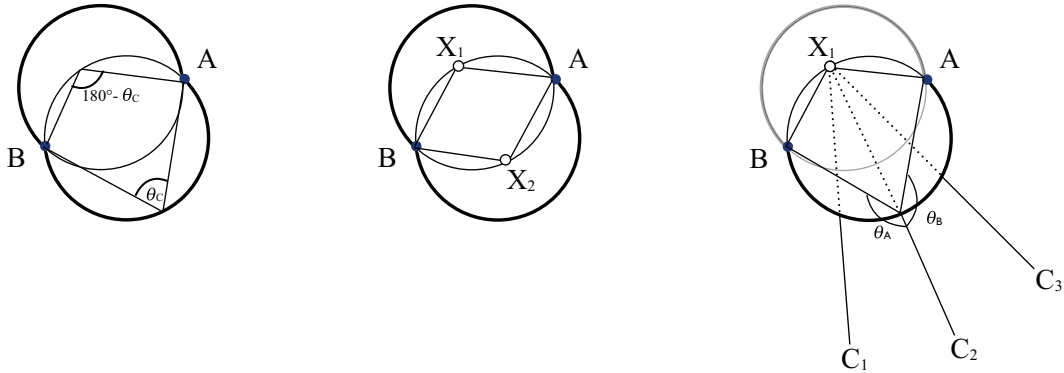
It is important to note that these expressions are obtained solely from the input of the three scalar weight values w_A , w_B and w_C and therefore a priori of any physical location [59].

2.2.4 Loci of Possible Formation Join Points

It has been show that given only the three arc weightings, the specific angles of interception θ_A , θ_B and what will be referred to as the ‘formation angle’, θ_C (*i.e.* the angle between the two solo legs of the flight), can be calculated. Knowing these angles eliminates the need for a fixed destination vertex, C . Two fixed points A and B and a formation angle θ_C , at which the trajectories must meet, describes the loci of possible formation points (the dark lines in Figure 2.3) for all possible destinations.

Furthermore, from the formation angle, two circles with A and B on their perimeter can be constructed. Each of the circles is comprised of two parts: the first contains, on its boundary, all the points P such that $\angle APB = \theta_C$, *i.e.* they meet at the angle required by Equation (2.3), the other, all the points that meet at $180^\circ - \theta_C$ as in Figure 2.3(a).

A similar approach to Torricelli’s in the original Fermat problem, of constructing equilateral triangles on the sides of the ABC triangle, can also be used here. Firstly along the arc AB two similar triangles ABX_1 and ABX_2 can be constructed as in Figure 2.3(b). The side lengths of these two triangles will be in the same proportions as the weights [59]. That is the ratios $w_A : w_B : w_C$, $\|AX_1\| : \|BX_1\| : \|AB\|$ and $\|AX_2\| : \|BX_2\| : \|AB\|$ are equivalent. As the length AB is already known the other two sides can easily be calculated. This generates two ‘back vertices’ X_1, X_2 as in Figure 2.3(b). Note also that the two circles, ‘inscribed’ by A, B and θ_C in Figure 2.3(a) are in fact the same circles which also circumscribe the triangle ABX_i (*i.e.* the circle passes through all three points A, B and X_i) in Figure 2.3(a).



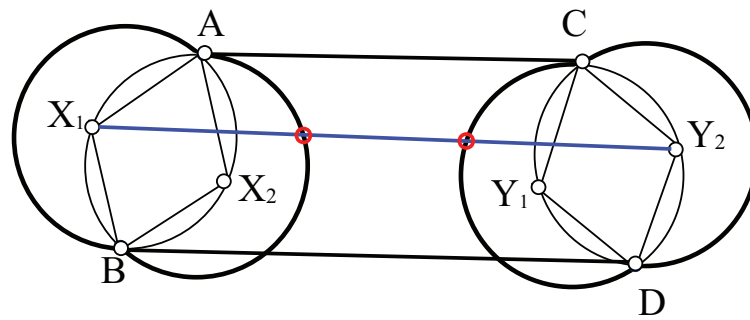
(a) Inscribed loci of possible (b) Back vertices of optimal (c) The final optimal route con-
 formation points given θ_C trajectory ensure all three in-
 tercept angles are satisfied destination.

Figure 2.3: Possible solution points given an angle of interception

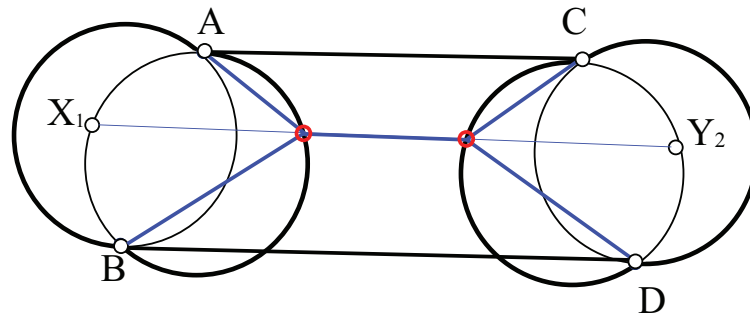
Therefore given any pair of nodes $\{A, B\}$ with three arc-weights w_A, w_B and w_C , two back vertices can be constructed along with the corresponding loci of possible formation points for any destination. Then for any destination C , the formation join point must lie at the intersection of the line CX and the locus arc of possible join points (at most only one of the back vertices will be used, with the choice depending on the location of the destination node).



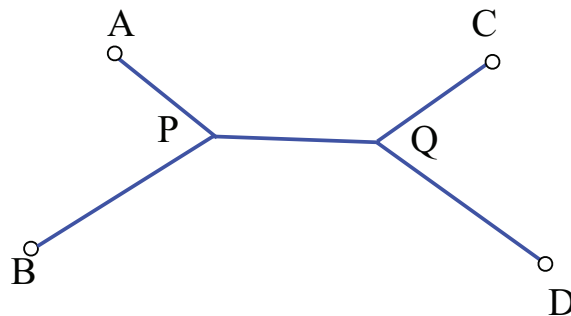
(a) Solo routes



(b) Trajectory between back vertices crossing loci of possible optimal points



(c) Trajectory intercept of loci



(d) Optimal route

Figure 2.4: Join and break points for two distinct routes

2.2.5 Routes With Distinct Departure and Destination Nodes

Knowing the loci of possible join points a priori of a destination allows the assessment of the more general problem of two routes with distinct departure and destination nodes. Where the problem is finding not only a rendezvous location for optimal formation flight, but also the point at which a formation should break away.

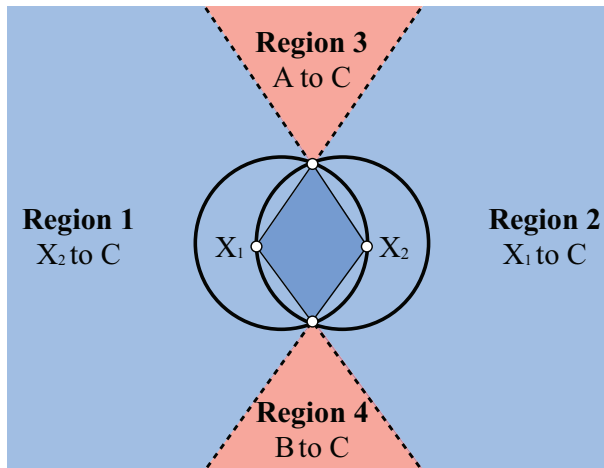


Figure 2.5: Snapshot of regions where a destination node C can be located and the corresponding connecting path.

Given two solo routes between AC and BD (Figure 2.4(a)), first the circles and back vertices are calculated for each pair $\{A, B\}$ and $\{C, D\}$. Then, as in Figure 2.4(b), the arc joining a back vertex X_i of $\{A, B\}$ to a back vertex Y_j of $\{C, D\}$ ($i, j \in \{1, 2\}$) should cross both circles at the required angles (Figure 2.4(c)). Which would result in two crossing points, P and Q , which are the respective join and break points of the formation (Figure 2.4(d)). However, If no single arc exists that satisfies the angles of Equation (2.3) on both circles, then the optimal path is the shortest path between either X_i and C or D , or Y_i and A or B so that the angles are satisfied only once. If that is not possible then the formation arc will simply connect A or B to either C or D . This is referred to as being ‘caught’ at an airport and in most scenarios is undesirable as aircraft need room to climb to an appropriate altitude. A solution to this however, is outlined in the following section. The departure side for each of these cases is outlined in Figure 2.5. For a given destination node C (which could also represent a back vertex), its location will be in one of four regions. Regions 1 and 2 outline when a back vertex is used while Regions 3 and 4 are when the route is instead ‘caught’ at an airport.

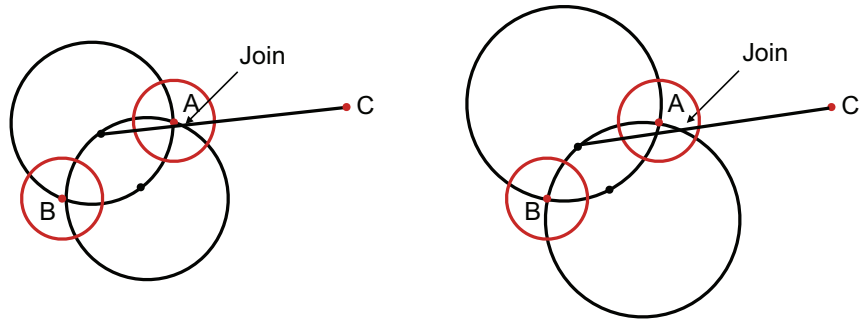
2.2.6 Incorporating a Minimum Distance to Climb and Descend

Some of our early results indicated that many join points were either at the airports themselves, as they were being ‘caught’ (as the destination was in Region 3/4 of Figure 2.5), or very close to an airport. Although this seems like a reasonable result, practicality issues could likely prohibit such a route. In this scenario, flights would either need to already be at a cruising altitude or else take off in formation (possibly on a parallel runway which would rule out many airports) and then engage in a series of step climbs in formation until they reached a cruising altitude. The implications of this, along with the difficulty of achieving formation drag savings along the way, meant the decision was made to only look at joining formations once aircraft are at a cruising altitude.

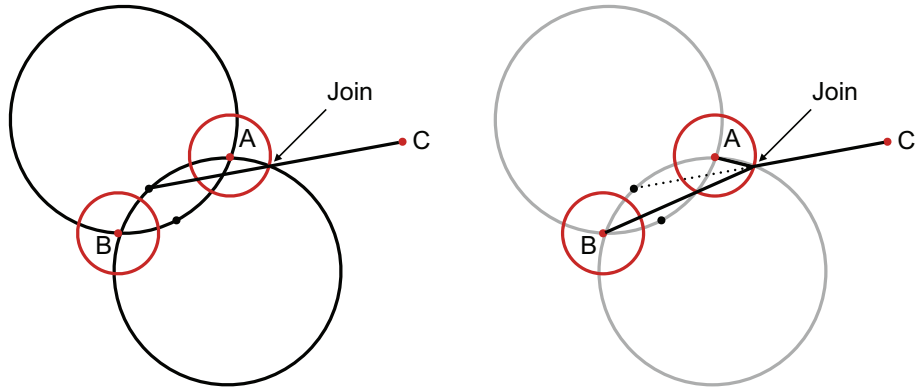
Horizontal distances between takeoff and an altitude at which formations can be joined (similarly a formation altitude and landing) are calculated for each flight. Realistic rates of climb and descent for any given aircraft can be taken from BADA, allowing the calculation of these radial distances. The distances then define the radii, r_A , r_B and r_C , of circular regions around the airports A , B and C respectively. If the optimal formation point lies within any of those regions (as in Figure 2.6), it should not be used and instead be moved onto the region’s perimeter. The problem is still to minimise Equation (2.1) but subject to $\|\mathbf{PA}\| \geq r_A$, $\|\mathbf{PB}\| \geq r_B$ and $\|\mathbf{PC}\| \geq r_C$. This constrained minimisation problem can still be solved using similar geometric methods outlined in this chapter. Take the example of Figure 2.6, whereby the intersection is too close to node A and therefore violates $\|\mathbf{PA}\| \geq r_A$. If P is to be moved onto the perimeter, the first point at which it can do so is when $\|\mathbf{PA}\| = r_A$. Substituting this into Equation (2.1) results in

$$f(P) = w_A r_A + w_B \|\mathbf{PB}\| + w_C \|\mathbf{PC}\|. \quad (2.4)$$

Therefore when minimising $f(P)$, $w_A r_A$ can be considered a constant, being independent of the location of P and so any choice of w_A will result in an analogous minimisation problem. While the relations of Equation (2.3) still hold for the constrained problem (as they are true a priori of location and therefore any distances), they will not necessarily be satisfied by points on the radial circles. Therefore in order to satisfy both the angles equation and the radial constraints the weights must be adjusted. Adjusting w_A does not effect the minimisation problem and therefore the constrained problem becomes the problem of picking the smallest w_A so that the



(a) Initial attempt too close to airport A (b) Decrease w_A and update back vertices and circles - better, but join still too close



(c) Decrease w_A and update back vertices and circles - join sufficiently far away (d) New formation meets at required distance away

Figure 2.6: New join point required to be at least a certain distance from each airport

angles of Equation (2.3) and the radial constraints are both satisfied. In the absence of an entirely analytic solution to finding the necessary value of w_A , a simple bisection search can be used. Given an interval for w_A to be in, w_A is predicted and then the resulting P is found, the interval is then reduced until $\|\mathbf{PA}\| = r_A$. This process is shown in Figures 2.6(c) and 2.6(d). In line with the table and weight analogy of Section 2.2.3, if P is too close to node A , one can imagine slowly reducing the hanging weight w_A until P is sufficiently far enough away.

2.2.7 Extension onto the Sphere

An important note is that the original Fermat Problem, and adaptations described in this chapter, have been inherently planar. As such any planar solutions for routing for

formation flight will not be optimal on the globe. The properties of a curved surface mean it is impossible to find a 2D Earth projection system which is isometric [73] (*i.e.* preserves both angles and distances). The weighted Fermat point problem has been extended to surfaces [62–64, 66] and to even higher dimensions [65]. Most notably Zachos et al. [62] proves Equation (2.3) holds for the problem on the sphere.

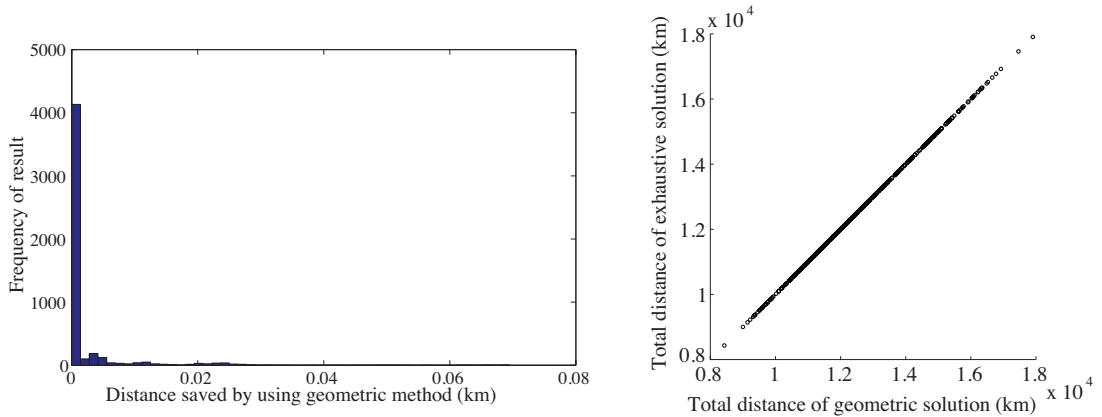
Therefore it is possible to take the Earth to be spherical (with points constrained to its surface) and translate our method for use in spherical coordinates, by increasing the dimension of each element of the method. Straight lines become planes, intersecting the Earth through its center, creating great circle paths. Inscribed circles become inscribed spheres which, as we are constrained to the Earth’s surface, intersect the Earth along a planar surface known as a ‘small circle’. Each one of these small circles will contain a back vertex, two nodes and a loci of formation points all of which will be coplanar. Therefore the original two-dimensional problem is translated to the three-dimensional coordinates of this small circle. Where the plane, defined by the great circle path between the back vertex and the destination, intersects the small circle determines the optimal formation point. This provides an analogous solution on the sphere while preserving the angles of intersection [62] and previously outlined methods. It is somewhat intuitive that the angles will be the same as the 2D case: as an ever smaller region around the join is considered, the sphere appears flatter and the great circles appear straighter, but the angles between them remain the same.

The transition to the spherical problem also enables the more appropriate distance calculation using great circle paths. In general, commercial flights do not fly completely great circle paths due to a number of factors, notably the effects of wind and weather. Although this chapter assumes only great circle paths, the problem of more complex routing such as routing through winds is explored in Chapter 3. The approaches of this chapter were used to first solve the assignment problem (as in Section 2.6, then the much smaller subset of formations is post-processed to take into account the more complex wind-routing problem, allowing a balance of tractability and realism.

2.2.8 Verification

An exhaustive search for 5000 random pairings of solo routes has been used to verify the spherical geometric method. For each pairing all possible join points on a discrete grid (increments of 0.01 degrees of latitude and longitude) are calculated and one with the lowest cost is taken. Figure 2.7 shows the difference in total formation distance

of the geometric solution against the brute force approach. Figure 2.7(a) shows the frequency of a difference in solution. There are no instances of the geometric method giving a worse result and it is clear that the geometric method accurately finds the optimal point of formation, whilst taking a fraction of the time.



(a) Frequency in deviations between geometric solution and exhaustive search at 0.01 degree increments (b) Distances of geometric solution against exhaustive search

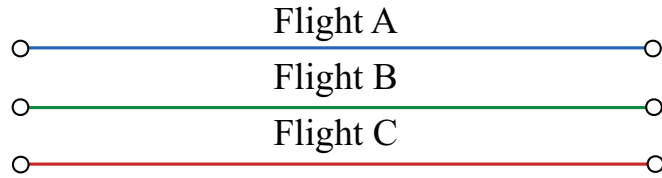
Figure 2.7: Deviations in results between geometric solution and exhaustive search at 0.01 degree increments

2.3 An Extension for Larger Fleet Sizes

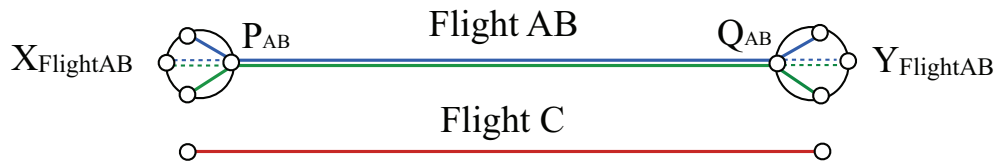
2.3.1 The Decoupled Problem

The framework outlined in Section 2.2 is a powerful result, allowing the routing problem to be decoupled, reducing pairs of nodes to their back vertices and inscribed circles. The optimal route for any formation appears to project from a back vertex regardless of destination. As this information is independent of the destination it depends only on the relative weights and fixed pairs of nodes. This fairly elegant method of projecting from a back vertex can be further extended to not only solve for formations of 2 aircraft, but theoretically any size.

Given the three flights, Flight A, Flight B and Flight C (as in Figure 2.8(a)), first take two of them, for example Flight A and Flight B. Then, by finding the back vertices $X_{\text{Flight AB}}$, $Y_{\text{Flight AB}}$ whose arc crosses at the required angles, a ‘virtual’ Flight AB can then be created. The projected route is going from $X_{\text{Flight AB}}$ to $Y_{\text{Flight AB}}$ and is shown in Figure 2.8(b). The third route Flight C can now be added. This is done just as before, only the arc weightings need to be updated to take into



(a) Solo routes Flight A, Flight B and Flight C



(b) Create projected 'virtual-route' Flight AB between back vertices $X_{\text{Flight AB}}$ and $Y_{\text{Flight AB}}$



(c) Join Flight AB with Flight C via new back vertices $X_{\text{Flight ABC}}$ and $Y_{\text{Flight ABC}}$



(d) Update Flight AB join points given the future join to give final route

Figure 2.8: Join and break points for three distinct routes

account the new size of the formation at each stage of the route. That is, as Flight AB contains two aircraft it will burn fuel at a combined rate of

$$w_{\text{Flight AB}} = (w_{\text{Flight A}} + w_{\text{Flight B}}) \times \lambda_{f,2},$$

while Flight C is weighted at $w_{\text{Flight C}}$. The final formation weighting will then be

$$w_{\text{Flight ABC}} = (w_{\text{Flight A}} + w_{\text{Flight B}} + w_{\text{Flight C}}) \times \lambda_{f,3}.$$

The augmented problem is then solved where Flight AB and Flight C should join (the point P_{ABC}) and break away (the point Q_{ABC}) as in Figure 2.8(c), with the updated weightings $w_{\text{Flight AB}}$, $w_{\text{Flight C}}$ and $w_{\text{Flight ABC}}$. All that is left is to split Flight AB back to two separate flights and update the points P_{AB} and Q_{AB} based on their respective new destinations P_{ABC} and Q_{ABC} as Figure 2.8(d) outlines.

Figures 2.8(a) to 2.8(d) depict the case where Flight A joins Flight B then Flight AB joins Flight C, and breaks away in a similar way. However, realistically it is also necessary to find the order of joining that minimizes total fuel-burn for all flights. Therefore the various combinations of the order of joining formation, including scenarios whereby it might be optimal for only two routes to join while one flies solo, must be also be computed and then the minimum is taken.

2.3.2 An Example of Creating Fleets of Size 2 and 3

For any 3 distinct routes, formations of size 2 can be made in three different combinations, each with its own cost. When trying to find fleets of size 3 there are an additional nine combinations, consisting of 2 choices from 3, one for the join-up and one for the break-away. For example, looking at one of the combinations, given the two routes (both flown by an Airbus A340-300):

$$\text{Flight A} = \{\text{Atlanta, Barcelona}\}, \text{Flight B} = \{\text{Cincinnati, Frankfurt}\},$$

and by using the above methodology with weight values from Table 2.1 results in the desired points for the formation flight. Figure 2.9(a) shows the formation of Flight A and Flight B. The total solo great circle distance, and therefore distance at which fuel is burnt over, for Flight A and Flight B is 14,359 kmeq, where 1 kmeq is the equivalent fuel burnt by an aircraft flying solo for 1 km. When flown in formation the fuel-burn (the kmeq covered using the discounted fuel-burn rates) is reduced by

737 kmeq to 13,622 kmeq. This equates to a saving of roughly 5.1%.

By adding a third flight:

$$\text{Flight C} = \{\text{Miami, Zurich}\},$$

and following the previously outlined method and evaluating all combinations, the optimal ordering of join and break points and their respective locations (as shown in Figure 2.9(b)) can be found. The order of joining formation is: Flight A joins Flight C, then the virtual Flight AC joins Flight B. Flight ABC then flies over the Atlantic in formation followed by Flight A first breaking away, leaving Flight BC, then Flight B and Flight C split to fly to their respective destinations. The resulting total saving is about 8.4% against all three flying solo.

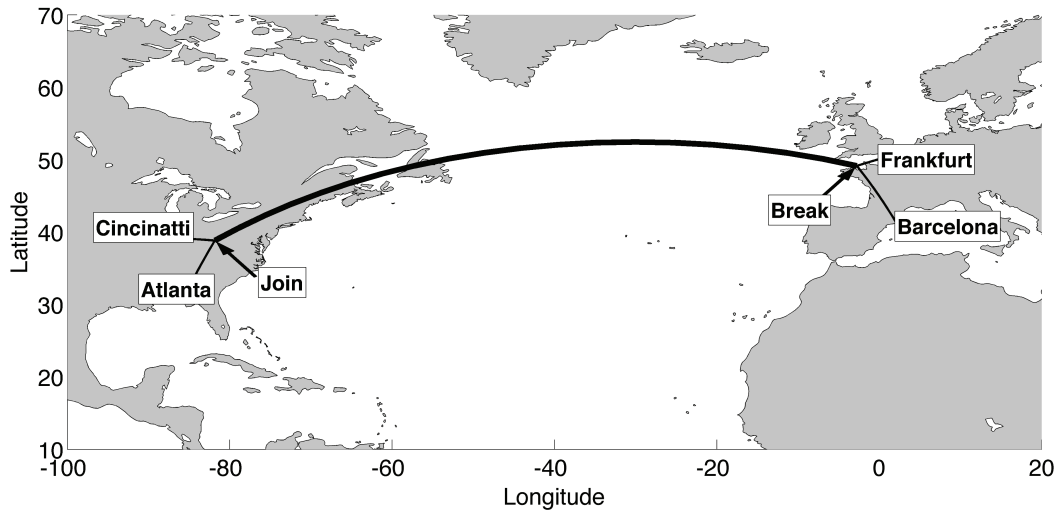
This outlines a simplistic framework for deciding the locations where fleets of size 2 and 3 should join up and break away in order to minimize total fuel-burn. Furthermore the use of virtual flights, as described in Section 2.3.1, in principle, means it is possible to solve for fleets of any size by decomposing into permutations of sub-problems of size 1, 2 and 3.

2.4 Modelling Aircraft Fuel-Burn

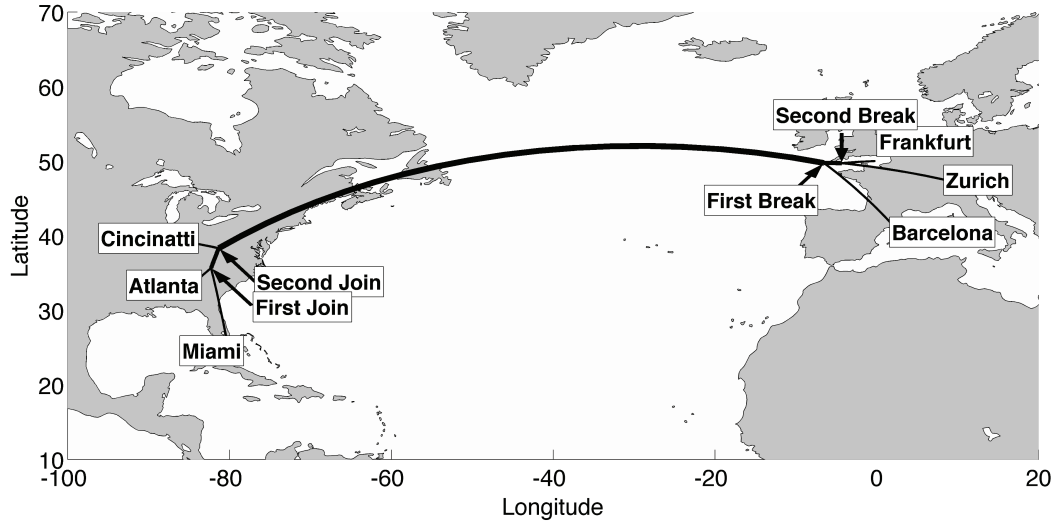
2.4.1 Differential Fuel-Burn Model

A nominal rate of fuel-burn for the aircraft-specific arc-weightings acts only as a reasonable estimate for the final problem. This nominal amount, however, does not incorporate the fact that as an aircraft flies it burns fuel, so decreases in weight, resulting in a lower rate of fuel-burn at later stages of a flight. For example, if one flight travels 1000 km before it meets another, which has flown only 300 km, then a nominal ratio of weights may not accurately reflect this. Therefore the method needs to be able to move from a notion of a constant nominal fuel-burn to one which changes with respect to distance flown. Furthermore, in this model the drag reduction benefits of flying in formation will be applied directly to the coefficient of drag via a discounting factor λ .

A model of an assumed weight change profile for each aircraft can be developed using a rearrangement of the Breguet range equation, outlined by Anderson [58]. To calculate the standard Breguet range equation it is necessary to know the airspeed, drag and thrust specific fuel consumption factor in terms of the aircraft weight. However, assuming a constant thrust specific fuel consumption factor, there are three



(a) Optimal formation route for the two routes Flight A and Flight B



(b) Optimal formation route for the three routes Flight A, Flight B and Flight C

Figure 2.9: Optimal join and break points for fleet size 2 and 3

typical cases for calculation of the range equation. These entail fixing two and allowing one of airspeed, angle of attack, and altitude to vary as the aircraft flies, burns fuel and its weight decreases.

While the methodology of this chapter allows for any of these three cases, the following rearrangement, and the fuel burn equation used in this thesis, assumes that the aircraft's angle of attack and altitude remain fixed while the airspeed is allowed to vary. Thus as fuel is burned the airspeed will slowly decrease. However, for completeness the corresponding equations of the remaining two cases will be briefly outlined in Section 2.4.1.1.

Let dW denote a change in weight W of an aircraft due to fuel consumption over an increment of time, dt . Assuming constant level flight during cruise, thrust available T_A equals thrust required T_R . Thus, given a thrust specific fuel consumption factor, C_t the following relation holds:

$$dW = -C_t T_R dt, \quad (2.5)$$

which rearranged with respect to time, dt , is

$$dt = -\frac{dW}{C_t T_R}. \quad (2.6)$$

For the incremental distance, dr , travelled by the aircraft, over an increment of time, dt , Equation (2.6) is multiplied by a stream-free velocity V_∞ so

$$dr = V_\infty dt = -\frac{V_\infty dW}{C_t T_R}. \quad (2.7)$$

Rearranging Equation (2.7) leads to the rate of fuel burnt per unit of distance

$$\frac{dW}{dr} = -\frac{C_t T_R}{V_\infty}. \quad (2.8)$$

Then for a given coefficient of lift, C_L , and drag, C_D , $T_R = \frac{W}{C_L/C_D}$. Using the definition that for a given density ρ_∞ ,

$$V_\infty = \sqrt{\frac{2W}{\rho_\infty S C_L}}, \quad (2.9)$$

and substituting these into Equation (2.8) leads to

$$\frac{dW}{dr} = -\frac{C_t C_D W}{V_\infty C_L} = -\sqrt{\frac{\rho_\infty S}{2}} \frac{C_t}{C_L^{1/2} / C_D} W^{1/2}. \quad (2.10)$$

To include the formation drag reduction a discounting factor λ is used and the coefficient of drag is therefore replaced by $C_D = \lambda C_{D_{solo}}$. Given a fixed angle of attack and assuming constant C_t , C_L , C_D and density ρ_∞ (i.e. a constant altitude) then $\gamma = \sqrt{\frac{\rho_\infty S}{2}} \frac{C_t}{C_L^{1/2} / C_{D_{solo}}}$ can be used as the contribution of the constant terms. Equation (2.10), for a given weight W and drag-discounting factor λ , then becomes

$$\frac{dW}{dr}(W, \lambda) = -\lambda\gamma W^{1/2}. \quad (2.11)$$

The constant terms of γ can all be calculated directly from BADA while the W required to evaluate this equation is determined after a certain flight distance, by following through with this derivation enables it to be calculated. First integrate dr between the limits $r = 0$ (when $W = W_0$, the initial weight) and $r = R$ (when $W = W_1$, the final weight),

$$R = \int_0^R dr = - \int_{W_0}^{W_1} \frac{dW}{\lambda\gamma W^{1/2}} = \int_{W_1}^{W_0} \frac{dW}{\lambda\gamma W^{1/2}}. \quad (2.12)$$

Given λ and γ are constant then

$$R = \frac{1}{\lambda\gamma} \int_{W_1}^{W_0} \frac{dW}{W^{1/2}} = \frac{2}{\lambda\gamma} \left(W_0^{1/2} - W_1^{1/2} \right), \quad (2.13)$$

completing the derivation of the Breguet range equation [58].

Equation (2.13) can be rearranged to give the final weight W_1 , given an initial weight W_0 , discount factor λ and distance flown R :

$$W_0(W_1, \lambda, R) = \left(\sqrt{W_1} + \frac{\lambda\gamma}{2} R \right)^2. \quad (2.14)$$

This equation starts with a fuelled aircraft (*i.e.* knowledge of W_0) and gives an estimate of the final aircraft weight, W_1 , after flying a given distance, R , with a drag-discounting factor λ and thus W_1 is a function of initial weight W_0 , distance and λ . Similarly, given a final weight, W_1 , (*i.e.* when all normal fuel has been used) one can estimate the ‘fuelled’ initial weight W_0 needed to fly a distance R with discount factor

λ and thus W_0 is now a function of final weight W_1 , distance and λ :

$$W_0(W_1, \lambda, R) = \left(\sqrt{W_1} + \frac{\lambda\gamma}{2}R \right)^2. \quad (2.15)$$

Equations (2.14) and (2.15) are exploited in the following sections to both estimate the fuel burnt and estimate the arc-weightings used in the geometric method.

2.4.1.1 Alternative Forms of Fuel Burn Equation

As previously mentioned, the Breguet range equation and thus the resulting fuel burn equations can be formulated in a number of ways. The two main alternatives to the above outlined form are as follows.

The first is if airspeed and angle of attack are fixed and altitude (i.e. ρ) varies as fuel is burned. This leads to the form:

$$W_0 = \exp\left(R \frac{C_t C_D}{V C_L}\right) + W_1. \quad (2.16)$$

Whereby, looking at Equation (2.9), to match the decrease in the aircraft weight from fuel burn the air density ρ must also decrease and thus the altitude slowly increases.

The second is assuming airspeed and altitude remain fixed and the angle of attack varies. This includes the additional assumption of a parabolic drag polar, that is for the induced drag correction factor K , and zero-lift drag coefficient C_{D_0} , then $C_D = C_{D_0} + KC_L^2$. This leads to the following form:

$$W_0 = \frac{\sqrt{\frac{E}{F}} \tan\left(R \frac{C_t \sqrt{EF}}{V}\right) + W_1}{1 + \tan\left(R \frac{C_t \sqrt{EF}}{V}\right) \sqrt{\frac{E}{F}} W_1} \quad (2.17)$$

where for brevity $E = \frac{1}{2}C_{D_0}\rho_\infty V^2 S$ and $F = \frac{2K}{\rho V^2 S}$. Again, using Equation (2.9), in order to match the decreasing W the coefficient of lift C_L must decrease and consequently the aircraft's angle of attack must be lowered.

2.4.2 Initial Weight Estimation

In order to predict the fuel burn rate at different points along a flight an initial weight value is needed. The total initial fuel is defined to be the fuel required to fly the entire journey plus enough reserve fuel. The initial fuel will be a large factor in the overall take off weight. In general, formations must deviate from their individual

solo routes in order to meet up with other formation members, increasing the total distance travelled (even if they burn less fuel in doing so). Therefore in order for an aircraft to safely fly a formation route it must, as a conservative estimate, carry enough fuel so that it could, if necessary, fly the longer ‘formation’ route entirely solo without any reduction in fuel-burn. In general this means that any aircraft planning to join in formation must carry more fuel relative to the same aircraft flying solo and in turn it will burn fuel at a slightly increased rate. As there are currently no rules in place for commercial formation flight to address this, an assumption is made that for either solo or formation flight each aircraft must carry enough fuel to take off, land and fly 110% of the full cruise distance solo. This additional 10% of the cruise distance will represent the reserve fuel requirement.

Finally in the absence of specific aircraft payloads this work assumes the same nominal payload of 70% that is used in BADA and so the Zero Fuel Take Off Weight (ZFTOW) can be taken directly from BADA. To this ZFTOW, the weight of the fuel required is then added to reach an estimate for the initial weight W_0 . This assumption means that the initial take off weight is a function of cruise distance. This can be incorporated into the weight Equation (2.15) using $\lambda = 1$ and $R = 110\%$ of the formation distance.

2.4.3 Differential Fuel-burn Arc-weightings

At the point of rendezvous (and similarly break-away) each aircraft will have burnt a certain amount of fuel, be a particular mass and therefore burn fuel at a particular rate. The difference in the individual amount of fuel burnt (and the range of fuel-burn rates) to reach the rendezvous (or break-away) point may be vast, but at least at a point-wise level one can consider the fuel-burn rates to be essentially constant. Therefore it is only necessary to calculate the point-wise fuel-burn arc-weightings at the rendezvous and break-away points for use within the geometric model. These can be calculated from Equation (2.11), using the current weight at the join point W_{join} (or similarly break point W_{break}) and the current discounting factor λ .

Optimal formation paths will still be built up of great circles, with the join angles calculated for the rate of fuel being consumed at the point of join. That is, using

Equations (2.8) and (2.14) all the arc-weights at the join:

$$\begin{aligned}
w_{A_{join}} &= \frac{dW_A}{dr}(W_{A_{join}}, \lambda_{A_{solo}}) \\
w_{B_{join}} &= \frac{dW_B}{dr}(W_{B_{join}}, \lambda_{B_{solo}}) \\
w_{C_{join}} &= \frac{dW_A}{dr}(W_{A_{join}}, \lambda_{A_{form}}) + \frac{dW_B}{dr}(W_{B_{join}}, \lambda_{B_{form}})
\end{aligned} \tag{2.18}$$

or similarly the arc-weights at the break point:

$$\begin{aligned}
w_{A_{break}} &= \frac{dW_A}{dr}(W_{A_{break}}, \lambda_{A_{solo}}) \\
w_{B_{break}} &= \frac{dW_B}{dr}(W_{B_{break}}, \lambda_{B_{solo}}) \\
w_{C_{break}} &= \frac{dW_A}{dr}(W_{A_{break}}, \lambda_{A_{break}}) + \frac{dW_B}{dr}(W_{B_{break}}, \lambda_{B_{break}})
\end{aligned} \tag{2.19}$$

can be calculated and the method of Section 2.2 can then be used. For solo flight $\lambda_{A_{solo}} = \lambda_{B_{solo}} = 1$, while during formation this work assumes that the discounting factors during formations are $\lambda_{A_{form}} = \lambda_{B_{form}} = \lambda_{f,2}$, that is, an equal share of the discount for all formation members. The values for $W_{A_{join}}$ (similarly $W_{B_{join}}$) are calculated according to $W_1(W_{A,0}, 1, R_{A_{join}})$ where $R_{A_{join}}$ is the distance from $A_{departure}$ to the join point and $W_{A,0}$ is the initial weight of A. The values for $W_{A_{break}}$ (similarly $W_{B_{break}}$) are calculated according to $W_0(W_{A,1}, 1, R_{A_{break}})$ where here $R_{A_{break}}$ is the distance from $A_{destination}$ to the break point and $W_{A,1}$ is the final weight of A.

The following outlines the method for estimating the differential arc-weightings for a formation route. This is essentially a fixed-point iteration algorithm, starting with nominal entries for initial values and then updating and recalculating through each iteration to improve the solution. The steps are as follows:

1. Take the inputs of two flights: The aircraft types and departure and destination airports.
2. Using BADA, assign nominal initial values for: Aircraft initial masses W_0 and Geometric weights w_A , w_B , w_C for both join and break point.
3. Find the optimal formation route.
4. Calculate the distances flown by each flight for the formation route.

5. Calculate the aircraft fuel burn for the given distances using Equation (2.14) or Equation (2.15).
6. Update estimated initial masses W_0 and final masses W_1 based on fuel required.
7. Update Geometric weights w_A , w_B , w_C for both join and break point using Equations (2.18) and (2.19).
8. Calculate total fuel burnt for each flight: $W_0 - W_1$.
9. If the difference between the new geometric weights and previous ones is significant enough, repeat steps (3)-(9).

In the Transatlantic examples studied in Section 2.6, this algorithm converged in two or three iterations, even with significant variation in weighting factors between rendezvous and break away.

2.5 The Global Fleet Assignment Problem

Given the optimized routes and costs for all possible pairings, it remains to select compatible fleets. That is, by assigning each aircraft to one formation, find the subset of all possible formations which minimises the total cost. This is known as the fleet assignment problem.

2.5.1 Fleet Assignment Using a Mixed Integer Linear Program

Each flight can only belong to one formation (or fly solo), a Mixed Integer Linear Program (MILP) solver is used to generate the optimal subset of formations which minimise the total cost. The optimization problem, based on similar work by Xu et al. [52], is formulated as follows: for N_a aircraft there are N_f possible favourable formations, that is formations which produced a fuel saving, including N_a solo formations (those which do not produce savings are discarded). A pairing, $p_{j,i} = 1$, if and only if aircraft i is included in formation j . Furthermore if formation j is used it will incur a cost of c_j . The binary choice is then, whether formation j is chosen in the solution, (so $x_j = 1$) or not ($x_j = 0$). Therefore the MILP is used to optimally assign each aircraft into a formation by choosing the state of each x_j . That is:

$$\begin{aligned}
& \underset{x}{\text{minimize}} && \sum_{j=1}^{N_f} c_j x_j, \\
& \text{subject to} && \sum_{j=1}^{N_f} p_{j,i} x_j = 1, \quad \forall i \in \{1, \dots, N_a\}, \\
& && x_j \text{ binary}, \quad \forall i \in \{1, \dots, N_a\}.
\end{aligned} \tag{2.20}$$

Therefore there are N_f variables and N_a constraints, so solving in such a way is highly effective for smaller problems. However, a MILP's complexity grows with the number of variables, number of constraints and the non-convexity of the problem [74]. The non-convex nature of this problem, *i.e.* there are many possible local minima, means that finding a global minimum is already a difficult task. Therefore as the size of the problem increases (*i.e.* the number of aircraft or formations) the amount of resources needed to solve the assignment problem will also increase.

2.5.2 The Combinatorial Impact

For a formation size n , from a list of N_a possible aircraft, the number of possible formations that can be made is calculated by the binomial coefficient:

$$N_a \text{ choose } n \equiv \binom{N_a}{n} = \frac{N_a!}{n!(N_a - n)!} \text{ for } 0 \leq n \leq N_a. \tag{2.21}$$

This number grows dramatically with an increase to either N_a or n and is the main reason for developing the quick geometric approach of Section 2.2 for calculating the formation routes. When considering the global problem, although the number of routes N_a will vary the most, it is combinatorially more important to keep n low. Very roughly speaking, an increase to N_a by an order of magnitude, will increase the number of combinations by n orders of magnitude. Similarly, with N_a fixed, increasing n by 1 results in an increase in combinations by a factor of $N_a/(n + 1)$. Therefore an increase in n by 2 would then increase combinations by $N_a^2/(n + 2)(n + 1)$. This increase in combinations impacts both aspects of the problem, firstly the routing problem, as more formation routes need to be calculated, and secondly the assignment problem, with each additional flight adding a constraint and each additional combination adding a variable.

The geometric method has been developed to be a very fast way of calculating individual formation routes, making it possible to evaluate lots of combinations very quickly. However there comes a point when the combinatorial effect overcomes this

Table 2.2: Binomial coefficients for varying number of aircraft N_a or formation size n

	100 Aircraft	500 Aircraft	1000 Aircraft
Formation size 2	4,950	124,750	499,500
Formation size 3	161,700	20,708,500	166,167,000
Formation size 4	3,921,225	2,573,031,125	41,417,124,750

computational advantage. One can see from Table 2.2 how changes in N_a or n can quickly effect the number of combinations that need to be evaluated. Therefore for much larger problems it is a balancing act, between the number of flights N_a considered and the size of the formation n . Despite this, it should not provide too much of an issue, as global lists of flights can be partitioned in a number of ways to keep n relatively low. Examples of partitioning are direction and location: such as eastbound Transatlantic; time of day, morning or evening; and individual airline companies. These choices are, on the most part, not explored in this thesis, however the following case study demonstrates an example of 210 eastbound Transatlantic flights.

2.6 Case Study: Transatlantic Formation Flights

Using the methodology of the previous sections, an Official Airline Guide (OAG) dataset for the month of September 2010 of 210 common Transatlantic flights, between 26 US and 42 European airports is now examined. Each flight has a particular aircraft to fly it and so individual performance factors can be taken from BADA.

The aim is to create formations in order to minimize the total cost (kg of fuel burnt) of the entire fleet. Each flight is treated as non-greedy, doing what is best for the fleet as a whole rather than individual gain. In this sense the fleet could be thought to represent a single airline company. Furthermore, remaining with our initial assumptions, the results are also time-free, based on the optimal location for joining a fleet and breaking away and are therefore not constrained to a specific schedule. In order to directly compare time-free solo routes with time-free formation routes, aircraft are instructed to fly at speeds which minimise their fuel-burn (or while in formation, the total fuel-burn of all members). An analysis of the possible impact of scheduling on formation flight is explored in Section 2.6.4.

As discussed in Section 2.2.6, minimum horizontal distances to climb and descend are used and are calculated based on rates of climb and descent in BADA. The inclusion of this ensures the join and break points lie suitably far away from each

airport, in order to allow each aircraft to reach a fixed cruising altitude of 37,000 feet. The case study is evaluated for two maximum formation sizes, which are: the $n = 2$ problem and the $n = 3$ problem. Finally the fixed proportional discounting rates of $\lambda_{f,2} = 0.9$ and $\lambda_{f,3} = 0.85$ (from Table 2.1) are used to represent the fuel saving benefits of formation flight.

All the methods to find the optimal formation route outlined within this thesis (unless otherwise stated) have been implemented in Matlab. The MILP used in the assignment problem is first formulated in Matlab and then run through Gurobi's [75] MILP solver. Both stages of the problem were implemented on the same machine: *Macbook Pro - 2.4GHz i5 with 16GB of RAM.*

2.6.1 For Formations of up to Two Aircraft

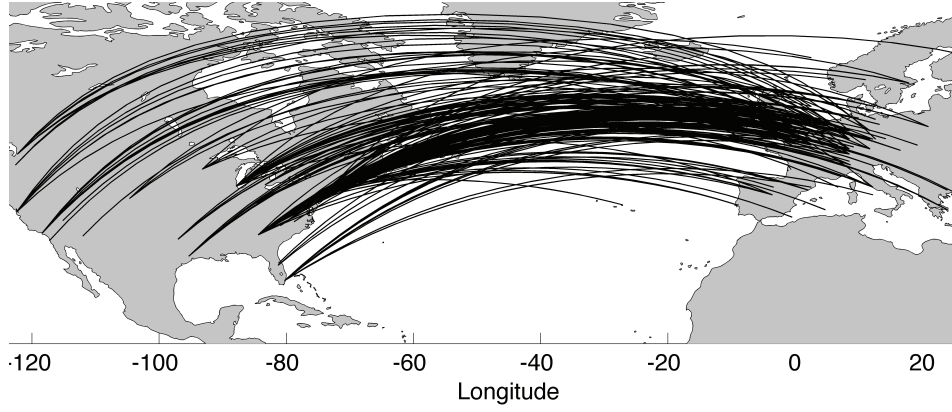
Using all the methods outlined in the previous sections, all 21,945 formations of size 2 were calculated along with each corresponding fuel-burn cost. This took roughly 0.0005 seconds per combination resulting in a total time of around 11 seconds. Then given the cost for each possible formation (including solo flights), the MILP was run, taking a further 3 seconds, and resulted in the 210 flights being assigned to 105 formations of size 2, (Figure 2.10(b)) with the entire process taking 14 seconds. Compared with solo flight, the total average saving was a very reasonable 8.7%.

2.6.2 For Formations of up to Three Aircraft

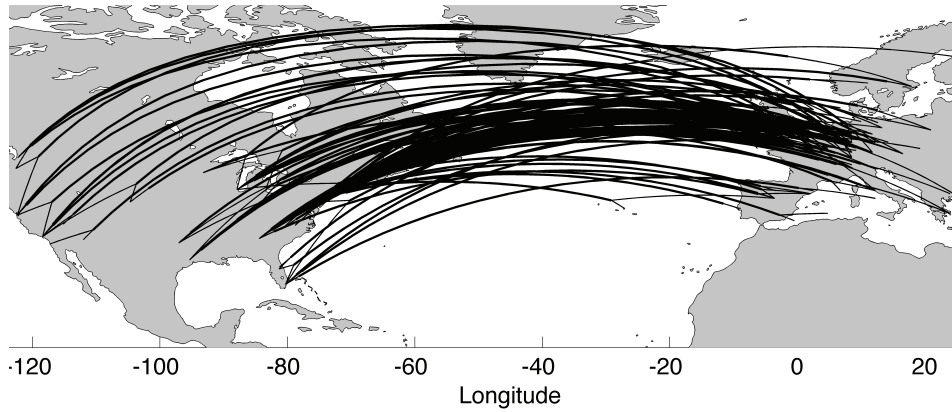
For formations of size 3 there are 1,521,520 possibilities, followed by the 21,945 formations of size 2, bringing the total number of combinations to evaluate to 1,543,465. The $70\times$ increase in combinations has lead to a $245\times$ increase to enumeration time. This is due to the increased complexity of calculating formations of size 3 and the need to evaluate all different orders of joining and breaking away. The mean individual formation computation time has increased to roughly 0.0018 seconds (about $3\times$ that of a formation of size 2) and amounts to 45 minutes for all combinations.

The MILP also suffers from the increase in combinations and takes just over 10 minutes to optimally assign the formations. Therefore an optimal solution can be calculated for formations of size three in under an hour. This runtime could be substantially reduced, if required, by parallel evaluation of the formation costs.

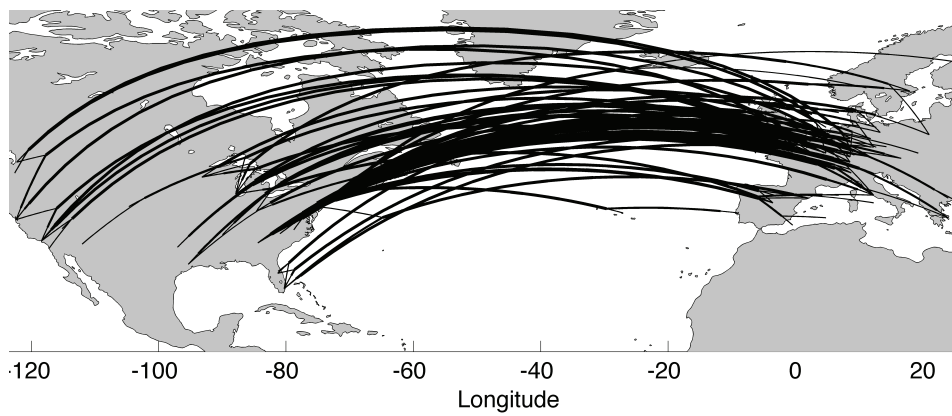
The 210 flights were assigned into 70 formations of size 3 (Figure 2.10(c)). The average fuel-burn saving of the formations of size 3, compared to solo flight, was 13.1%.



(a) Original solo flights



(b) Formations of size two



(c) Formations of size three

Figure 2.10: Transatlantic formation routes

Table 2.3: Fuel saving against corresponding solo routes for formations of size 2 and formations of size 3

Formation Size	Combinations	Computation Time (mm:ss)			Fuel Saving
		Route Enumeration	MILP		
2	21,945	00:11	00:03	8.7%	
3	1,521,520	45:00	10:12	13.1%	

Table 2.4: Deviation (km) in route distance between formations and their solo routes.

Formation size	With common airport	Formation total			Per aircraft		
		Min	Avg	Max	Min	Avg	Max
2	72 (69%)	0	26	240	0	13	190
3	69 (99%)	0	70	530	0	23	308

2.6.3 Comparison of Results

The first observation is that many formations were made between routes which required little deviation from their original solo path (the levels of deviation between formation and solo routes can be seen in Table 2.4). Even though distance to climb and descend restrictions were implemented, many pairings found the best gain to be between other flights that shared either their departure or destination airport (all flights are distinct so could not share both). These low deviations also mean a minimal increase to both the amount of reserve and main fuel required for each flight.

For $n = 2$, all aircraft were assigned into pairs and so there were 105 final pairings. Of these pairings, 72 (69%) shared either a departure or destination airport. The total formation deviation (*i.e.* the total difference in distance between the formation and solo route for all formation members) ranges from 0 km to 240 km, while the average over the whole fleet remains low at around 26 km for the entire formation, or 13 km

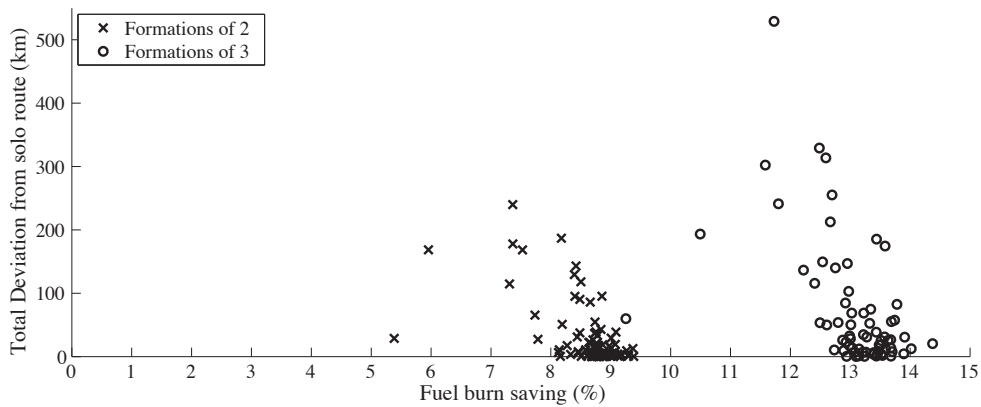


Figure 2.11: Deviations in distance between solo and formation flight

per formation member.

For $n = 3$, the 210 flights were all assigned into 70 formations of 3. For the 70 formations, 69 (99%) shared at least one common airport with other fleet members. Furthermore of these, 29 (41%) shared exactly 1 airport, 26 (37%) shared exactly 2 and 14 (20%) sharing exactly 3 airports (as all flights were unique, the maximum number of common airports possible was 3). The total deviation ranged between 0 and 530 km with an average fleet total of 70 km resulting in a per aircraft average deviation of just 23 km.

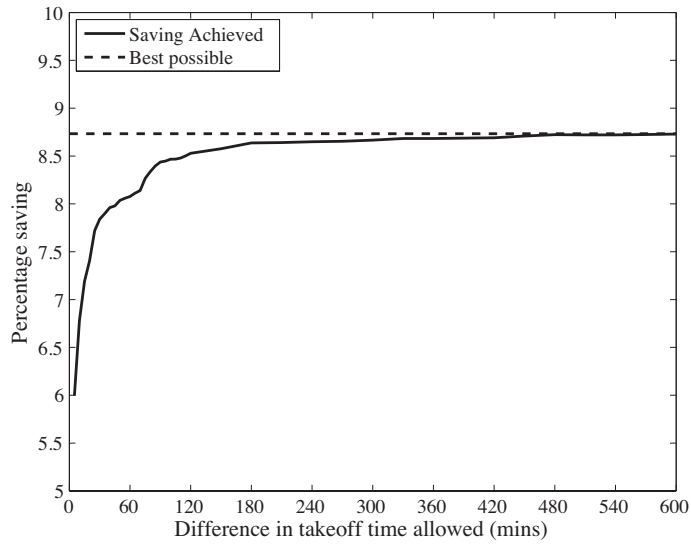
The results in Table 2.4 along with Figure 2.11 show that the formations who achieve greater savings are, as expected, those who need to deviate from their solo routes least. There is clear trend in Figure 2.11 towards lower deviation and higher fuel-burn percentage savings (some of the outlying results are likely due to the geographical sampling of the airports, for example the west and east coast of the US).

2.6.4 Analysis of Aircraft Scheduling on Formation Flight

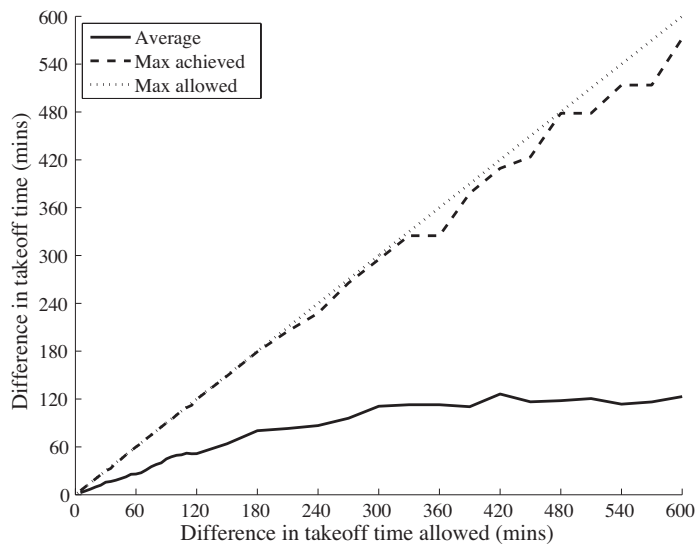
The development in this chapter has optimized purely for fuel use and ignored the impact of scheduling. However, scheduling factors such as crew rosters, passenger demand and airport capacity all influence flight timing and are included in multi-objective schedule optimisation [52,76–78]. The incorporation of scheduling objectives in formation flight is beyond the scope of this thesis. However, this section shows how the effect of formation flight on scheduling can be analyzed and constrained.

One indicator of likely schedule impact is the deviation. The per-aircraft average deviations in route distance between formations and their solo routes as outlined in Table 2.4 are relatively low, at around 20 km (or roughly 1 minute at mach 0.8). This means that per-aircraft flight durations for formations can remain close to their solo counterparts.

Given a formation and its optimized route, determined using the methods presented, the formation’s schedule impact can be evaluated. This is measured in terms of the total change in takeoff times in minutes, assuming that all flights in the formation land no later than their original scheduled landing time. It is equivalent to the total delay if no flight takes off earlier, or any other sharing of the schedule ‘shift’ between the flights in order to accommodate the formation. Once all the formations have been evaluated, it is then possible to discard all those whose takeoffs are shifted by more than a certain schedule threshold prior to solving the assignment problem as in Section 2.5.



(a) Percentage saving achievable for limits on allowable total change to takeoff



(b) Take-off time changes allowed against takeoff time changes achieved

Figure 2.12: The effect of maximum permitted takeoff changes

Figure 2.12(a) shows the variation in total fuel saving with the maximum permitted takeoff change, denoted Δt , for the Transatlantic case study in formations of two aircraft. As expected, the fuel saving decreases as the takeoff change limit is tightened. However, roughly 8% saving is still available without changing any takeoffs by more than an hour, and 6% is available with no changes of more than five minutes.

The background of this result is explored in Figure 2.12(b), which shows that even with very long schedule shifts available, the average takeoff time change is about two hours. The implication is that the Transatlantic flight set in the case study includes a large selection of flight pairs that can benefit from formation, including many with compatible timing.

2.6.5 Utilisation of Potential Saving

Finally it is interesting to analyse the fuel-burn savings compared to the maximum-achievable potential of the routes. For $n = 2$, $\lambda_{f,2} = 0.9$, meaning if both aircraft started and finished in formation, observing the fuel-burn saving over the entire flight, the maximum achievable saving would be 10%. As outlined in section Section 2.2.6, aircraft need time to climb and descend and so cannot save fuel over the entire flight. Allowing for this means that in the case of the 210 Transatlantic flights, on average, for 7.2% of the flight the aircraft is unable to achieve any fuel reduction benefits. This leaves 92.8% of the flight available for fuel-burn saving by flying in formation.

With all this taken into account the average maximum-achievable-formation-saving is actually 9.3%. Therefore for $n = 2$ the case study results of 8.7% represent a 94% utilisation of the possible savings. Similarly for $n = 3$, $\lambda_{f,3} = 0.85$ which leads to an average maximum-achievable-saving of 13.9%. In this case the 70 formations of 3 achieved 13.1%, resulting in a utilisation of 94%.

Therefore what this utilisation metric can quickly express, is how well suited a group of flights is to flying in formation, with the eastbound Transatlantic case study being a good example. The routes are all in a similar geographical location, heading in a similar direction and so achieve a high percentage of utilisation.

2.7 Summary

This chapter has explored a method for finding optimal routes for formation flight. Firstly an extension to the Fermat-Torricelli problem allowed the decoupling of a complex problem, providing a fast and effective framework to find optimal formations for a given list of routes. Using a set of general aircraft performance coefficients from Eurocontrol's Base of Aircraft Database allows a more accurate representation of routes containing distinct aircraft to be incorporated into the solution. The introduction of a differential aircraft weighting scheme allows formation fleets to be more accurately assigned and routed to account for differing aircraft efficiencies. The sim-

ple iterative-updating scheme also allows room for possible expansion in future, such as a more accurate calculation of the specific proportionality discount factor between particular aircraft pairings.

The methods and fundamentals of this chapter have been designed to be both extensible and scalable. Allowing assessment of the potential of formation flight, on large sets of routes and varying sizes of formation fleets, while remaining computationally tractable. The analytic nature of the proposed method means millions of possible combinations of formations can be quickly calculated, allowing a Mixed Integer Linear Program to tackle the global fleet assignment problem. The outlined methods have then been tested against a case study, of a representative-region of possible formation flight, for 210 Transatlantic flight routes. Despite some of the discussed combinatorial impacts, the globally optimal formation fleets for the case study were found in under an hour. Results show possible average fuel-burn savings against solo flight of around 8.7% for formations of 2 and 13.1% for formations of up to 3. While even for a relatively small problem, the optimal results had a high degree of utilisation against maximum achievable saving.

This details some of the core methodology used throughout this thesis to find solutions to the formation flight routing and assignment problem. While the use of an analytic approach outlined is somewhat simplistic and requires a number of assumptions to be made, it allows for tractability within the global problem. It will be shown in Chapter 3 that this geometric methodology of calculation of all formation combinations followed by a MILP assignment can still act as a reasonable estimate to the assignment problem when more complicated routing is required. Whereby more complex problems such as routing through wind can feasibly be left for a post-process after the assignment. Finally, initial results of this chapter, specifically Section 2.6, will be explored in greater depth, alongside additional case studies, in Chapter 5.

Chapter 3

Wind-optimal Routing for Formation Flight

3.1 Introduction

The work of Chapter 2 has laid out the main foundations and methodology for this thesis. For simplicity we assumed that flights would fly great circle paths, however in reality flights alter their routes for a number of reasons, but notably to avoid wind. It is therefore necessary to observe the effect the assumption of using great circle paths has on the routing and assignment stages of the formation flight problem.

In the presence of wind, aircraft will commonly need to deviate from a great circle path [79]. Aircraft usually attempt to fly the path of least-time, avoiding areas of large headwinds, opting for those with less resistance or even tailwinds. Flying a large proportion of a flight with a reasonable tailwind, such as an east bound Transatlantic flight, can reduce flight times by as much as a few hours compared to their west bound counterparts [80]. Due to this, North Atlantic Tracks (NATs) are published daily, both to alleviate traffic and to account for areas of greater predictability in certain weather patterns.

Research into optimal path planning and trajectory optimisation in the presence of wind is broad, encompassing a number of techniques, including optimal control [81–85], Markov decision processes [86] and genetic algorithms [87, 88]. It is clear that a full trajectory optimisation (or a least a higher-fidelity solution) for aircraft looking to fly in formation would likely be needed. However, the added computational time required to compute trajectories would be significant.

The previously discussed geometric method of Chapter 2 does not take into ac-

count the almost certain likelihood of encountering wind. It is therefore beneficial to see how well assignment based on geometric formations performs in the presence of predictable wind. The method outlined is not intended to be a likely candidate for routing on a global scale, rather it is designed to simply gauge the use of the faster geometric approach.

The same methods as Chapter 2 will be used, to first route for all possible combinations of formations and then use the associated costs to optimally assign flights to formation pairings. The aim of this chapter will then be to assess, and ultimately benchmark, the use of the geometric approach to calculate an ‘approximate cost’ for use within the assignment stage. It will be shown that by using the geometric approach to obtain an assignment based on the approximate geometric-cost of formations, will allow the more complex problem of routing through wind to be left as a post-process. In turn significantly reducing computational time with relatively small reductions to the overall savings.

3.2 Wind Routing Method

While weather is chaotic and difficult to accurately predict over longer periods of time, short term ‘day-to-day’ weather forecasts are generally easier to predict. Aircraft flight plans, chosen prior to takeoff account for a wide variety of both predictable and unpredictable factors. This typically includes the use of weather forecasts to avoid potentially adverse weather [89]. Changes to the weather during a flight can often then be accounted for with en-route flight path adjustments and altitude changes [90]. For the work of this chapter a number of assumptions to the wind routing problem are made. Flights are fixed to a single flight level (altitude) where they will encounter a single ‘layer’ of predicted wind which remains unchanged for all the flights at all times.

3.2.1 Generating Static Wind Fields

To generate wind fields, a set of evenly spaced geographical points is first generated across the entire map. Each of these points are then assigned vectors with a random direction and a random (but bounded) magnitude. A 1-dimensional interpolation, using the fast fourier transform [91], is then applied between these points, both horizontally and vertically, to remove noise and create a smooth vector field. The volatility of the wind field can also be prescribed in order to produce ‘bumpier’

surfaces. The final wind field is essentially a matrix of discrete sample points, each with a corresponding vector of a given magnitude and direction, with the assumption that the wind at any point in the space can be estimated via interpolation. Some example wind plots for varying levels of volatility are shown in Figure 3.1; the colour corresponds to a lateral movement from left to right, with green indicating tailwind and red indicating headwind.

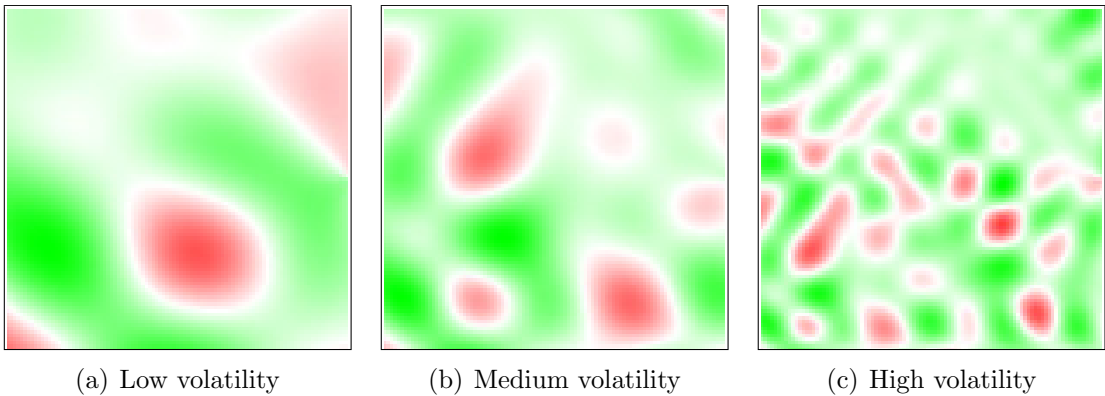


Figure 3.1: Random wind fields for different levels of volatility

3.2.2 Routing Through Wind Fields

To route optimally for formations in the presence of wind, a numerical approach must be used. In line with the approach of Chapter 2 a fixed proportional discount factor is used to represent the fuel burn reduction when aircraft fly in formation. For formation routing the formation path is broken down into five connected sections: two solo legs from departure to the rendezvous; the formation leg between rendezvous and break point; and two solo legs from the break point to the destinations. Each of these five sections is determined by a finite number of variable way-points, each with a longitude and latitude location, interconnected by great circle paths, as depicted in Figure 3.2. The five sections are then connected via the corresponding rendezvous and break-away points (the grey points of Figure 3.2 which are also variable), the combination of all these way-points simulates a formation route.

In order to cost the relative benefits of formation flight over flying solo, the solo flights must also be routed through the wind. This is a similar process and is essentially the same implementation but with only one path, the solo leg, and its one set of variable way-points to optimise for.

In order to account for the wind let us first define a simple relationship between

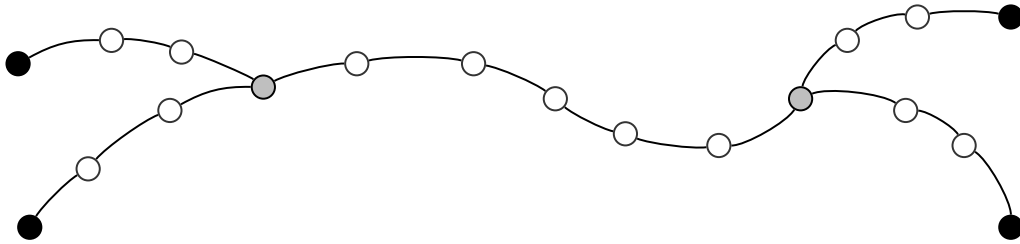


Figure 3.2: Variable way points making up formation route

the aircraft motion through the air and the wind, as represented in Figure 3.3, often referred to as the ‘wind-triangle’. The ground vector, \mathbf{V}_G , is the vector corresponding to the aircraft movement over the ground; *i.e.* the ground track and ground speed. The wind vector, \mathbf{V}_W , is then the vector representing the motion of the air; *i.e.* the direction the wind is blowing to with a given wind speed. Finally the heading vector, \mathbf{V}_H , represents the aircraft motion through the air; *i.e.* the heading and airspeed. Finally, the angle, α , between the ground vector and the wind vector is called the wind angle. For this problem \mathbf{V}_G will be predefined by the path and speed we wish to fly, therefore given the wind vector \mathbf{V}_W the problem is to find the necessary heading vector \mathbf{V}_H required to match our desired ground vector \mathbf{V}_G .

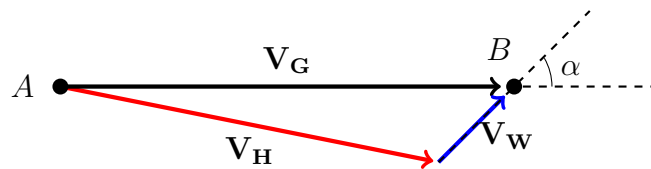


Figure 3.3: Wind-triangle and contributing vectors

For a given placement of the variable way-points the cost of the formation-path connecting them is estimated numerically. Each way-point is fixed and further sample points are generated via interpolation, the entire formation path is then just a list of sample points. At each of these sample points the wind vector, \mathbf{V}_W , is calculated, *i.e.* the direction and magnitude of the wind field at that point, while the ground vector, \mathbf{V}_G , which corresponds to the desired speed and direction the aircraft is already known. Therefore we have knowledge of two of the three components of the wind-triangle of Figure 3.3 and can then calculate the third vector, the heading vector, \mathbf{V}_H .

The contribution of the wind, cont_w , to the ground vector, \mathbf{V}_G , can be calculated using the dot product and the corresponding property of scalar projection. The amount the vector of wind, \mathbf{V}_W , adds to the ground vector, \mathbf{V}_G , is proportional to

the wind angle, α , as follows

$$\text{cont}_w = \|\mathbf{V}_w\| \cos(\alpha). \quad (3.1)$$

Thus angles of α (measured clockwise) between $(-\pi/2, \pi/2)$ will increase the magnitude (the speed) of \mathbf{V}_G (*i.e.* tailwind) and angles of α between $(\pi/2, 3\pi/2)$ will decrease the magnitude of \mathbf{V}_G (*i.e.* headwind). Therefore, to maintain the same ground speed our heading vector, \mathbf{V}_H , must be adjusted.

The wind-triangle is essentially a vector equilibrium stating that

$$\mathbf{V}_G = \mathbf{V}_H + \mathbf{V}_w, \quad (3.2)$$

or equivalently as we require the vector of heading

$$\mathbf{V}_H = \mathbf{V}_G - \mathbf{V}_w. \quad (3.3)$$

Therefore as the ground vector, \mathbf{V}_G , and wind vector, \mathbf{V}_w , are known, the heading vector is straightforward to calculate. The airspeed the aircraft adjusts its throttle setting for, is then the magnitude of \mathbf{V}_H which, given Equation (3.3) and vector law of cosines can be calculated as follows

$$\begin{aligned} \mathbf{V}_H &= \mathbf{V}_G - \mathbf{V}_w \\ \|\mathbf{V}_H\|^2 &= \|\mathbf{V}_G - \mathbf{V}_w\|^2 \\ &= (\mathbf{V}_G - \mathbf{V}_w) \cdot (\mathbf{V}_G - \mathbf{V}_w) \\ &= \|\mathbf{V}_G\|^2 - \|\mathbf{V}_w\|^2 - 2\|\mathbf{V}_G\|\|\mathbf{V}_w\|\cos(\alpha) \\ \mathbf{V}_H &= \sqrt{(\|\mathbf{V}_G\|^2 - \|\mathbf{V}_w\|^2 - 2\|\mathbf{V}_G\|\|\mathbf{V}_w\|\cos(\alpha))} \quad (3.4) \\ &= \sqrt{(\|\mathbf{V}_G\|^2 - \|\mathbf{V}_w\|^2 - 2\|\mathbf{V}_G\|\text{cont}_w)} \quad (3.5) \end{aligned}$$

This is then incorporated into the cost function along with the proportional discounting factor, as in Chapter 2, for the section of the flight flown in formation. The contribution of the wind is therefore, for a given aircraft's required ground velocity V_G , the cumulation of all the incremental wind contributions of all the sampled points. It is important to note here that with tailwind aircraft can fly at lower speeds and achieve additional fuel saving directly from this.

Therefore, given a formation pair and a predefined wind field, a numerical optimisation is used to alter the location of the variable way-points with the objective of

minimising the cost of the path. This optimisation is done using an ‘active-set’ method within Matlab’s ‘fmincon’ function. For each formation pair this takes around 30 seconds to find a suitable solution within a predefined threshold, which is significantly slower than the geometric approach which takes roughly only 0.0005 seconds per route. Due to the problem’s highly combinatorial nature, explored in Section 2.5.2, of calculating routes for all formations means solving for every possible pairings is time consuming. However, the enumeration process is ‘embarrassingly’ parallelisable, and can be split into a number of subproblems and run on a cluster of computers.

Routing for all possible formation combinations for an unrefined and relatively slow method of optimising routes in the presence of wind is not ideal. It may, however be better suited to a ‘post-process’ optimisation applied after an initial assignment is made and thus for a significantly smaller number of formations. The focus of this chapter is to therefore assess if the assignment made, using the geometrically optimal routes, would still act as a reasonable assignment in the presence of more complicated routing.

This completes the outline of the required methods for the wind routing problem. In Section 3.3 an example formation route will be explored, showing a sample solution path. A case study will then be presented in Section 3.4, looking at the same set of Transatlantic routes first outlined in Section 2.6. This will show the potential of using the geometric method for the enumeration and assignment stages while leaving the computationally expensive wind routing method to a post-process.

3.3 Example Formation

The North Atlantic Jet Stream (JS) has a major impact on aviation routing and being a key area of potential for formation flight it is necessary to use the jet stream as an representational wind field. Although the method of Section 3.2.1 is essentially random, the ‘randomness’ can be seeded in order to create predefined wind fields. Thus this wind field is just a randomised wind field which has similar traits to the north Atlantic jet stream.

For this example the two flights used are:

Flight 1: Denver International Airport (DEN) to Frankfurt Airport (FRA)

Flight 2: George Bush Intercontinental Airport (IAH) to Heathrow Airport (LHR)

Routing numerically through the wind, using the methods of Section 3.2.2, results in the routes depicted in Sections 3.3 and 3.3. The solid black lines are the wind optimal

paths while the geometric route, using the methods of Chapter 2, are the dashed blue lines. The solo paths of Section 3.3 are clearly different with the wind optimal route going considerably out of its way to avoid adverse wind. For the formation paths, the two methods produce similar paths to the solo ones, with the two flights rendezvousing at similar locations, but the wind optimal route clearly moves toward the areas of tailwind. The wind optimal route for the formation achieves 5.4% saving, compared to equivalent solo flights routed through wind. If there was no wind and we simply flew the geometrically optimal route in ‘still-air’ then the formation would achieve around 5.3%, compared to the equivalent solo flight. However, if we flew the same geometrically optimal route through the static wind field then it would actually produce a negative saving of -1.4% against the solo wind-optimal routes.

Therefore it is clear, that in the presence of wind, it is necessary to optimise the route to take advantage of tailwind and avoid headwind. However, this has always been expected, the aim of this chapter is to see how well the global assignment performs in the presence of wind. With this in mind, the case study of Section 3.4 will be used to compare the two available routing and assignment methods.

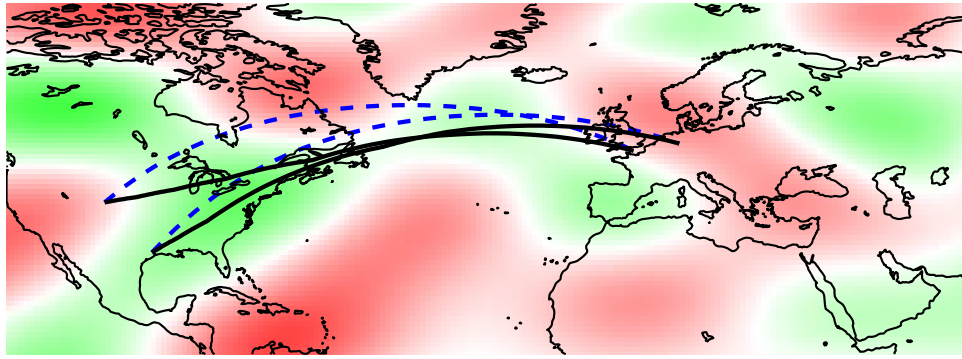
3.4 Comparison of Methods: Transatlantic Case Study

The same list of flights for the case study of Chapter 2 is now reintroduced. The data consists of 210 Transatlantic flights between 26 US and 42 European airports. Again the objective is to create formations of size two in order to minimize the total cost (kg of fuel burnt) of the entire fleet. A fixed proportional discount factor, outlined in Chapter 2, of 0.9 is used to account for the drag saving during formation.

3.4.1 Method

There will be two different routing methods used: Geometric Routing (GR), where the route is calculated via the analytic methods of Chapter 2 and Wind Routing (WR) where the route is calculated using the methods outlined in Section 3.2. Therefore given that wind now has an impact on the formation and solo cost, the GR method will only produce an approximate route and approximate cost for the wind-routing problem.

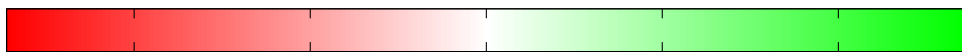
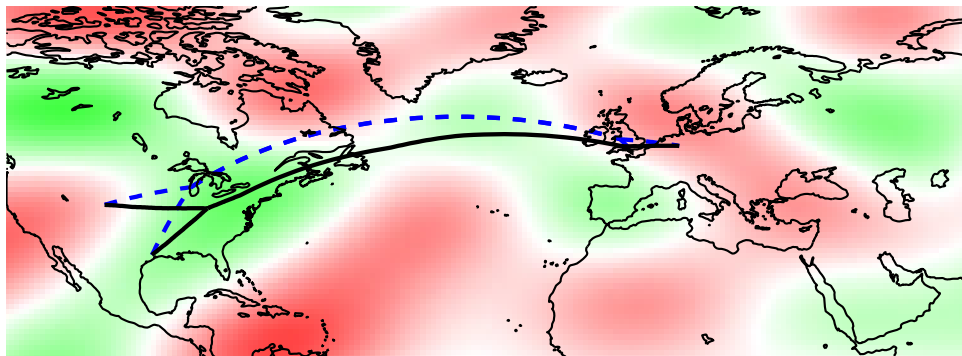
— Wind Optimal Route - - - Geometric Optimal Route



-200 -100 0 100 200
Wind Speed (km/h) in Direction →

(a) Solo Flights

— Wind Optimal Route - - - Geometric Optimal Route



-200 -100 0 100 200
Wind Speed (km/h) in Direction →

(b) Formation Flights

Figure 3.4: Wind Route vs Geometric Route

The approach of this chapter follows the same methods of Chapter 2 but instead uses a three stage solution process:

Enumeration For all possible combinations calculate the formation routes and corresponding costs of flying them.

Assignment Given the costs of all combinations, assign a final fleet of formations to fly in order to minimise total cost.

Post-process Calculate the optimal route for the formations included in the assignment.

The post-processing stage acts much like the enumeration stage, calculating the routes and corresponding costs, however as it comes after the assignment stage it is for significantly fewer flights. The idea being that if we can use an approximate method for the enumeration stage, such as the geometric approach, the more complex routing, such as wind-routing, can instead be added after the assignment stage, as a post-process.

The use of the Geometric method has been motivated by the combinatorial nature of the enumeration stage, calculating all combinations of formation flights. The Transatlantic case study contains 210 flights and therefore has 21,945 different formation combinations needing evaluation. The wind-routing method is essentially a numerical optimisation of the placement of a finite set of variable way points and as a result is not particularly fast. With a runtime of about 30 seconds per formation, enumerating all 21,945 combinations requires about 180 hours of computational time. Compared to taking a total roughly 4 seconds for all combinations for the GR there is clearly a substantial difference. Therefore we aim to see if the enumeration and assignment stage can be tackled by using only the GR, while the WR is left to the post-processing stage.

3.4.2 Enumeration and Assignment Workflow

The two workflows of Sections 3.4.2 and 3.4.2 outlines two different methods that can be used to reach a solution for the wind routing and assignment problem. The input consists of the list of all possible formation flights and we require the output to be these flights allocated into formations and the associated wind costs. Given a list of flights, all the combinations of making formations of size two are created. This list is likely very large, compared to the flight list, as first outlined in Table 2.3.

For Method 1, shown in Section 3.4.2, all formation combinations are routed for wind then the costs of those routes are then assigned using the MILP, creating our final solution. This Method will produce the optimal allocation of flights into formations but as a result will take a substantial amount of time.

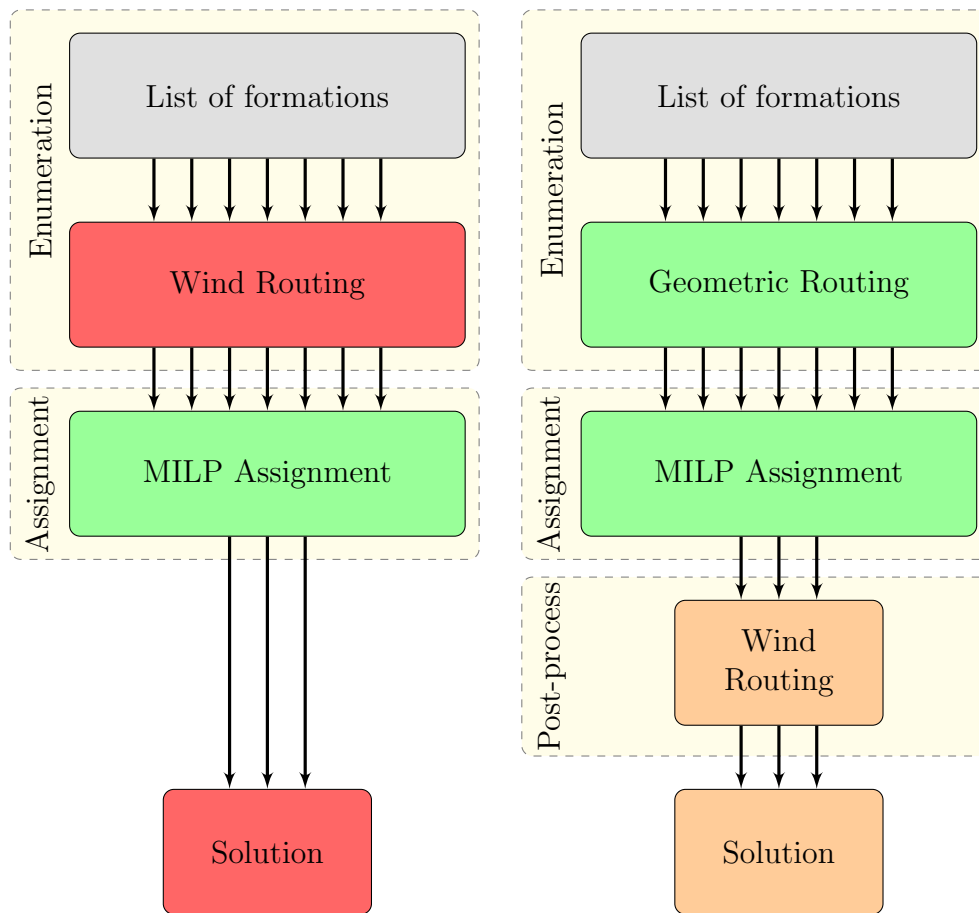
Method 2, shown in Section 3.4.2, uses the geometric routing approach to calculate routes for every combination of formation. As these do not take into account wind, the costs can only act as estimates to the wind routing problem. The costs of the geometric routes are then used within the MILP of the assignment stage to calculate an assignment of flights into formations. The final stage, as the geometric routes and the corresponding assignment are only estimates, requires the final allocation to then be routed for wind, this is called the post-processing stage. Method 2 is a much faster process than Method 1, but can only produce suboptimal allocations.

While Method 1 requires the wind-routing, which is relatively slow, to be used for all possible pairings, due to the embarrassingly parallelisable nature of running every combination, to get results in a reasonable time the enumeration stage is split into a number of smaller subproblems run on a computer cluster. The computationally intensive work of this chapter has been carried out on the University of Bristol's High Performance Computer (HPC) BlueCrystal Phase 2 [92]. Using the HPC does not improve efficiency as the actual total runtime (*i.e.* cpu time) remains the same, instead the time realised by the user (*i.e.* wall-clock time) required to produce the results is more or less the total runtime divided by the number of subproblems it is divided into. With that in mind the total cpu time for this is around 180 hours, drastically more than a geometric wind-free solution. However this method is more a proof of concept to benchmark the geometric results. Note that when referring to time/runtime throughout this section we will mean cpu time.

Finally, one motivating factor of looking at wind-routing is due to the weather phenomenon of the North Atlantic Jet Stream (JS) present when flying Transatlantic. For this case study, two wind fields are presented. The first, used in Sections 3.4.3 and 3.4.4, has been created to be representational of the jet stream over the Atlantic. The second wind field in Section 3.4.5 has been made to represent 'more-volatile' wind, to see how the routing and assignment is affected.

3.4.3 Jet-Stream West-to-East

The formation results travelling from West-to-East over the Atlantic, with a wind field representational of the Jet-Stream, are now shown. Using the two methods



(a) Method 1: Full Wind Enumeration

(b) Method 2: Geometric Estimate

Figure 3.5: Workflow diagrams of two methods for route calculation and assignment for wind routes. Color indicating cpu time required: green - seconds; orange - minutes; red - hours

outlined in Section 3.2, to either apply the wind routing at the enumeration stage or the post-process stage.

By first routing all possible formations combinations for wind and then using those cost to assign flights into formations pairs, as in Section 3.4.2, results in a globally optimal solution. The resulting assignment equates to a saving of roughly 9.48%, against the corresponding solo flights routed through wind, and takes around 180 hours. This runtime break down as follows:

Enumeration: Solo 210 solo flights routed for wind: 10 minutes;

Enumeration: Formations 21,945 formations routed for wind: 180 hours;

Assignment MILP assignment: 4 seconds;

Post-Process None;

Total 180 hours.

A proportion of the saving achieved comes from cruising with a tailwind, so makes any possible saving clearly dependent on the particular wind field used. The routes of this solution are shown in Section 3.4.3. Travelling West-to-East the green area represents a tailwind, while the red area represents a headwind, the intensity of these colours corresponds to the speed of the wind. What is clear from Section 3.4.3 is that the wind routes are not great-circle, rather they are funnelled through the strong-green area over the Atlantic.

If instead Method 2 is used, where the enumeration stage is done using geometric routing and then using these costs an assignment is made, as in Section 3.4.2, the wind routing can be added as a post-process on the substantially smaller list of routes (105 in this case). This assignment is optimal when the metric is GC but under the metric of WC this assignment is suboptimal. The resulting assignment, when routed through wind, achieves 8.31% against solo flight, but takes significantly less time at around 60 minutes. This runtime breaks down as follows:

Enumeration: Formations 21,945 formations routed geometrically: 10 seconds;

Assignment MILP assignment: 4 seconds;

Post-Process: Solo 210 solo flights routed for wind: 10 minutes.

Post-Process: Formations 105 assigned formations routed for wind: 50 minutes.

Total 60 minutes

The geometric assignment routed through wind is plotted in Section 3.4.3 and while it

is not too dissimilar to Section 3.4.3, only 14 formation pairs, roughly 13.33%, remain the same between the two assignments. There are clearly a few formations which are not ideal, such as the flight over Greenland, but as a first ‘estimate’ the geometric assignment is fairly promising. In fact, it will be shown in Section 3.5 that the initial geometric solution can be further improved by a simple iterative process at the cost of more computation time.

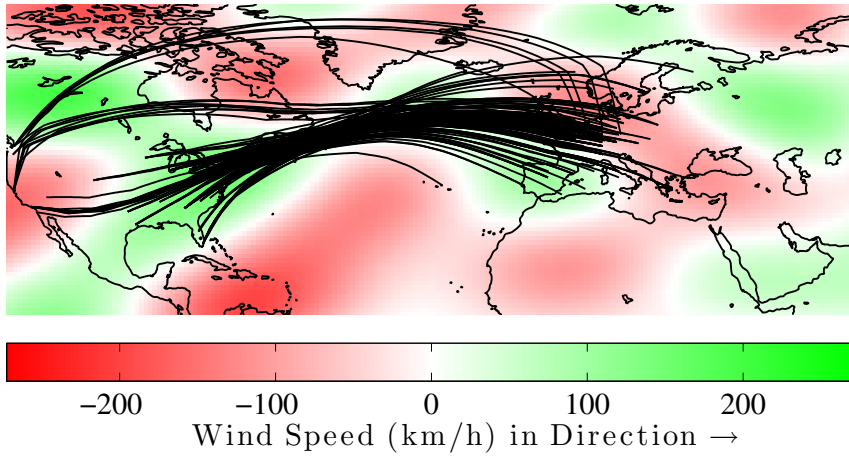
3.4.4 Jet-Stream East-to-West

Examining the results for the same set of flights for the same wind field but flights now travel East-to-West. The globally optimal wind solution results in a saving of 8.67% against the corresponding solo flights through wind, taking roughly the same time as Section 3.4.3. These routes are shown in Section 3.4.4, while the wind field is identical to that of Section 3.4.3, the green areas now act as headwinds travelling westerly and red areas act as tailwinds. What is clear is that the formations look to avoid those areas of large headwinds and aim to fly the regions of tailwind above and below the green-area over the Atlantic. These are clearly distinctly different routes from those of Section 3.4.3, with the more southerly paths flying significantly away from what would be the great circle path.

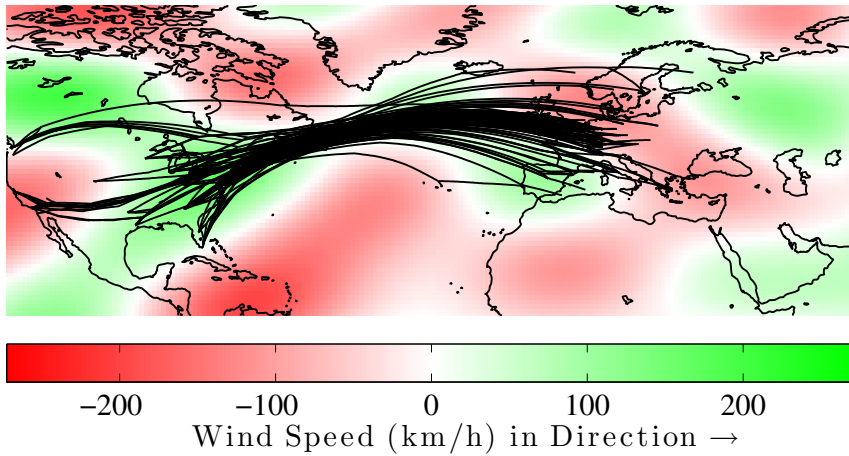
Again, if Method 2 is used, where the geometric routing is used for the enumeration and assignment stage with the wind-routing added as a post-process then this results in a saving of 7.45%. Less than the global optimal, using Method 1, of 8.67%, but a reasonable first estimate. The routes for the geometrically assigned formations are outlined in Section 3.4.4 and is not too different from the global optimal of Section 3.4.4. Between the two assignments, only 8 (7.62%) of the formations remain the same, showing that while the saving may be close, the actual allocation is almost entirely different.

3.4.5 More-Volatile Wind

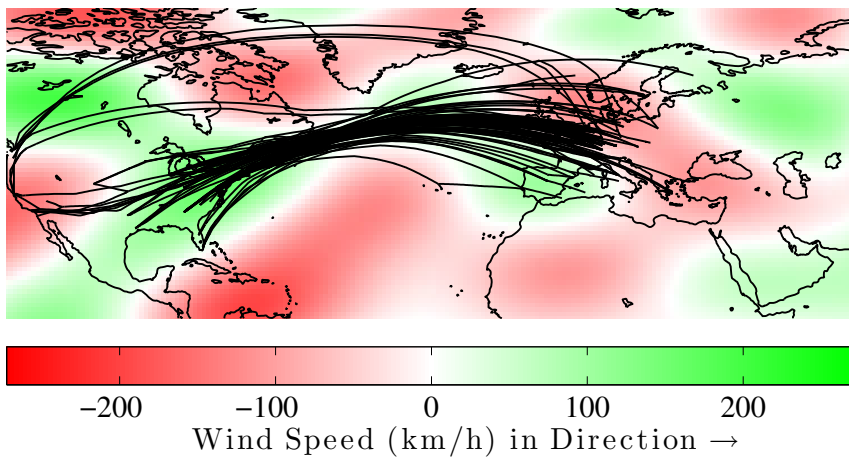
Finally now consider a much more volatile wind field where there are a greater number of distinct areas of differing winds with a larger difference in peak values (up to 300 km/h in any direction). The aim of using this more-volatile wind field is to observe what happens when the routes differ more significantly from great circle paths. Furthermore the results of this are only explored for flights travelling West-to-East.



(a) Solo Routes

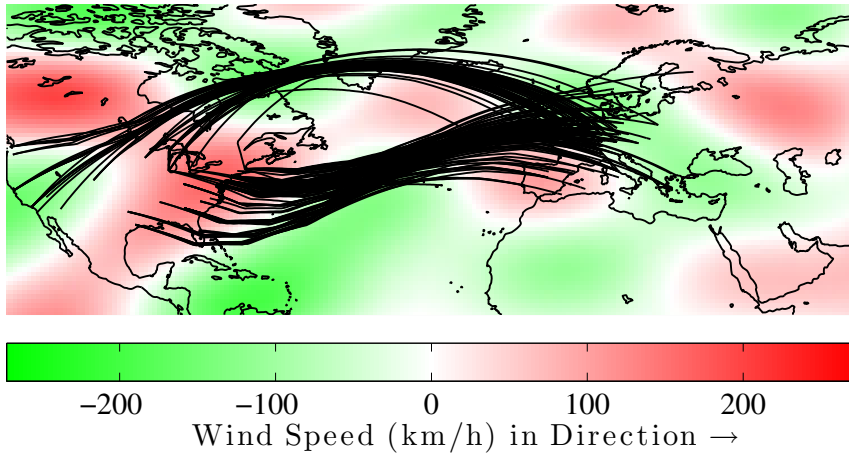


(b) Wind-Optimal Assignment

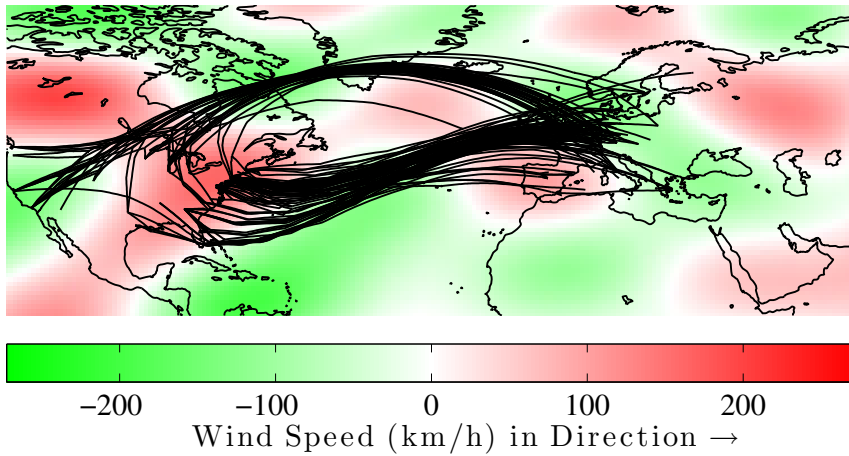


(c) Geometric-Optimal Assignment

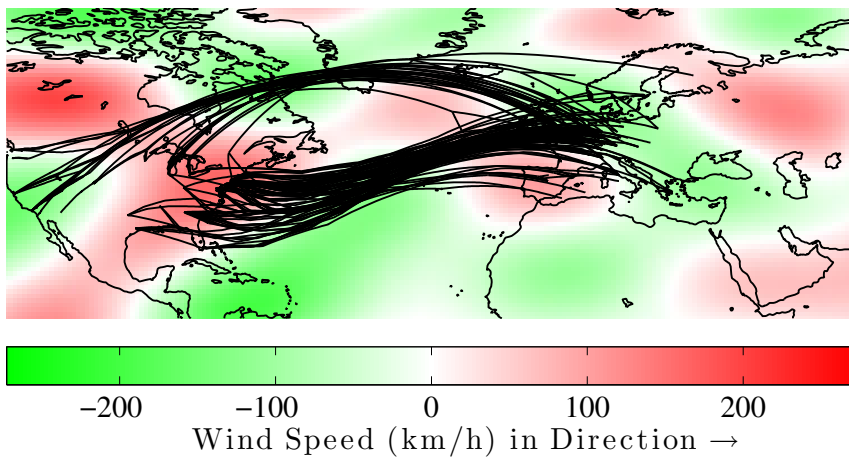
Figure 3.6: Wind Routes for Jet-Stream West-to-East



(a) Solo Routes



(b) Wind-Optimal Assignment



(c) Geometric-Optimal Assignment

Figure 3.7: Wind Routes for Jet-Stream East-to-West

Table 3.1: Case study comparison of savings

Wind Field	WC Assignment		GC Assignment		Difference
	Runtime	Saving (%)	Runtime	Saving (%)	
JS - East	180 hr	9.48	1 hr	8.31	1.17
JS - West	180 hr	8.67	1 hr	7.45	1.22
Volatile	180 hr	10.54	1 hr	9.22	1.32

Calculating routes for all possible formations through wind results in 10.54% saving against the corresponding solo flights. This saving is in fact greater than 10% due to the strength of winds in some areas effectively multiplying the formation benefit. The routes, plotted in Section 3.4.5, are a little more erratic than the less-volatile examples.

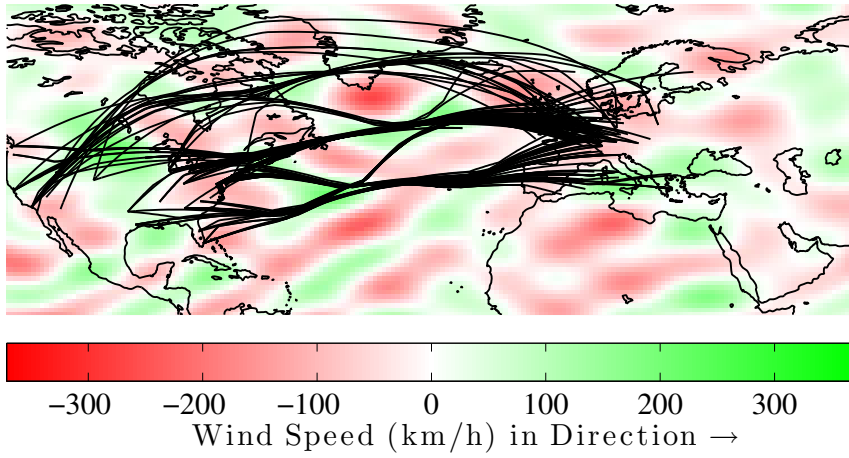
Using the geometric assignment provides a saving of 9.22%, again showing that the geometric methods act as a reasonable estimate to the global solution. The resulting paths, plotted in Section 3.4.5, are similar to Section 3.4.5 with only a few notable differences. Again, while the routes look fairly similar the actual allocations only have 5, or roughly 4.76%, formation pairs in common.

3.4.6 Comparison of Results

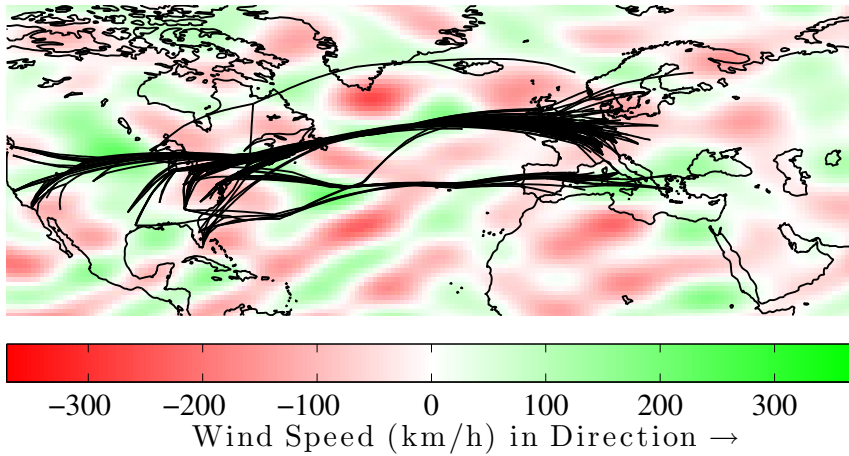
While the results of Table 3.1 are mostly clear, the actual percentage savings are more heavily influenced by the particular winds encountered. The level of disparity, between savings for wind-optimal and geometrically-optimal formation assignment are more important. Although it may not attain a global optimum for a more realistic model, it allows the use of the fast geometric approximation to estimate solutions to much larger problems in realistic time-frames (some 200 times faster). Once a smaller solution set is determined it can be post-processed to further improve the route to account for predictable wind patterns.

3.5 Improving the Geometric Assignment in the Presence of Wind

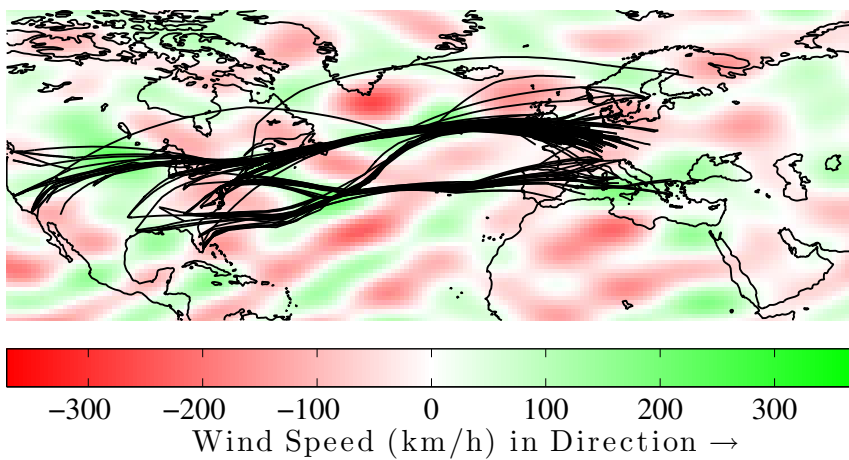
The aim of this chapter is to benchmark the geometric method for routing and assignment, as a way of trying to counteract the combinatorial impact of needing to look at all formation combinations, whereby an approximate routing method can be used to tackle the enumeration stage. The results of Section 3.4 show that the geo-



(a) Solo Routes



(b) Wind-Optimal Assignment



(c) Geometric-Optimal Assignment

Figure 3.8: Wind Routes for volatile wind field

metric method does act as a reasonable estimate to the assignment problem in the presence of wind. We now look to see if this initial estimate can be improved to get results closer to the global optimal with wind without having to do the entirety of the enumeration stage.

There are of course a number of different approaches that could be used to improve upon our initial assignment. The aim of this section is not to find the best approach, rather just show that it is possible to improve this assignment with a relatively small amount of extra work, with the assumption that in future ‘smarter’ approaches can be used. The method of this section will use the already known costs of formations and solo paths to create an estimated cost for those combinations which are unknown.

3.5.1 Estimated Assignment Method

After completing Method 2 of Section 3.2, using the geometric approach for the initial enumeration and assignment stage, we have knowledge (*i.e.* we know the values) of the following:

- The Geometric Cost of all 21,945 formation combinations
- The optimal geometric assignment (105 formations)
- The Wind Cost of the 105 assigned formations
- The Wind Cost of all 210 solo flights

To guarantee a globally optimal assignment using MILP for the wind routes, the cost of all 21,945 formations would need to be known. As Section 3.4 has shown, to calculate all these routes would require a significant amount of computation time. Instead, using the currently known values we look to improve the initial assignment.

The main obstacle this problem has is that in order to learn the value of another formation combination it needs to be routed through the wind, at a cost of around 30 seconds. Therefore, the number of extra route calculations needs to be kept low enough to allow a solution in reasonable time whilst learning enough of the other values to improve the current assignment.

To achieve this an ‘Estimated Assignment Method’ (EAM) is implemented to assign formations based on estimated formation costs. Firstly a cost-estimating function must be defined, recalling the notation for Wind Cost, WC, and Geometric Cost, GC, for a formation pair consisting of two flights, A and B , then the cost-estimating

function is as follows

$$\mathbb{F}(A, B) = (WC_{\text{solo}}(A, B)) \times \left(\frac{GC_{\text{form}}(A, B)}{GC_{\text{solo}}(A, B)} \right). \quad (3.6)$$

This estimating function takes the sum of the costs of the two solo wind routes and applies the same saving that is achieved by the geometric routes. The assumption being, that if the solo costs increase/decrease due to the wind routing so too will the formation routes. This is a fairly simplistic estimating function, which could easily be improved or even ‘trained’ as we learn more about the solutions, however this is beyond the scope of this thesis.

With the cost-estimating function in place the main idea of the Estimated Assignment Method will be as follows:

1. Create a list of costs used for the assignment stage. Made up of both the currently known formation costs and a number of estimated costs (Equation (3.6)) for each of the unknown values.
2. The MILP is run for these mixed-costs.
3. If the assignment includes any formation which only has an estimated cost, those costs must be properly evaluated for wind and the cost list is then updated.
4. If the cost list has changed go to step 2, otherwise we are finished.

This process of estimating and assigning is then simply iterated over until either the list of costs no longer changes (*i.e.* all assigned formation costs are known) or until some timing constraint is met. This is a fairly simple approach but allows us to improve the initial assignment without having to enumerate all the combinations for wind. Therefore given an cost-estimating function and a total allowable computation time this algorithm can be run.

3.5.2 Results of Using an Estimated Assignment Method

The method outlined in Section 3.5.1 is now applied to the three case study examples of Section 3.4. While each of the case studies has a different set of solutions, the Jet-Stream West-to-East shares the same geometric costs and geometric assignment as the Volatile wind-field West-to-East.

We assume that the computational-cost of calculating a wind-route is 30 seconds per formation. Then for each iteration of the assignment loop the amount of added

Table 3.2: Comparison of three assignment methods and their corresponding saving and cpu time

Wind Field	Method 1		Method 2		Method 2 + EAM	
	Saving	Time	Saving	Time	Saving	Time
JS - East	9.48%	180 hrs	8.31%	1 hr	9.28%	17 hrs
JS - West	8.67%	180 hrs	7.45%	1 hr	8.43%	38 hrs
Volatile	10.54%	180 hrs	9.22%	1 hr	10.04%	12 hrs

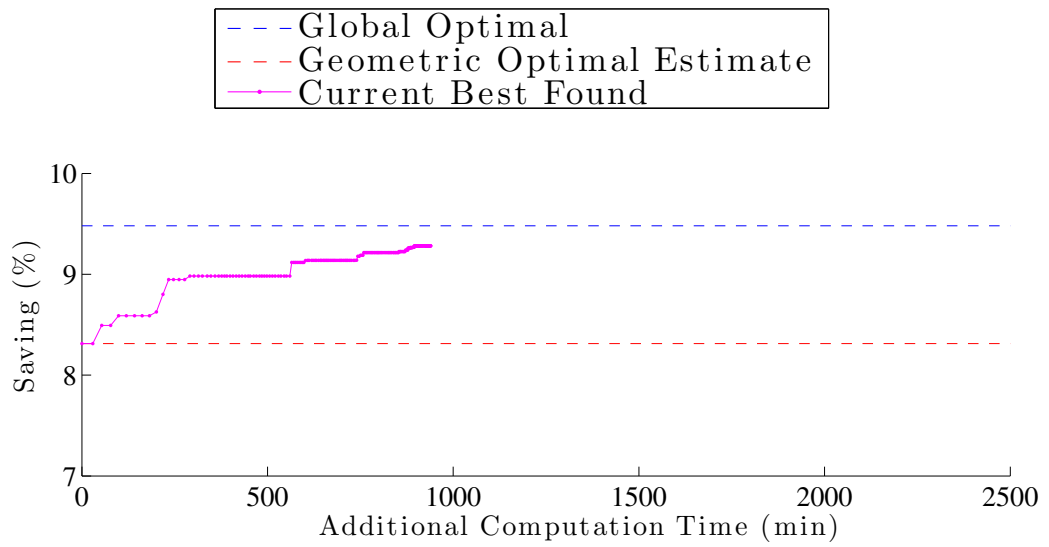
Table 3.3: Comparison of three assignment methods and the number of formations matching the globally-optimal assignment

Wind Field	Method 1	Method 2	Method 2 + EAM
JS - East	105 (100%)	14 (13.33%)	43 (40.95%)
JS - West	105 (100%)	8 (7.62%)	34 (32.38%)
Volatile	105 (100%)	5 (4.76%)	15 (14.29%)

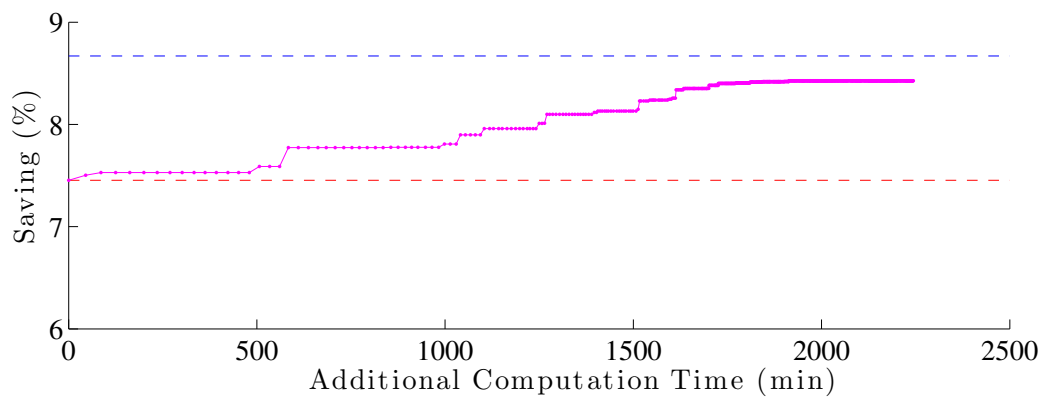
solution time is the sum of the time it takes to run the MILP plus 30 seconds for every estimated cost that needs a WC. Running the MILP is fast relative to the time it takes to calculate the WC for a formation and usually only requires 2-3 seconds each time. Therefore the majority of the computation time is down to the number of new WCs needing evaluation. For these case studies the method is allowed to run up until the total calculation time exceeds 2500 minutes, around 1/4 of total enumeration. While this is still around 41 hours, it is equivalent to about 5000 WC calculations against 21,945 for the total enumeration. This duration obviously also scales with the runtime of the wind routing, and improved wind-routing method would in turn reduce this computation time.

The results of running the iterative assignment loop for each of the case studies is outlined in Table 3.2, showing very promising results. As shown in Figure 3.9 each solution begins at the assignment cost estimated by the Geometric approach and gradually improves as more wind routes are evaluated. The currently best found assignment cost is plotted against the computation time required. As the algorithm progresses more wind costs are ‘learned’ and the MILP can get closer to the global optimal assignment. None of the cases use all the allotted time and all finish when the current assignment can no longer be improved by using any of the estimated values.

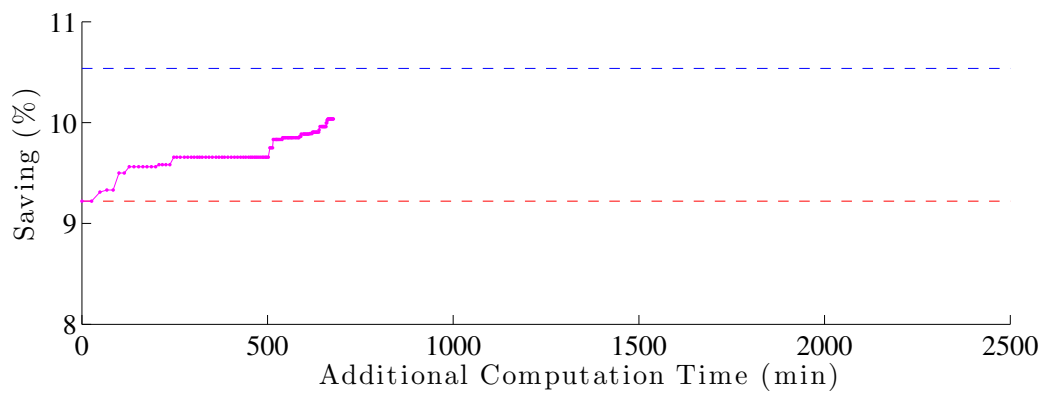
The Jet-Stream West-to-East results, discussed in Section 3.4.3, begin with an initial estimated assignment saving of 8.31%. As shown in Section 3.5.2 and Table 3.2 the estimated assignment method proves to be very helpful in improving the assignment solution in the presence of wind, reaching an assignment with a saving



(a) Jet-Stream West-to-East



(b) Jet-Stream East-to-West



(c) Volatile Wind West-to-East

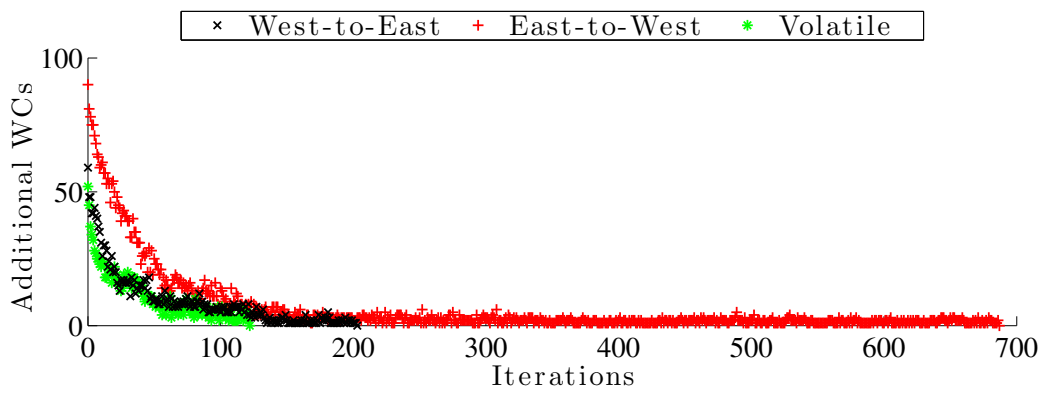
Figure 3.9: Estimated assignment method solutions

of 9.28%, which is a percentile difference of just 0.20 away from the global optimal. Getting to this assignment would however require 17 hours of computation time (this includes the 1 hour for the initial GC assignment), but is significantly faster than the full enumeration. There were 204 iterations of the EAM to reach this solution taking 6 minutes, which is negligible relative to the time required to calculate the wind routes. What can also be seen from Table 3.3, is that the final estimated assignment only contains 43, roughly 40.95%, of the same formations as the global-optimum. This shows that while the two assignments' costs are relatively close, there is still a reasonable amount of difference between them.

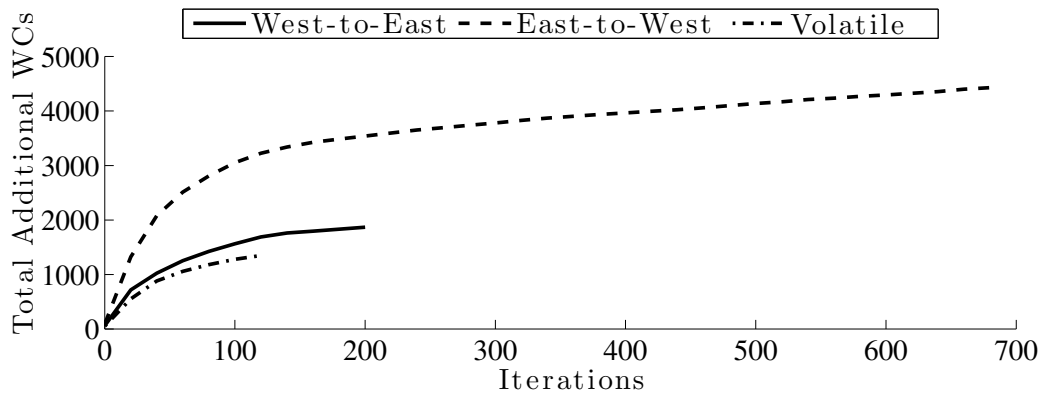
The East-to-West results, Section 3.5.2, starts at an initial assignment of 7.45% and gradually improves until its final solution of around 8.43%. This takes almost twice as long as the West-to-East at around 38 hours but is still significantly faster than the full enumeration. For around one fifth of the computation time you can get to within 0.24 percentile difference of the global solution. The EAM required 688 iterations to obtain this solution, which is clearly many more than the West-to-East, taking a total of 25 minutes. Furthermore on average each iteration took slightly longer than the West-to-East case. This indicates that the problem the MILP was solving each time was 'harder', typically this is a result of a 'bumpier' solution space with many local minima. Finally, the number of formations which are the same between the estimated assignment and the global-optimum, from Table 3.3, is 34 (32.38%).

The West-to-East volatile wind solutions shown in Section 3.5.2 settles to a solution faster than the other two cases. Starting from the initial assignment saving on 9.22% and within 12 hours reaches a saving of 10.04%. Being within 0.50 of a percentile from the global optimal is still a greatly improved result, but of the three case studies it has the biggest difference. This could be due in part to it settling quickest, after 12 hours and 123 iterations, resulting in the equivalent of getting stuck in a local-minimum. The resulting estimated assignment only has 15 (14.29%) in common with the global assignment, again showing some way to go until the global assignment is reached.

There is a clear trade-off between the saving achievable and the time required to reach the solution. At one end there is the Method 1, taking around 180 hours but resulting in the maximum amount of saving possible. At the other end there is Method 2, with the initial assignment provided by the geometric method with wind routing added as a post-process, taking around 1 hr. While in between there is a reasonably steady progression from the geometric initial estimate towards the global



(a) Additional WC required per iteration



(b) Total Additional WC Required

Figure 3.10: Estimated assignment method's addition of Wind Costs

optimum. The values depicted in Sections 3.5.2 and 3.5.2 compares the number of iterations of the Estimated Assignment method versus the number of additional Wind Cost calculations required. What can be seen is that during the first few iterations lots of estimated costs are chosen and so lots of WC need to be evaluated. Many of these will correspond to formations which save a lot and stand out against other possibilities, while some will be those with lower savings that are a result of moving around other flights within the assignment. The number of addition WCs needed per iteration then gradually decreases until no changes are made and the process stops.

Due to the heuristic nature of the estimated assignment method no level of solution improvement can be guaranteed. However, the results presented show a clear indication that an initial assignment can be readily improved. The cost estimating function, such as Equation (3.6), plays a key role in the performance of the Estimated Assignment Method. A function which over estimates the costs (*i.e.* predicts larger values) will mean less movement from the original assignment as there will be less incentive for the MILP to choose other formations. Conversely, under estimating the costs will result in the estimated assignment method choosing lots of the estimated costs due to them being lower. Choosing many new estimated solutions requires the algorithm to then go and calculate the wind cost, which results in a longer runtime. Tuning heuristic algorithms, such as this, is a large area of research in itself and work into improving the EAM could be useful. However, the results of this section show that even with a fairly simplistic cost-estimating function significant improvements to the initial assignment can be made.

3.6 Summary

This chapter has investigated the impact of wind on the problem of routing for formation flight and how well the geometric assignment method performs as an estimate to the global solution.

Firstly a numerical method for optimal formation routing in the presence of a static wind field is presented. The wind routing method involves using a number of variable way-points to estimate the flight paths for two aircraft looking to join in formation. The way points are optimally chosen using an active-set optimisation within matlab and provide a reasonable method for formation routing in the presence of wind. The aim of the method was to provide fairly simple routing when more-analytic approaches are unsuitable.

The implementation of the Geometric method has been motivated by the combi-

natorial nature of calculating all combinations for formation flights. The numerical approach of the wind-routing method means it is not particularly fast. With a runtime of about 30 seconds per formation, calculating all combinations requires significant computational time. The core aim of this chapter has been to benchmark the geometric method, outlined in Chapter 2, to see if the enumeration and assignment stage can be achieved using only the geometric method, while the more complex wind routing is left to a post-process.

With this in mind two separate workflows were presented for assigning aircraft into formation fleets in the presence of wind. The first, requires the calculation of the cost of the wind route for all combinations, followed by a MILP assignment. This produces a globally optimal solution but is a significantly more computationally intensive process. The second approach is to use the fast geometric method to estimate the highly-combinatorial assignment process and then simply calculate the wind-routes as a post process.

The two approaches were compared in Section 3.4 using the Transatlantic case study for three different wind routing problems. The results showed that the geometric approach produces an assignment which performs very well, around a 1 percentile difference from the global optimal, for a radical reduction in the required computation time.

An Estimated Assignment Method for improving this initial assignment was also introduced in Section 3.5. Here an cost-estimating function was used to try to estimate the wind-cost of formation routes, yet to be calculated, using only the knowledge of the solo wind-cost and the saving achieved using the geometric method. The assignment stage became an iterative process making optimal assignments based on a mixture of known and estimated costs. If an assigned formation only had an estimated cost then the wind cost was then calculated, the costs updated and the MILP was re-run. This method therefore allowed the improvement of the initial assignment in exchange for additional computation time. The results of the Estimated Assignment Method were extremely promising, creating assignments which differed from the global solution by as little as 0.2 of a percent while taking significantly less computation time.

The benchmarking results of this chapter show that the Geometric Approach proves a very reasonable estimate to the global assignment problem. This essentially counteracts the large combinatorial impact of enumerating all combinations of formations. Therefore, more computationally challenging routing, such as the wind routing method of this chapter, could feasibly be left for a post-process stage.

Chapter 4

Mitigating the Impact of Ground Delay on Formation Flight

4.1 Introduction

The core routing approach of this thesis has been that of time-free solutions, whereby takeoff and landing times should be chosen to adhere to the optimal formation route. However, once these are set, it is clear that any rendezvous would need to be carefully timed and coordinated. Therefore the impact delay could have on such an operation is of major interest.

Delays were estimated to cost the European airline industry 1.25 billion Euros in 2010 [93], with weather and airport operations contributing significantly. Such delays will always be a possibility and any commercial flight is at risk from being affected. However, when trying to design rendezvous operations such as formation flight, timing becomes a significant factor.

With many commercial aircraft flying between 300-450 knots during cruise, missing the rendezvous location by a minute can mean spatially missing it by 10-15 km. Therefore as the level of delay increases the potential fuel saving from formation flight decreases. If the aircraft simply ‘waits’ at the formation point, flying a holding pattern, it will burn through any savings. Similarly, if the aircraft continues along its path alone, the distance required to ‘catch-up’ and rejoin formation will also increase. Any such catch-up manoeuvre will cause loss of performance compared to the ideal formation flight. The combination of this speed change along with the section of the formation fuel saving ‘lost’ means that any attempt to regain formation needs to be carefully costed and weighed against other possible solutions.

Delay can occur at any stage of the flight for a number of reasons. A significant proportion however, occurs at the airport (due to factors such as airport congestion [94]) known as ground delay. For the purpose of this chapter, we assume that the takeoff time is uncertain but that all subsequent operations are perfect. The approach taken could potentially extend to uncertainty in en-route flight such as turbulence, but this is left for future work. The main focus of this work is therefore addressing the impact of a delay in takeoff has on other formation members.

Formation routes are first calculated using the methods of Chapter 1. The routes are spatially considered to be fixed, thus once an aircraft commits to a formation it must fly the geographical route regardless. Furthermore, formations of two aircraft are used while assuming a constant fuel burn discount of 10%, of Table 2.1, during formation [21, 26, 27, 40]. The concept presented could also extend to formations of more than two aircraft, with an added dimension for each additional aircraft, however this would also cause increases in computation time.

This chapter will compare two different approaches to handling ground delay. Firstly a simple holding pattern approach is modelled, whereby one flight ‘waits’ at the rendezvous location by entering a holding pattern until the other arrives. The second method is to use a state space approach, using Dynamic Programming to calculate optimal policies for any possible realisation of delay. The idea being that the first approach will act as a benchmark for how well the second approach performs.

4.2 Probability Density Functions of Airport Ground Delay

In order to calculate the probabilities of a particular delay occurring Probability Density Functions (PDFs) have been fitted to historical data [95]. The set of US airports used within the Transatlantic case study of Section 2.6 was chosen and the scheduled and actual takeoff times were recorded for the month of October 2013. Discrete Negative Binomial Distributions (NBDs) were chosen as a reasonable fit, to the data (given more data, better statistical models could also be used).

Table 4.1: NBD n_r and p values for four US airports

Airport	Early		Late	
IATA	n_r	p	n_r	p
ATL	3.371	0.478	0.858	0.029
BOS	2.702	0.365	0.755	0.024
MIA	2.372	0.349	0.751	0.025
PDX	2.187	0.319	0.573	0.017

4.2.1 Negative Binomial Distributions

Given a succession of independent Bernoulli trials, each having a probability of success p and probability of failure $1 - p$, then the number of trials needed in order to observe a given number n_r of successes defines a NBD. For some n_r and p the NBD is then defined as

$$f(k|n_r, p) \equiv \mathbb{P}(X = k) = \frac{\Gamma(n_r + k)}{\Gamma(n_r)\Gamma(k + 1)}(1 - p)^{n_r}p^k, \quad (4.1)$$

for $k \in \mathbb{N}^0$, where $\Gamma(x) = \int_0^\infty e^{-t}t^{x-1}dt$.

The Gamma function $\Gamma(x)$ is an interpolation of the binomial coefficients, allowing for non-integer values of n_r (Table 4.1 gives some typical n_r values).

4.2.2 Modelling Airport Ground Delay

While the majority of delay occurs from aircraft taking off later than scheduled, some aircraft may also takeoff early (early-delay), but not to the same extent, in fact it is common for aircraft to takeoff between 0-15 minutes early. The distribution of this early-delay can also be modelled with a NBD, however it usually has a slightly different form to that of ‘late-delay’. The resulting final PDF is therefore the normalized combination of two NBDs producing an asymmetric curve peaking at 0 minutes (scheduled time) and rapidly decreasing at either side. The majority of the probability of takeoff lies, as you might expect, within 30 minutes either side of scheduled takeoff time, however it is not uncommon for aircraft to be delayed upwards of 60 minutes.

Table 4.1 contains example n_r and p values used for the four US airports: Hartsfield-Jackson Atlanta International (ATL); Boston Logan International (BOS); Miami International (MIA); and Portland International (PDX). The corresponding NBDs are plotted, between -60 and 120 minutes, in Figure 4.1 for time difference in minutes of

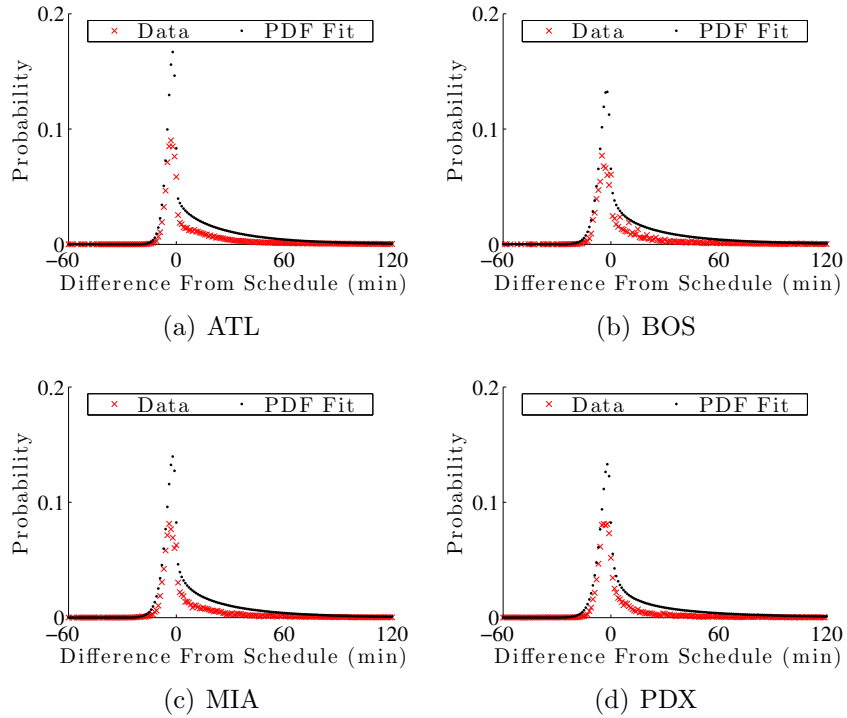


Figure 4.1: Negative Binomial Distribution fits to real delay data

observed takeoff time against scheduled takeoff time (note, while they are only plotted up to 120, the NBD is defined up to 600 minutes). As the NBD is a discrete PDF, each point on the graph then represents the probability of takeoff occurring within the corresponding minute interval. What can be seen is that all the distributions can be reasonably characterised by the use of a two-sided NBD, however it is not an exact match, as the NBD fit has a tendency to overshoot at the peak.

4.3 Holding Pattern Approach

With the delay PDFs now outlined by Section 4.2, we can look at ways of mitigating the effects of these delays. One way to try and cope with the effect of ground delay on a formation is for the first aircraft arriving at the rendezvous location to simply wait for the later aircraft, as shown in Figure 4.2. The first aircraft takes off and flies to the predefined rendezvous location (Figure 4.2(b)). If the other aircraft is delayed, then the first aircraft performs a holding pattern and waits for the other to arrive. Once the other aircraft reaches the rendezvous point, Figure 4.2(c), then they enter a formation, continuing for the whole of the formation leg (Figure 4.2(d)).

This is a somewhat naïve approach as it is entirely reactive to any delay. A minute of delay will then be absorbed as the cost of the aircraft flying in a holding pattern speed for 1 minute. This seems like a reasonable approach when encountering small delays (< 15 minutes), but as the delay increases so too does the cost to the holding aircraft and could results in significant losses.

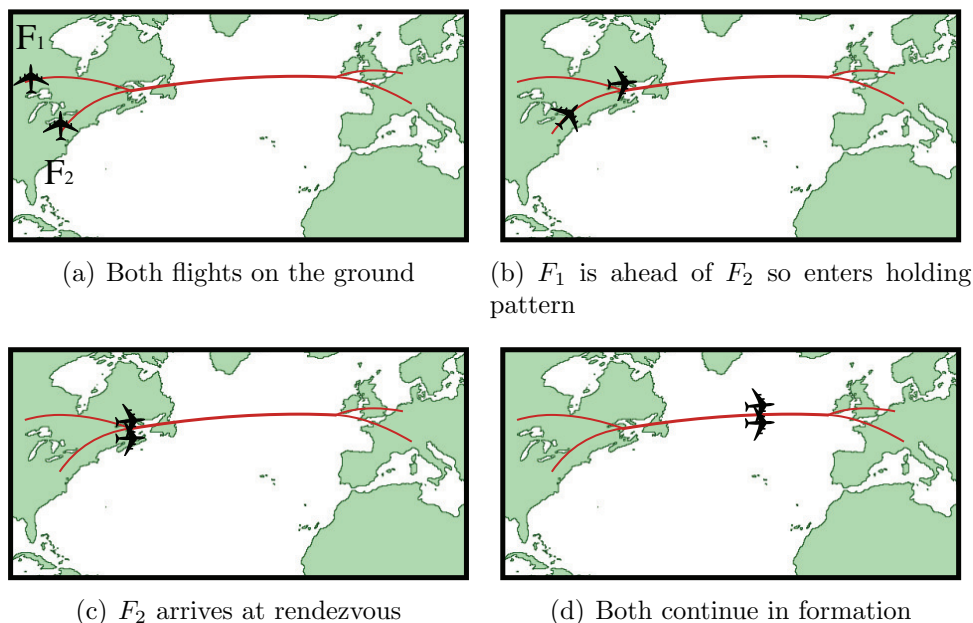


Figure 4.2: Geographical representation of the Hold Approach

Very roughly speaking for two 8 hour flights joining in formation and saving 10% (the theoretical maximum from Table 2.1) each would equate to 48 minutes of additional flight time at cruise per aircraft. Therefore at the limit (ignoring additional fuel weights) one flight holding could tolerate up to about 96 minutes of delay until the entire formation would just break-even. However, if the other aircraft was delayed more, then the entire formation would start to cost more than the corresponding solo flights. Further still if some cutoff was set, so that a flight could only hold for a maximum of say 90 minutes before it had to continue on, then if the other flight was delayed 91 minutes the entire formation would never get the savings from formation flight and end up instead costing 10% more than just flying solo. Due to this, and as aircraft realistically should not spend hours holding, it is clear some cutoff time needs to be set.

4.3.1 Holding Pattern Cutoff Time

The cutoff choice will be an important factor in the overall performance of the hold approach. If the cutoff is too low only a few flights will achieve formation, and therefore any saving, while all the others will incur the cost of holding, receive no formation saving and cost more than solo flight. Conversely if the cutoff is set too high, while lots of flights will achieve formation (with mixed savings corresponding to the level of delay), some will achieve huge penalties for holding and thus incur large losses. Furthermore this does not take into account the knock on effects of greatly increased flight times.

For a given cutoff time t_{co} and delay t_{del} the total cost is:

$$\text{Total Cost} = \begin{cases} \text{Cost(Holding for } t_{del}) + \text{Cost(Formation)} & \text{if } t_{del} \leq t_{co} \\ \text{Cost(Holding for } t_{co}) + \text{Cost(Solo)} & \text{if } t_{del} > t_{co} \end{cases} \quad (4.2)$$

This kind of cost function produces a discontinuous jump in the cost as the level of delay exceeds the cutoff time (an example of this is shown in Figure 4.3)

Realistically each flight of each formation would have a distinct cutoff time assigned, individually tailored to suit each particular aircraft and formation scenario. However for simplicity and clarity of results it is assumed that a single cutoff time will apply to all flights wishing to join a formation.

4.3.2 Example Formation

To illustrate the Hold approach, take as an example a formation between the two flights:

Flight 1: Los Angeles International Airport (LAX) to Heathrow Airport (LHR)

Flight 2: Phoenix International Airport (PHX) to Heathrow Airport (LHR)

An exploration of what the cost function looks like for a variety of cutoff times for this sample flight is shown in Figure 4.3. The horizontal axis depicts the relative delay between the two flights, Flight 1 (F_1) and Flight 2 (F_2). That is, the difference in time of Flight 1 arriving at the rendezvous location and Flight 2 arriving, *i.e.* a negative value indicates Flight 2 arrives after Flight 1. The y-axis is then the fuel saving corresponding to such a delay. As the relative delay moves from 0, the fuel

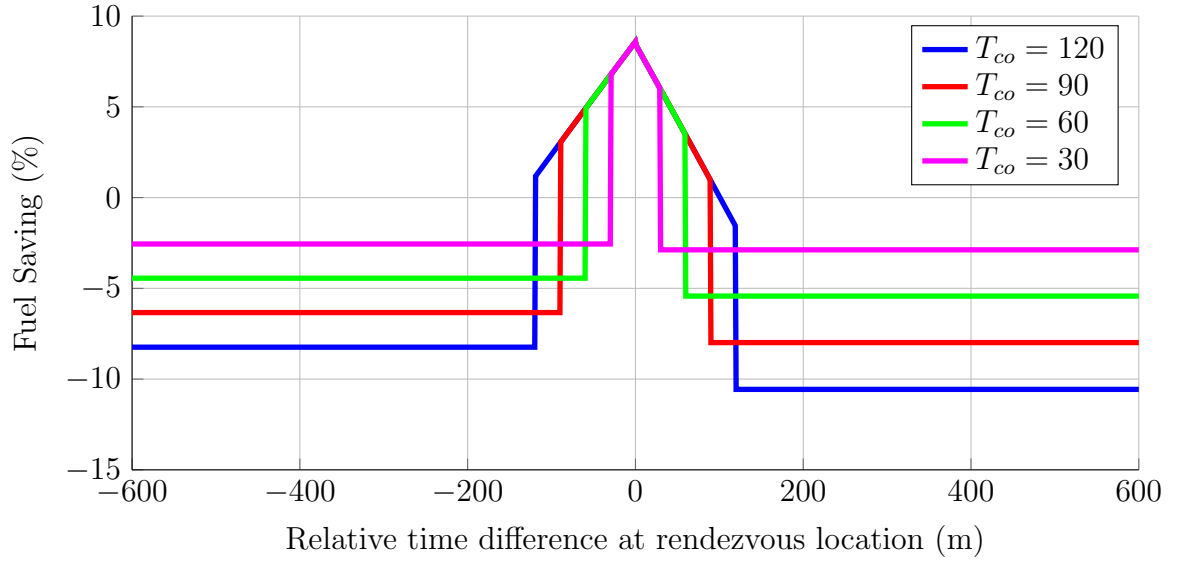


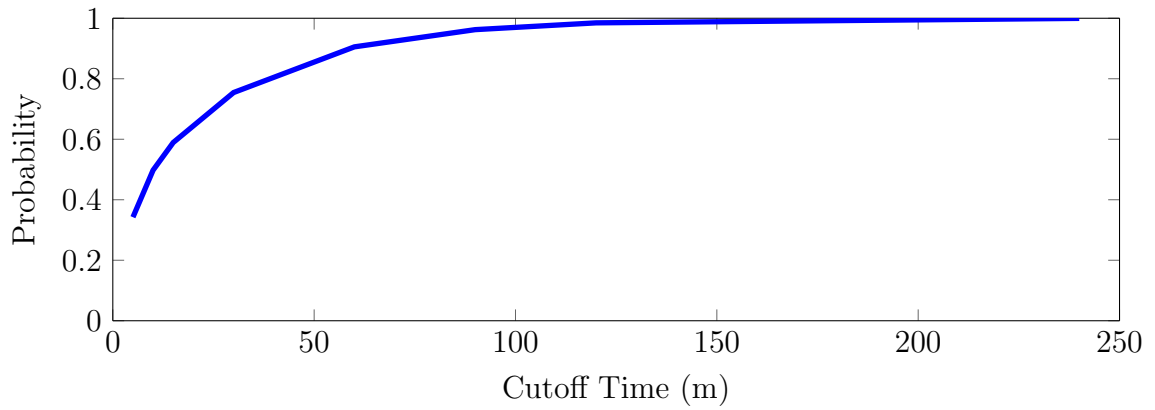
Figure 4.3: Example saving for all possible relative delays between Flight 1 and Flight 2 for different cutoff levels

saving percentage gradually decreases until the cutoff time is reached, at which point it switches to a much lower, negative saving.

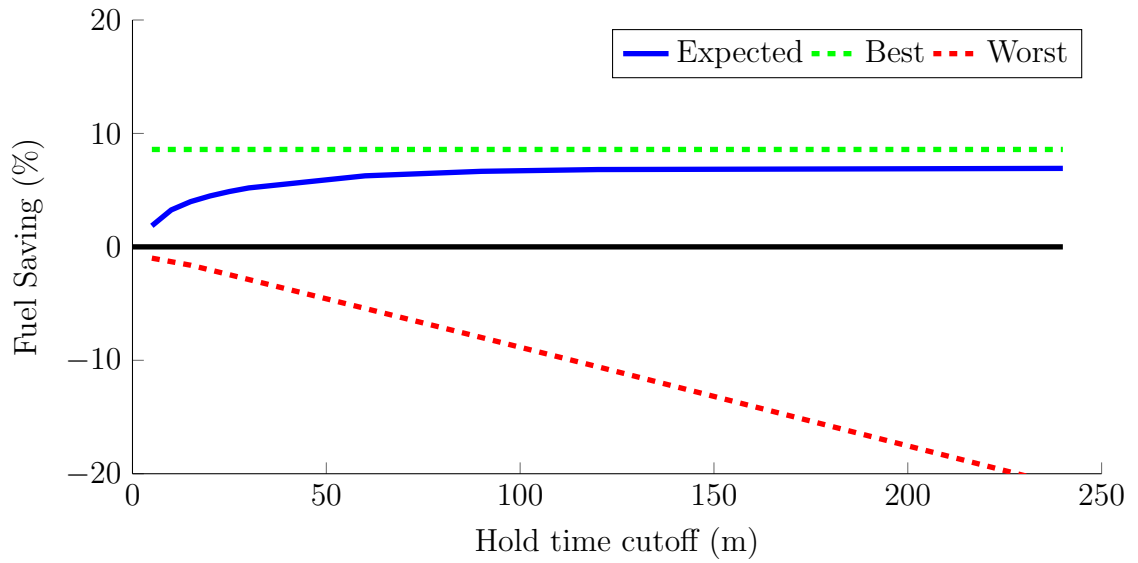
The likelihood of the relative delay lying within a given cutoff time, shown in Figure 4.4(a), is the cumulation of both airports' PDFs (from Section 4.2). Although this probability tends to 1 as t_{co} increases, this probability has to be weighted against the risk of also making huge losses. In this example, even with a cutoff time of 60 minutes, there is still roughly a 10% chance of being outside it.

The results for varying cutoff times t_{co} are plotted in Figure 4.4(b). There are three different values plotted: the probabilistic expected saving; the best possible saving achievable; and the worst possible saving achievable. As the cutoff time increases, the likelihood of both aircraft taking off within that window also increases as shown in Figure 4.4(a). The result of the potential savings are interesting as increasing the cutoff time forever improves the expected saving at the expense of forever decreasing the 'worst case' saving. The reason for this increase to expected cost is that the expected cost is essentially the multiplication of Figures 4.3 and 4.4(a). The cost function of Equation (4.2) is piecewise linear with the cost of no formation being a significant jump. As the cutoff increases the probability of the aircraft being outside the cutoff window approaches zero. As a result the impact on the overall cost of jumping to the much worse cost diminishes as t_{co} increases.

The cutoff time though, in reality, would most likely be determined by external factors such as regulatory considerations (*e.g.* divert rules). One could realistically



(a) Probability of relative delay between Flight 1 and Flight 2 being within the cutoff time t_{co}



(b) Effect of cutoff time on range of saving for optimally assigned formations

Figure 4.4: Example saving and cumulative probabilities for LAX-LHR and PHX-LHR

imagine cutoff times of anywhere up to about 60 minutes. The Hold approach is now fully defined, the results of applying this approach to a case study will be shown in Section 4.10.1. A State Space approach will now be introduced as an alternative way of handling ground delay. The two methods will be compared in Section 4.10.3, whereby it will be shown that the State Space approach is far superior.

4.4 A State Space Approach

An alternative approach to a simple hold pattern is to model the problem using state-spaces and two types of Dynamic Programming (DP) which are then solved using Value Iteration (VI). Dynamic Programming (DP) is used to solve the deterministic region of the flight (*i.e.* when all aircraft have taken off). Then through Stochastic Dynamic Programming (SDP) [96] the uncertainty of ground delay is assessed via the assignment of optimal speed-policies, defining what speed airbourne aircraft should fly, for any possible realization of delay. Finally the results of a case study of 210 Transatlantic flights are compared and the results discussed.

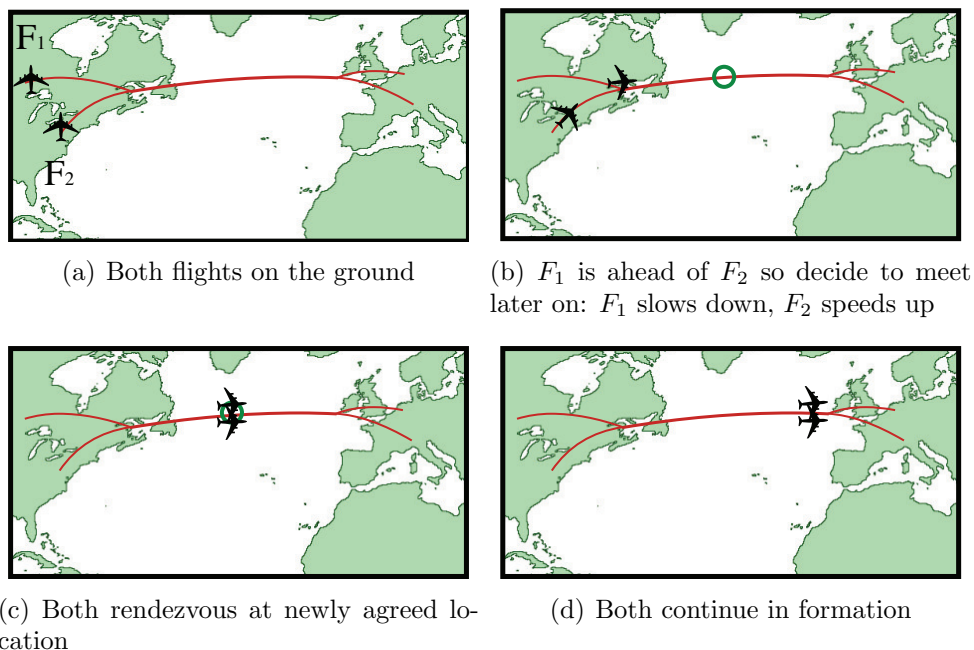


Figure 4.5: Geographical Representation of the State Space approach

This approach differs from the Hold Policy of Section 4.3, shown in Figure 4.2, as airbourne aircraft will instead continue along their route, flying at a certain speed,

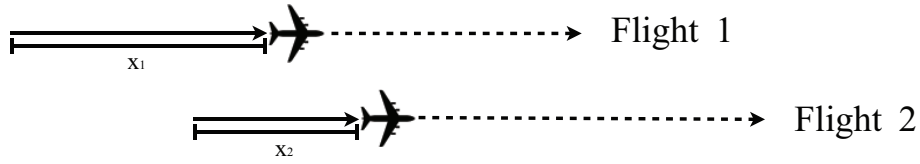


Figure 4.6: The two state variables x_1 and x_2 of aircraft location

which will be optimally determined by the method outlined in this section. A geographical example of what happens during this approach is shown in Figure 4.5. The first flight to take off continues along its route until the other flights delay is realised. If no delay occurs then they will meet at the normal rendezvous location. However if one flight is delayed then the other flight will continue along its route until the other aircraft takes off. As soon as both aircraft are airbourne a new rendezvous location is calculated, the circle in Figure 4.5(b), which minimises the total fuel burn. The flight in front would slow down, to wait, while the flight which is behind would speed up in order to ‘catch up’. The two flights then rendezvous at the new location, Figure 4.2(c), and fly in formation for the rest of the formation leg, Figure 4.2(d).

4.4.1 Formation and Non-Formation States

Fixing the geographical route removes the dimensions of varying the longitude and latitude locations. Instead only the current distance each aircraft has flown along its own path is varied. The location of each aircraft is therefore reduced to being implicitly defined by a one-dimension state variable. For the two aircraft formation case, where a flight F_1 and a flight F_2 takeoff from two distinct airports. We are said to be in a state $(x_1, x_2) \in S$ if flight F_1 and flight F_2 are x_1 and x_2 km along their respective paths (as in Figure 4.6). The specific longitudes and latitudes can simply be recovered from the already defined route.

A subspace $S_F \subseteq S$, the formation section, is defined to be the possible states that result in both aircraft being at the same geographical location and can therefore fly in formation (the diagonal line in Figure 4.7). A solution can transition from the non-formation state S to the formation state S_F and begin to receive the fuel-reduction benefits of formation flight.

The final state space is also discretized into a number of sample grid points, each of which will need to be evaluated and costed.

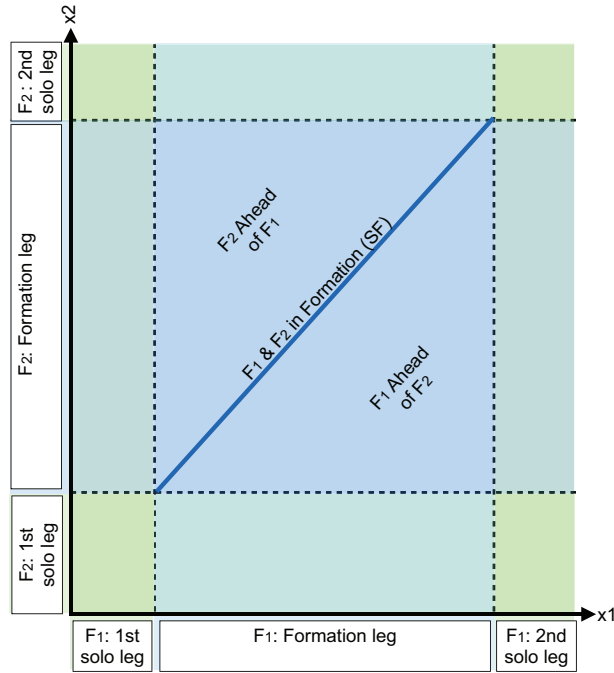


Figure 4.7: Regions within the State Space

4.4.2 Moving Through the State Space

A graphical representation of this space is shown in Figure 4.7. From any point in this space, a positive horizontal movement means flight F_1 has moved along its trajectory and a positive vertical movement means flight F_2 has moved along its trajectory. Any other movement, is a combination of the two, and so both are travelling along their paths at some given speed. The dynamics of moving through the state space can be defined as:

$$\begin{aligned} \dot{x}_1 &= V_1 \\ \dot{x}_2 &= V_2 \end{aligned} \tag{4.3}$$

Assuming no time constraints, during solo flight each aircraft will fly at its best speed (*i.e.* the speed that minimizes its total fuel burn). The ratio of the aircrafts' solo-speeds describe a single nominal solo-gradient to move through the state-space. Therefore if it is decided that the aircraft should fly solo, then the path through the state-space will attempt to closely follow this gradient.

If an aircraft misses their rendezvous (and so are not yet in formation) then this implies one of the aircraft is further along the route than the other. Thus if one aircraft is ahead, by adjusting the speeds (a decrease for the aircraft that is ahead and an increase for the aircraft behind) a 'catch-up' is performed and the formation

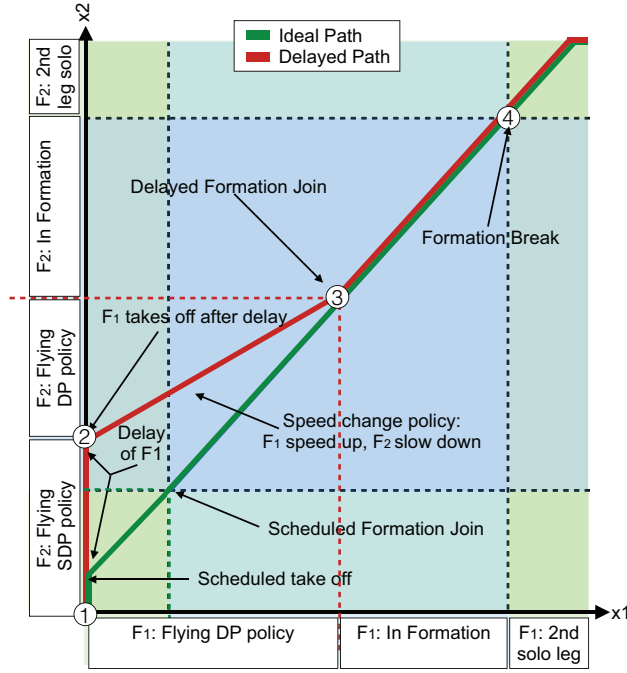


Figure 4.8: Solution example with delay. Between 1 and 2 F_2 flies the SDP policy as F_1 is delayed. Between 2 and 3 both aircraft fly the DP policy, to meet at 3 and fly in formation until 4 where the break and fly solo.

joins at a later time (as outlined in Figure 4.8).

For any given state one can also define a reachable region, which are the future states which can be realistically reached through the ratio of possible velocities each aircraft can fly. These are defined by the gradients $m_A = \frac{V_{1min}}{V_{2max}}$ and $m_B = \frac{V_{1max}}{V_{2min}}$ and are the extremes of the aircraft speeds. The two lines defined by the two gradients m_A and m_B along with the current state (x_1, x_2) define this region,

$$R_r(x_1, x_2) = \{(x'_1, x'_2) : m_A \geq \frac{x'_2 - x_2}{x'_1 - x_1} \geq m_B\}. \quad (4.4)$$

The shape of this region will be defined by the relative performance characteristics of each aircraft. In principle this region extends infinitely, however it is only necessary to look a small amount ahead.

The cost of reaching any point in this space will vary with the aircrafts' velocities. Given the current state and a chosen next state will define a single gradient m , relating the two aircraft velocities V_1 and V_2 , of flight F_1 and flight F_2 respectively. In order to successfully get to the next state the relationship $V_1 = m \times V_2$ must be satisfied. Importantly, this relationship dictates the ratio but not the individual speeds. For example if $m = 1.1$, then as long as F_1 flies 10% faster than F_2 then they

will meet at the desired next state. Therefore, since each speed can be defined as a function of the other *i.e.* $V_2 = f(V_1)$ and similarly $V_1 = f(V_2)$, to find the optimal pair of speeds, it suffices to simply minimise with respect to one of the speeds. A one dimensional search over V_1 (or analogously V_2) can therefore find the optimal pair of speeds, which satisfies the ratio m and minimizes the total cost. As a result it is possible to calculate the minimum cost to get between all possible reachable states, along with the corresponding speeds.

An example walk-through of a possible movement through the state-space is shown in Figure 4.8. The deterministic DP and the Stochastic DP have first been solved and we are following the resulting policies. The solution starts at point 1, when both aircraft are on the ground, and as flight F_2 takes off we begin to move along x_2 . Throughout the time F_2 is airborne and F_1 is still on the ground, the problem is uncertain and the solution to the SDP will provide the policy for F_2 to follow until F_1 eventually takes off. This optimal speed-policy will be followed until flight F_1 takes off at point 2 and the level of delay will be fully realised. From this point on the problem is deterministic and the solution policy calculated by the DP will be used. The DP policy tells flight F_2 to slow down, to try and wait, and tells flight F_1 to speed up. Given this speed change policy, the aircraft should eventually rendezvous at point 3, about half way along the formation leg of the journey, and continue in formation until they break away at point 4. Therefore, the relative increase in cost from F_1 flying faster and F_2 flying slower has been counteracted by the fuel saving achieved by flying in formation, albeit for less than the scheduled amount.

With this in mind, for any level of delay, the aim of using the DP policies is to optimally control movements from point 2, when the delay is realised, onwards. The aim of the SDP is to optimally control the speed of the airborne aircraft between points 1 and 2. The following sections explore the use of Value Iteration (VI) and Dynamic Programming (DP) to find solutions for optimally moving through this state-space. Specifically, Section 4.5 will outline the use of DP for solving the deterministic problem, while Section 4.7 will outline the use of a stochastic DP for solving the stochastic problem.

4.4.3 Deterministic and Stochastic Regions of the State Space

One key assumption made in this chapter is that uncertainty is isolated to takeoff times. Therefore the state-space approach outlined can be thought to be comprised of two main regions. The boundaries of the state-space, *i.e.* when $x_1 = 0$ or $x_2 = 0$, are

the areas where one of the aircraft has yet to take off, so uncertainty exists and will be require stochastic methods to solve. Conversely, once both aircraft are in the air there is no longer any uncertainty and the problem is deterministic. The interior *i.e.* when $x_1, x_2 > 0$, defines the deterministic region, with no uncertainty and so deterministic methods can be used. Therefore with this in mind the entire state-space can be split into two subsets:

$$S = \{(x_1, x_2) : x_1, x_2 > 0\}, \quad (4.5)$$

$$\hat{S} = \{(x_1, 0)\} \cup \{(0, x_2)\}, \forall x_1, x_2. \quad (4.6)$$

This results in two connected sub-regions and corresponding sub-problems. The deterministic region, S , will be solved in Section 4.5 using deterministic Dynamic Programming (DP). The stochastic region, \hat{S} , will be solved in Section 4.7 and will require the use of a Stochastic DP (SDP). It is necessary to solve for the deterministic region, S , first, as its solutions will act as an important cost function within the stochastic problem.

It is useful to note here that the use of a DP for the interior of this state-space is not entirely necessary and other, continuous methods can be used. However this work chooses to use a DP approach to develop a framework which in future could also include the effects of uncertainty en route (for example risks of adverse weather at certain locations).

4.5 The Deterministic Problem: Both Flights Airbourne

To solve the deterministic side of the problem, *i.e.* when both aircraft have taken off and there is no longer any uncertainty, we can use Dynamic Programming. The principle of Dynamic Programming (DP) is to reduce a complex problem into a sequence of smaller, simpler, subproblems, working backwards from an end goal to the starting point. That is, if it is known how to get optimally from a state $n - 1$ to the final state n , then from $n - 2$ you need only work out how to optimally get to $n - 1$ to know how to then get all the way to n . The starting point can then be reached by recursively working backwards through all the states to arrive at a final solution.

This chapter uses the method of Value Iteration, often used in reinforcement learning [97], to solve the DP problem. Starting from the final state and working backwards, at each new state the ‘cost-to-go’ to all reachable states is calculated.

The best course of action, for example choosing the next state based on a minimal cost, is then decided. By working backwards (backwards induction) each new state will be assigned a best cost and best action to take based on the previously calculated costs.

4.5.1 Problem Formulation

For the deterministic formation flight problem the state s defines how far along each aircraft is along its path, the entire state-space

$$S \subseteq (0, x_{1max}] \times (0, x_{2max}], \quad (4.7)$$

consists of the finite set of all these possible discrete states s . The control $u \in U$ is then the decision of what speed the aircraft should fly until the next state $s' = u(s)$. The finite set of all applicable controls

$$U = [V_{1min}, V_{1max}] \times [V_{2min}, V_{2max}], \quad (4.8)$$

is assigned a reachability function

$$U_r(s) : S \rightarrow S,$$

where $U_r(s)$ is the set of all controls which can be applied at the state s , that is, all the possible speeds which the aircraft can fly. The possible choices of control $u \in U_r$ are therefore defined by the reachability region R_r of Equation (4.4). The cost function

$$C : S \times U \rightarrow \mathbb{R}^+,$$

is the cost $C(s, u)$ of executing a given control u at a given state s . That is, the fuel the aircraft burn by applying $u(s)$ and moving from state s to s' .

Therefore, at each step the system is at a distinct state $s \in S$ and can follow out any applicable action $u \in U_r(s) \subseteq U$ for a cost $C(s, u)$. A policy, $\pi : S \rightarrow U$, is then defined as a mapping from the state-space S to the space of actions U , describing which action to take at each state s to get to the next state s' . That is, a policy defines the speeds the aircraft should fly for any given state.

Finally, for a given goal state G (*i.e.* when both aircraft have landed) and an initial state s_0 a solution is obtained by finding the policy $\pi \in \Pi$ which is optimal, denoted π^* . This optimal policy, will be the speed choice which minimises the total

fuel.

An example of a possible solution from a state $s \in S$ is shown in Figure 4.8 where the current state, s , is point 2. The deterministic problem only applies between the points 2 – 4 (*i.e.* when both aircraft have taken off). At point 2 flight F_1 and F_2 apply the control u to fly the best speeds to meet at 3. Then a formation is made and flown between 3 and 4. The path between point 1 and 2 is stochastic and will be covered in Section 4.7.

With that in mind, how well a policy π performs is based on the cost-to-go function J :

$$J_\pi(s) = C(s, \pi(s)) + \sum_{s' \in S} J_\pi(s'), \quad (4.9)$$

defined as the cost to reach the next state s' , by following the policy $\pi(s) = s'$ plus the (already calculated) cost to finish from s' . Finally, for any policy to be optimal it must satisfy the Bellman equations [96]:

$$J^*(s) = 0 \text{ if } s \in G, \text{ otherwise,} \\ J^*(s) = \min_{u \in U(s)} \left[C(s, u) + \sum_{s' \in S} J^*(s') \right]. \quad (4.10)$$

The corresponding optimal policy $\pi^*(s)$ at a state s can then be deduced from Equation (4.9) by choosing the sequence of controls u_1, u_2, \dots, u_k such that $J_\pi(s)$ is minimized. This optimal policy, therefore contains the set of speeds the aircraft should fly for all possible states to minimise the total cost.

4.5.2 Value Iteration

With this in place, Value Iteration is used to solve the Dynamic Programming problem. The concept (first introduced by Bellman [98]) involves an initialized value function (usually zero) which is then iteratively updated with the best currently-found value as the algorithm progresses. That is, at each state s encountered, the minimal value

$$J(s) \leftarrow \min_{u \in U_r(s)} [C(s, u) + \sum_{s' \in S} J(s')], \quad (4.11)$$

is assigned to the value function for that state to create successively better solutions. Therefore the Value Iteration algorithm starts at a final state and iterates backwards until a sufficient solution is found.

4.6 Sampling of the State Space

The discretization of the state space means there are a fixed and finite number of points needing evaluation. However, the dynamics of the state space are essentially continuous while Dynamic Programming is naturally discrete. Therefore it is essential to be able to use the efficiency of DP without being too negatively impacted by the need to use sample points. Computation time will roughly scale linearly with the number of sample points. Therefore the way the state space is decomposed into sample points will have a large impact on both the quality of solution and also the time required to solve.

4.6.1 Grid-based Sampling

The use of grid-based state spaces are frequently used within artificial intelligence and machine learning [99,100], often treated as a Markov Decision Process. Splitting the state space into a grid for the purposes of our problem will mean there are a finite number of discrete states $s \in S$ (the grid points) and therefore moves through the state space can only occur between these grid points. At each state (or node), s , the reachable region R_r , defined in Equation (4.4), restricts the possible control choices to a subset of S . Furthermore the reachable region is further restricted so as to only look ahead by some fixed radial distance, as in Figure 4.9(a). Therefore for each state there is manageable subset of possible states $S_R \subset S$ to feasibly move to. The corresponding cost, $C(s, s' = u(s))$, of reaching each of these states, $s' \in S_R$, by applying the control u , plus the cost-to-go, $J(s')$ from s' is calculated. The next state, s' , and the corresponding control $s' = u^*(s)$ which minimises the total cost is then chosen.

In a basic grid-based system, a movement can only occur between fixed nodes. Therefore the control choice is a discrete one and must be chosen such that by applying the control we arrive at another grid point. That is, given the current state $s = (x_1, x_2)$ and the state we wish to move to $s' = (x'_1, x'_2)$ the control choice of speeds V_1 and V_2 must be such that:

$$\frac{x'_1 - x_1}{x'_2 - x_2} = \frac{V_1}{V_2}. \quad (4.12)$$

Therefore, in the grid-based system, the control is as much a choice of next state, s' , as it is the choice of speed. As a result, the restriction of choosing a control from only a handful of choices can lead to very ‘choppy’ control policies and thus poor solutions. The region of reachable states from the current state is shown in Figure 4.9(a) and is

defined by some radial Manhattan distance, Δr as follows

$$0 < |(x'_1 - x_1) + (x'_2 - x_2)| \leq \Delta r, \quad (4.13)$$

with the added condition, due to the dynamics of the aircrafts' flight, that we can only move in a positive direction in x_1 and x_2 , that is $x'_1 > x_1$ and $x'_2 > x_2$.

In order to counteract this 'choppiness' a greater level of grid-resolution is required. The higher the resolution, the greater the number of control choices available and the smoother the resulting solution. However, a simple grid-based approach would, for a set resolution of n points for x_1 and m for x_2 , yield $n \times m$ sample points. Thus increasing the resolution by a factor of g_r in each direction would result in $g_r \times n \times g_r \times m = g_r^2 \times n \times m$ sample points, that is, it scales quadratically. Although increased resolution is important for good quality solutions there is clearly a compromise between evaluation time and resolution.

In order to mitigate some of these negative effects of using a discretized state space two modifications are made to the simple grid-based approach. Firstly linear interpolation is used to create a continuous control choice and associated cost. Secondly a quadtree decomposition is used to efficiently allocate resolution to areas of greater non-linearity.

4.6.2 Interpolated Cost Function

To overcome the shortcomings of a discrete state space a more continuous method of linear interpolation can instead be used [101, 102]. In a grid-based system every grid square has four nodes, a Bottom Left (BL), a Bottom Right (BR), a Top Right (TR) and a Top Left (TL). Since x_1 and x_2 can only move in a positive direction, we take the BL to be our sample point s , and look to move in some direction across the square (our control u). The TL and BR points will already have a cost-to-go, $J(s_{TL})$ and $J(s_{BR})$, associated with them so via linear interpolation they can be used to estimate a potential cost-to-go from any point on the diagonal line connecting them. As shown in Figures 4.9(a) and 4.9(b), the reachable region overlaps the horizontal line between the TL and BR and so can be used for interpolation. Figure 4.10 shows that any point reached between TL and BR, by applying the control $u(s)$, has a estimated cost-to-go value $J(u(s))$. Therefore the total cost is the cost to reach the line plus the interpolated cost-to-go $J(s(u))$ from that point.

With discrete choices the optimal choice is just the minimum of the finite set of costs of possible choices. With a continuous choice there are an infinite amount of

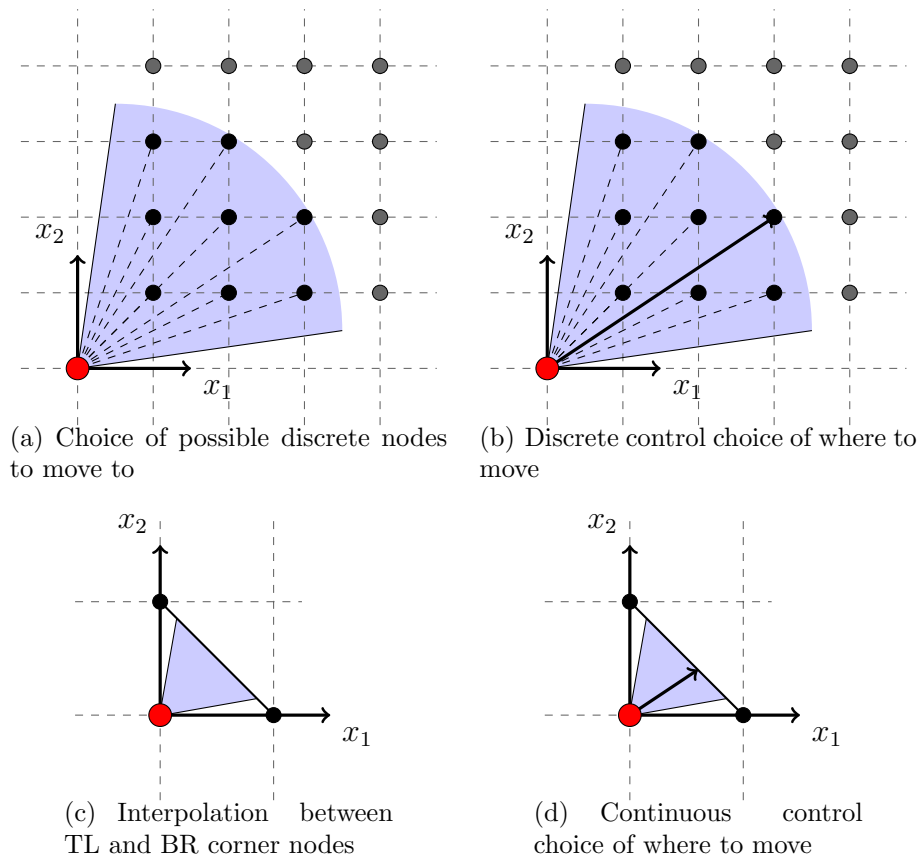


Figure 4.9: Discretisation of State Space for grid-based method

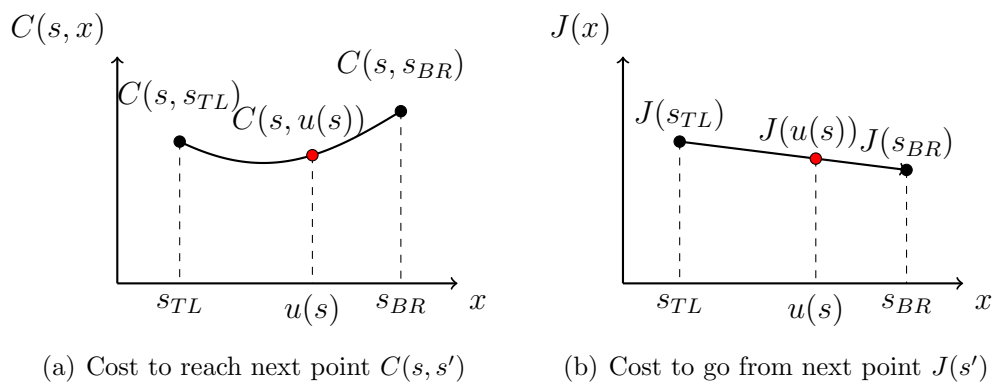


Figure 4.10: Interpolation of cost-to-go J

choices. Therefore a one-dimensional interval search is used to find, to a sufficient tolerance level, the optimal control choice over the interval of possible choices.

The basic grid-based method can still be used, but instead of choosing a discrete node to move to, the choice becomes a continuous one of which exact direction to move in and thus the speeds to fly at. This results in a much smoother and more realistic control choice.

The last major issue of using a linear interpolation for J is that it assumes that J can be locally approximated linearly to a sufficient level of accuracy. The finer the resolution of the grid sample points the better the linear approximation will be. However, the quadratic scaling of sample points to resolution means that we cannot just turn up the resolution without being heavily penalised computationally. Therefore a ‘smarter’ approach must be used to successfully incorporate linear interpolation into the problem.

4.6.3 Quadtree decomposition

In order to more efficiently allocate the placement of sample points a quadtree composition is used. Quadtrees are a type of hierarchical data structure, often used in image processing and recognition, and is based on a principle of recursive decomposition [103]. A region is recursively subdivided into four quadrants whilst some condition is met. The subdivision is usually specified to occur at areas of interest or where data points lie. The Quadtree decomposition in this scenario is used to subdivide the state space S into a series of rectangles. Each rectangle is comprised of four corners, and each of these corners represents a sample point to evaluate.

As linear interpolation is used, there is an assumption that the cost can be estimated linearly using sufficient sample points. Therefore for the purposes of this problem the subdivision will occur over the regions that a greater resolution is required, that is, the regions which are more non-linear. In particular the regions around the formation state S_F and those near the origin will be the most non-linear and therefore require the most decomposition. An example of this decomposition can be seen in Figure 4.11, where the diagonal blue line represents the formation state, S_F . In this case the requiring condition for subdivision is if the region is sufficiently close to any of S_F (up to a subdivision depth limit of 4). As S is recursively subdivided it results in greater resolution around the formation line. Comparing the quadtree decomposition against a grid-based method (Figure 4.12) shows that the number of nodes needing evaluating is more than halved whilst maintaining the same resolution

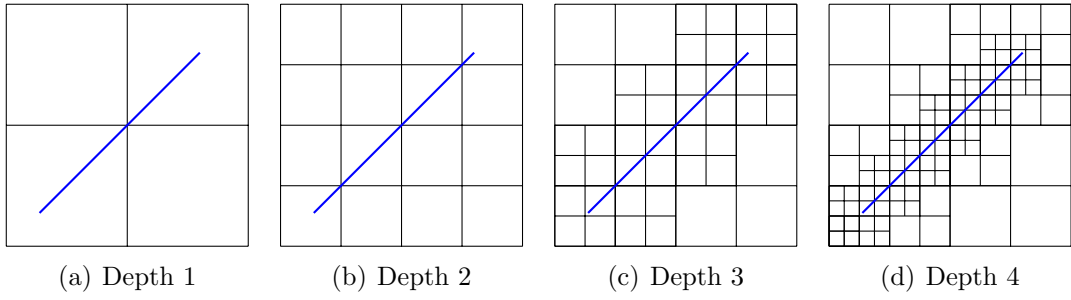


Figure 4.11: Quadtree based sampling method

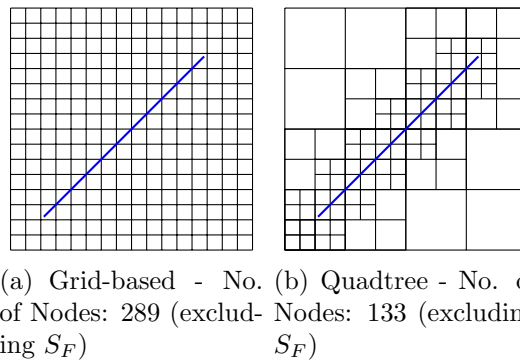


Figure 4.12: Node count grid-based vs. quadtree

around S_F identical.

Therefore, with a quadtree decomposition, by adding sufficient sample points around non-linearities, it is possible to reasonably linearly approximate the cost-to-go function J without being computationally prohibited.

4.6.4 Interpolation Between Quadtree Sampling

Combining linear interpolation with the quadtree sampling is mostly straightforward, following most of the same rules as on a grid-based approach. Every sample point s must be evaluated and a cost-to-go $J(s)$ must be assigned. For each sample point there are four possible evaluation types as follows:

- Regular node: The sample point corresponds to the BL of a quadrant. It uses the corresponding TL and BR nodes for interpolation. (As if Figure 4.13(a)).
- Hanging node 1: The sample point does not correspond to the BL of a quadrant, due to a change in resolution. It must use the TL and BR of the corresponding larger quadrant for interpolation. (As in Figure 4.13(b)).

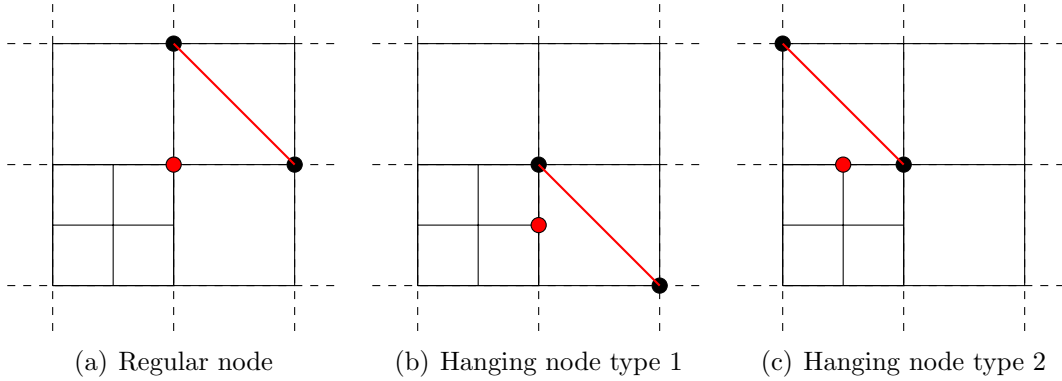


Figure 4.13: Types of quadtree nodes needing evaluation and the corresponding interpolation points

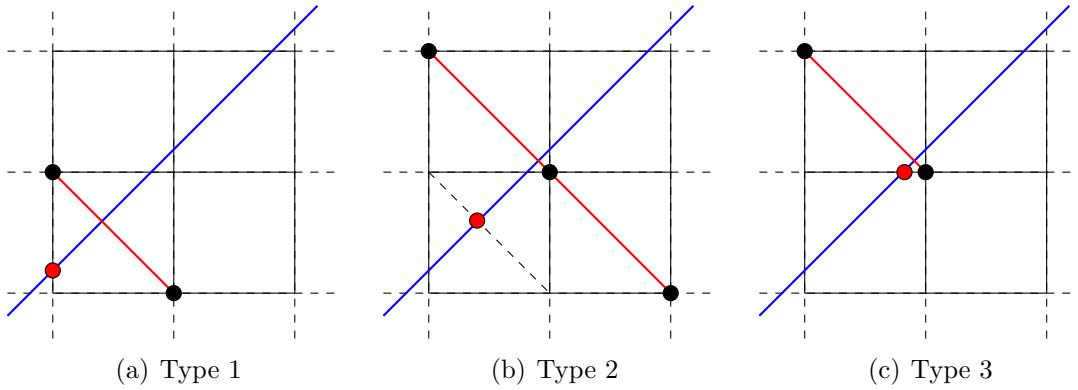


Figure 4.14: Types of quadtree nodes needing evaluation and the corresponding interpolation points between formation points and non-formation points

- Hanging node 2: The sample point does not correspond to the BL of a quadrant, due to a change in resolution. It must use the TL and BR of the corresponding larger quadrant for interpolation. (As in Figure 4.13(c)).
- Outer edge node: The sample point corresponds to an outside edge of S . F_1 (F_2) has landed so flight F_2 (F_1) moves vertically (horizontally) along x_2 (x_1).

Sample points along the formation state S_F are handled slightly differently. Formation sample points are placed at the following locations:

- Where any of the quadtree rectangle edges intersect the formation line S_F (as in Figures 4.13(a) and 4.13(c));
- Where any of the diagonal lines between the TL and BR of the quadtree rectangles intersect the formation line S_F (as in Figure 4.13(b)).

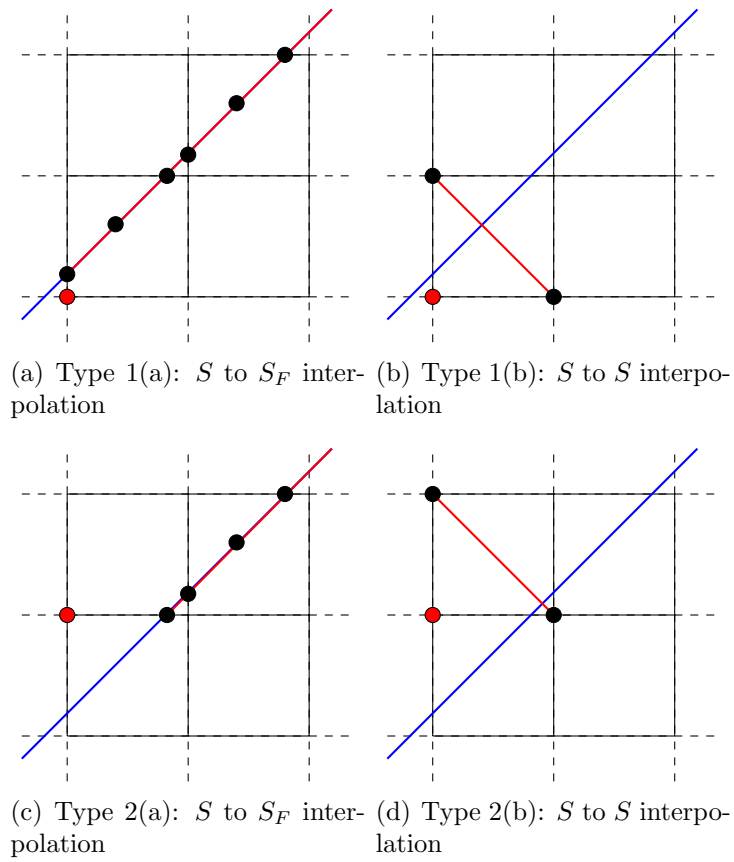


Figure 4.15: Interpolating when it is possible to move from S to S_F . The sample points used (black dots) and the corresponding interpolation line (Red line) used.

The three types of formation sample point and their corresponding line of interpolation are depicted in Figure 4.14. There is no real need for interpolation moving between points on S_F you simply choose the next sample point along S_F and calculate the corresponding cost. However, there is the option to move from S_F to S and in doing so leave formation. To calculate the cost to move from S_F to S linear interpolation can still be used.

When using a sample point on S which can feasibly reach S_F , then there is also the option to move from S to S_F and join in formation. Therefore it is necessary to calculate the costs of the two scenarios, moving to S_F or remaining on S , and choosing the best one to follow. That is, firstly interpolate along S_F (as in Figures 4.15(a) and 4.15(c)) and choose where the best location would be to move to on S_F , ignoring S , that would minimise the cost to reach it plus the cost-to-go (note that this interpolation could span much farther along S_F than current grid square). Then compare that cost to the cost of remaining on S , interpolating across the diagonal between TL and BR as usual (as in Figures 4.15(b) and 4.15(d)).

This covers the majority of scenarios for incorporating linear interpolation and a quadtree decomposition for use within this problem.

4.6.5 Climbing and Descending

There are regions of S whereby a policy assignment must be assessed slightly differently. First assume that each aircraft must climb and descend at a set rate over a set distance, with these values remaining consistent with estimates from Chapter 2. When both aircraft are at cruise then the speed-control choice is over the two variable speeds V_1 and V_2 , both of which can be adjusted in order to minimise the total fuel cost. If one of the aircraft is climbing then their speed $V_{i\text{climb}}$ ($i \in \{1, 2\}$) is fixed and therefore only the other aircraft can alter its speed. If both are climbing, then there is no control and they simply follow the speed defined by $V_{1\text{climb}}$ and $V_{2\text{climb}}$. The same also applies when one or both aircraft are descending, with their descent speed denoted by $V_{1\text{desc}}$ and $V_{2\text{desc}}$.

Therefore it is necessary to adjust S slightly. Instead, the quadtree decomposition is applied only over the region of S corresponding to both flight F_1 and F_2 are in cruise. The remaining climbing/descending regions are then simply divided using the outermost sample points of the inner cruise section as shown in Figure 4.16.

With all the elements of the deterministic problem in place, the region S can be solved via Dynamic Programming and Value Iteration outlined in Section 4.5. This

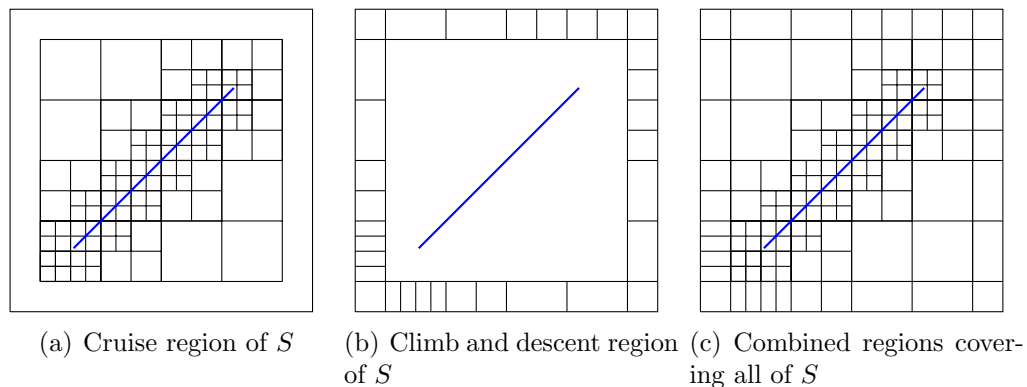


Figure 4.16: Combining the two subregions of S to account for climbing, cruising and descending

essentially solves the deterministic problem of deciding optimal speed-control policies for two aircraft to fly for any position within the state space. Therefore, given the current position of the two aircraft there is a pre-defined optimal policy of speeds for each of the aircraft to fly for the rest of the flight.

In the following section uncertainty is introduced into the problem. The speed-policies and corresponding values calculated for the cost-to-go function J for any point on S are fundamental. The deterministic solution provides a basis for calculating costs within the stochastic stage of the problem.

4.7 Stochastic Dynamic Programming

When uncertainty is introduced into the problem Stochastic Dynamic Programming (SDP) is needed. Without certainty about the next state, at each stage of the flight there needs to be a best action to take given any possible realization of the uncertainty. Therefore one does not know the absolute best solution, rather a set of policies to follow (based on the best expected outcome). Having costs which are sensitive to probabilistic events can be ‘risky’ and therefore the underlying solutions can also be sensitive. This methodology is key to trying to put a ‘cost’ on a possible solution with the intention of eventually making the solutions robust.

The SDP is very similar to the DP problem of Section 4.5. The main difference is that the cost-to-go at each state is minimized over a probabilistic expected cost and so cannot be guaranteed. The stochasticity of the problem is modelled using the same probability density functions as the Hold approach, first outlined in Section 4.2.

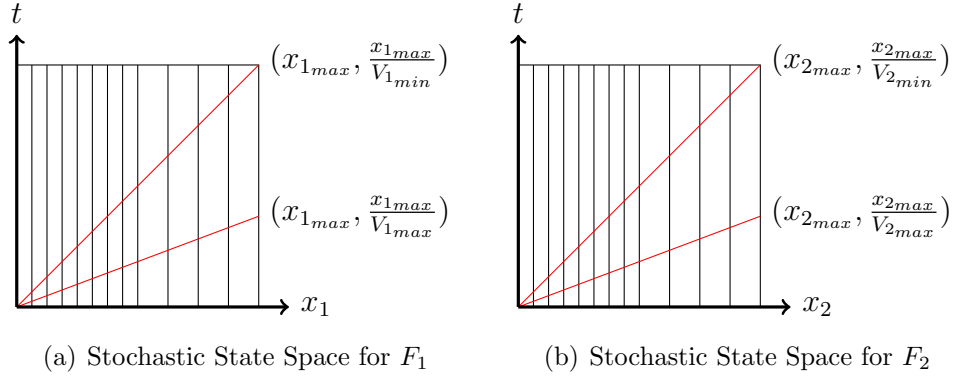


Figure 4.17: Stochastic State Space for the airborne aircraft

4.7.1 Problem Formulation

The stochastic region of the problem, \hat{S} , is defined by two dimensions, a spatial distance x_i (for $i \in \{1, 2\}$) and a time, t . Therefore the axes of S , *i.e.* when $x_1 = 0$ and/or $x_2 = 0$ comprise the spatial dimension. The time dimension is limited by the total flight duration, that is, for each aircraft $i \in \{1, 2\}$, t is bounded by

$$T_{i_{min}} = \frac{x_{i_{max}}}{V_{i_{max}}}$$

$$T_{i_{max}} = \frac{x_{i_{max}}}{V_{i_{min}}}$$

Therefore for each aircraft i the stochastic state-space is defined over t as:

$$\hat{S}_i \subseteq \cup_{i \in \{1, 2\}} \{x_i \times [T_{i_{min}}, T_{i_{max}}]\}. \quad (4.14)$$

A state $s = (x, t) \in \hat{S}$ consists of both a position x along a particular axis and the time t it is at that location, shown in Figure 4.17. The control $u(s)$ is the choice of speed, V , for the aircraft to fly to reach the next state. As each of the axes x_i are discretized into N_i spatial states $\{0, x_{i_1}, x_{i_2}, \dots, x_{i_{N_i}}\}$, spatially we need only look ahead by one step. Therefore for the current state $s = (x_{i_k}, t)$ a speed-control choice of $u(s) = V$ would result in the next state being

$$s' = (x_{i_{k+1}}, t + \frac{x_{i_{k+1}} - x_{i_k}}{V}).$$

Furthermore at each spatial state x_{i_k} there is a feasible time range at which the aircraft could be in. This temporal region of reachability, R_r , at time t is defined by

the minimum and maximum velocities of the aircraft, that is

$$R_r = t' : t' \in \left[t + \frac{\Delta x_{i_k}}{V_{i_{max}}}, t + \frac{\Delta x_{i_k}}{V_{i_{min}}} \right].$$

The stochastic state-space \hat{S} is then augmented with a binary state $\delta \in \{0, 1\}$ which defines whether or not the other aircraft has taken off yet. Therefore a solution begins in \hat{S} and as soon as the other aircraft takes off the solution transitions to the deterministic solution and the DP policies on S are used. Therefore at each state $s = (x_{i_k}, t, \delta) \in \hat{S}$ a decision $u(s)$ is made of how fast to fly and essentially how many time-steps t to take to reach the next discrete way point $x_{i_{k+1}}$, if the other aircraft takes-off ($\delta = 1$) the solution transitions to S and the DP solution is used. Finally given a finishing goal location G and initial state s_0 a solution is obtained by finding policies $\pi \in \Pi$ which are optimal.

In the stochastic problem, the performance of any policy π is determined by its expected cost, governed by a given probability distribution f . Given the current state $s = (x, t, \delta)$ and the control applied gives the next state $u(s) = s' = (x', t', \delta')$. With n_r and p corresponding to the values of Table 4.1, let us define the probability functional \mathcal{F} , that the other aircraft takes off at s' , as

$$\mathcal{F}(s, s') = \mathbb{P}(s', \delta) = \begin{cases} 1 & \text{if } \delta = 1, \\ f(t, t' | n_r, p) & \text{otherwise.} \end{cases} \quad (4.15)$$

With this in place the expected value function is defined as,

$$\mathbb{E}(s, s') = \sum_{s' \in S} [\mathcal{F}(s, s') J_\pi(x', t', 1) + (1 - \mathcal{F}(s, s')) J_\pi(x', t', 0)]. \quad (4.16)$$

where the corresponding cost-to-go function is,

$$J_\pi(s) = C(s, \pi(s)) + \mathbb{E}(s, s'). \quad (4.17)$$

The final Bellman equations are therefore

$$\begin{aligned} J^*(s) &= 0 \text{ if } s \in G, \text{ otherwise,} \\ J^*(s) &= \min_{u \in U(s)} [\mathbb{E}(s, s')]. \end{aligned} \quad (4.18)$$

Then for a given formation pair a final solution would consist of two parts. An optimal expected cost-to-go $J^*(s_0)$, from the initial state s_0 , and the corresponding optimal

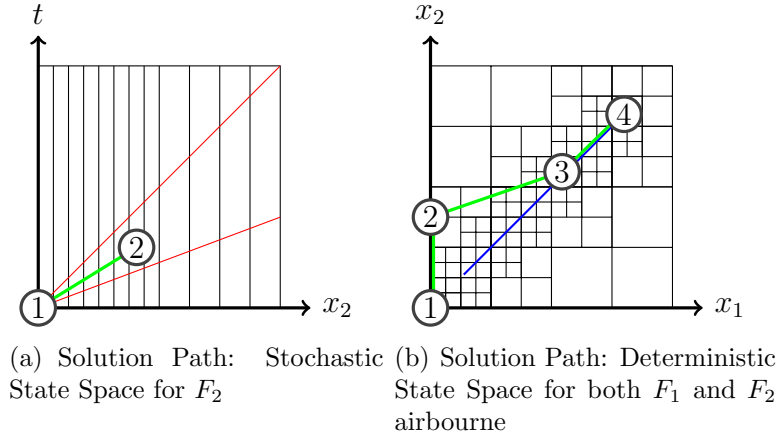


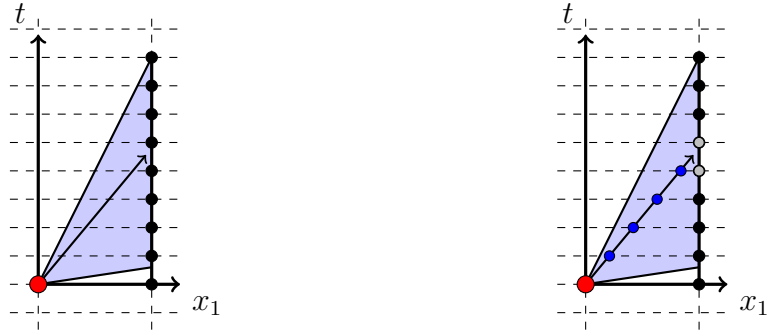
Figure 4.18: Solution Path for F_2 : Between 1 and 2 F_2 flies the SDP policy as F_1 is delayed. Between 2 and 3 both aircraft fly the DP policy, to meet at 3 and fly in formation until 4 where the break and fly solo.

policy π^* to follow until the total delay is realized.

Finally, the example of Figure 4.8 is plotted again in Figure 4.18, showing the relationship between the different state spaces, and outlines a possible solution path. When only F_2 is airborne the stochastic problem is responsible for assigning a policy between points 1 – 2. As soon as the other aircraft has taken off (point 2) and both are airborne the problem is deterministic and the solution path from the DP can then be used between 2 – 4. That is, the solution path jumps from the stochastic state space, \hat{S} , to the deterministic one, S , once the delay has been realised.

4.7.2 Sampling of the Stochastic State Space

The stochastic state space \hat{S} is comprised of two dimensions, a spatial dimension, either x_1 or x_2 , and a temporal dimension, t . Any movement through \hat{S} can only be in positive direction and with the gradient representing the aircraft speed ($V = \frac{\Delta x}{\Delta t}$). The entirety of this state space therefore cannot be reached (nor does it need to be) due to minimum and maximum aircraft velocities. Due to this, for a simple grid-based decomposition, there is already a drastic reduction in the number of nodes needing evaluation. Therefore a grid-based resolution is sufficient for \hat{S} . The temporal resolution for \hat{S} is simply chosen to be a sensible time step t_{step} (as the resolution of the PDFs is 1 minute we use that as the time step). The spatial resolution is chosen so that it matches up with the resolution of the quadtree of S . This effectively results in a binary tree decomposition (which is the one-dimensional analogue of a quadtree of Section 4.6.3), to again, add resolution to those areas with a greater level



(a) Reachable sample points of next spatial state and control choice to reach

(b) Corresponding expected cost-to-reach calculation: a cumulative calculation over each time-step

Figure 4.19: Grid-based interpolation for stochastic State Space

of non-linearity.

The methods for the stochastic state-space are similar to those methods already outlined in Section 4.5 but without needing to use quadtree sampling. The interpolation discussed in Section 4.6.2 will still be used, but the interpolation will only occur over the variable of time, t , at the fixed discretised values of x_i as shown in Figure 4.19. In that sense, with a known cost-to-go value from each sample point (the black dots of Figure 4.19(a)), the cost-to-go function will be a linear interpolation over those sample points and their values. The control choice, of what speed to fly, is then a continuous one.

The final modification is to the cost-to-reach function $C(s, s', \delta)$ to go between states s and s' , as this value is also uncertain. Recall the PDF distributions of possible delays outlined in Section 4.2, whereby an aircraft on the ground is assumed to take off only at the discrete time steps t , with each time step spacing of one minute. Therefore as the airbourne aircraft flies between two states, at the control-chosen speed $V = u(s)$, in km per time step (*i.e.* km per minute), there is the possibility that the other aircraft will take off. The cost-to-reach the next state for a given control choice, is thus the probabilistic expected cost over all possible outcomes.

That is, for each time-step $j_t \in \{1, 2, \dots, j_{t\max}\}$, between s and s' there is a corresponding spatial location $x_{i_k} + V \times j$ creating a set of possible locations,

$$\cup_{j_t \in \{1, \dots, j_{t\max}\}} \{x_{i_k} + V \times j, t + j\}$$

outwith the sample points where the other aircraft could take off, as show by the blue

dots in Figure 4.19(b). For each of these points there are two possibilities, either the other aircraft takes off at that point, or it does not. The cost associated with each possible event is then proportionally weighted by its probability of occurring, with the sum of these creating the expected value. Additionally, the maximum time-step value is defined as $j_{t\max} = \lfloor \frac{(x_{i_{k+1}} - x_{i_k})}{V} \rfloor$, meaning there is a remaining small part to take into account between the last integer time-step and the time we reach the next spatial grid point $x_{i_{k+1}}$. As it is impossible for the other aircraft to take off at the intercept, the cost is simply the interpolation between the two, already calculated, cost-to-go from the grid points at $(x_{i_{k+1}}, t + j_{t\max})$ and $(x_{i_{k+1}}, t + j_{t\max} + 1)$, as depicted by the grey dots of Figure 4.19(b).

4.7.3 Combining the Deterministic and Stochastic State Spaces

As outlined in Sections 4.5 and 4.7 there are two distinct but linked subproblems corresponding to the two state spaces S and \hat{S} . The deterministic problem is solved first, giving a cost-to-go function J of all possible points $s \in S$. Most importantly, however, is the calculation of the cost-to-go values along the states for which one of x_1 or x_2 is zero (*i.e.* the axes of S). This gives the cost-to-go values at the point the other aircraft takes off, and allows us to calculate an expected cost-to-go function for the stochastic problem.

There are three subproblems in total, one S for both flights and then an \hat{S} for each of the two flights F_1 and F_2 . The order of solving is as follows:

- S for both flights: The deterministic problem solved via DP.
- \hat{S} for F_1 : When F_2 has yet to take off. The values found for S when $x_2 = 0$ are then used within the SDP.
- \hat{S} for F_2 : When F_1 has yet to take off. The values found for S when $x_1 = 0$ are then used within the SDP.

Figure 4.20 shows how the three state spaces interact and share common states variables. A solution path starts at the origin $(0, 0, 0)$, as the first flight F_i (for an $i \in \{1, 2\}$) takes off it moves along through \hat{S} for F_i , then as soon as the other flight takes off, the solution moves onto S and continues until it reaches the end of S , as in Figure 4.18.

The cost-to-go function J along the axes of S is used within the stochastic problem. As S is sampled and a linear interpolation is used then so too will the function used within \hat{S} . As explored in Section 4.7.2 and shown in Figure 4.20, the resolution

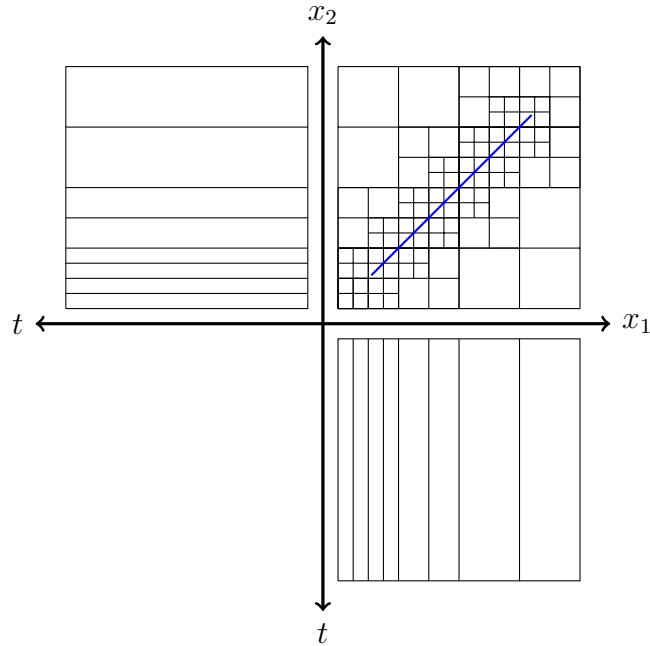


Figure 4.20: The shared variables between S and the two \hat{S} state spaces

along the axes extends into \hat{S} and is determined by the quadtree decomposition of S . Therefore it is important to keep a quadtree subdivision condition for S included for areas around the origin and initial takeoff. This allows for both a better approximation and also a suitable spatial subdivision of \hat{S} .

4.8 Example Formation

In this section the results of an example formation pairing are explored. The two flights used are:

Flight 1: Atlanta International Airport (ATL) to Manchester Airport (MAN)

Flight 2: Washington Dulles International Airport (IAD) to Heathrow Airport (LHR)

These flights represent a typical potential formation pairing within the Transatlantic flight list. This formation route is first calculated using the methodology of Chapter 2 and results in a very reasonable saving of 8.1% against flying solo. Although the two departure airports are located about 850 km apart, Flight 1's solo great circle path passes within 20 km of Flight 2's departure airport IAD. The total additional distance covered by the two flights when in formation is fairly low at 61 km, so it is clear there are minimal deviations resulting in a very good formation flight potential.

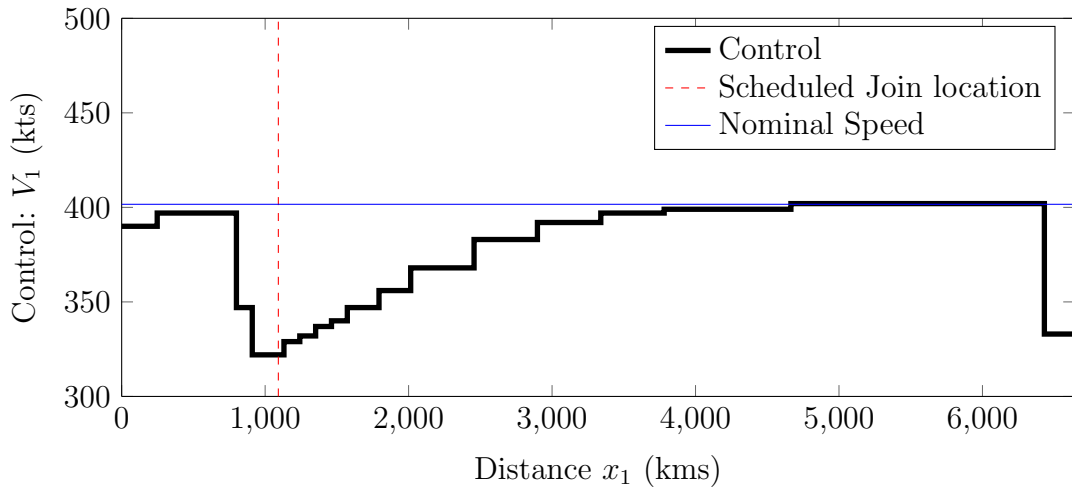
Using the methods outlined in this chapter the Deterministic and Stochastic Dynamic Programming problems are solved for this formation pairing. A ‘solution’ to this problem is considered to be a set of optimal policies, and their associated costs, to follow for any possible realisation of ground delay. Unlike the deterministic formation routing problem of Chapter 2 where the solution is a single number (the percentage saving), there are instead a number of different realisable outcomes and associated costs.

Figures 4.21(a) and 4.21(b) show the stochastic control policy π^* for what will be referred to as the No Takeoff Policy’ (NTP). The NTP is the control policy (in this case the speed to fly) the flight follows if the other flight never takes-off (infinitely delayed). It would outline a path through \hat{S} , then if the other flight were to takeoff the policy would switch from this, to the one calculated for the deterministic problem π^* . Note here that the first and last ‘step’ of each NTP are the predefined V_{climb} and V_{descend} (respectively) and of which we have no control.

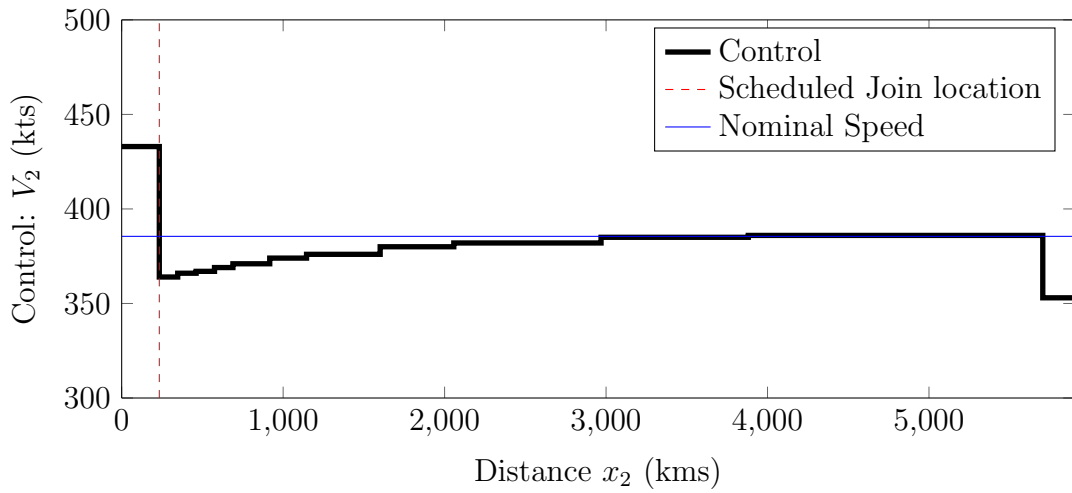
Flight 1 is scheduled to take off first (about 1 hour before Flight 2) and so the two NTPs are quite distinct, so in this scenario the focus lies on the NTP for Flight 1. The red vertical line indicates how far along the path each flight expects to meet in formation. Flight 1, as it takes off some time before, is just over 1000 km into its journey, while Flight 2 expects to meet just after it finishes climbing. The Flight 2 NTP can therefore only be reactive, slowing down slightly, in case Flight 1 is only a little late. It steadily converges back to its nominal speed as Flight 1 becomes too late for any possible saving. The NTP for Flight 1 on the other hand has the opportunity to be proactive, initially slowing down before the anticipated takeoff in case Flight 2 takes off early, then remains slow in an attempt to ‘wait’ for Flight 2. Again this control slowly converges back to the nominal cruise speed as the expected saving decreases.

The solution to this problem consists of a set of optimal policies for any possible delay, then each of these policies has an associated cost. That is, the total cost of following a particular policy, due to a particular delay having been realised. Clearly if either of the flights never takes off (and a NTP is followed) then the resulting realised cost will be the worst it can possibly be for that flight. Instead of a saving, as no formation is made and speed changes have likely been made the formation would result in a loss. Then the set of all possible outcomes and the associated costs define a set of realisable savings.

Figure 4.22 shows the probability distribution of achieving particular percentage savings (at intervals of 0.5%) for this particular formation. The expected saving,



(a) Flight 1 ATL-MAN



(b) Flight 2 IAD-LHR

Figure 4.21: No-Takeoff Policy for each flight

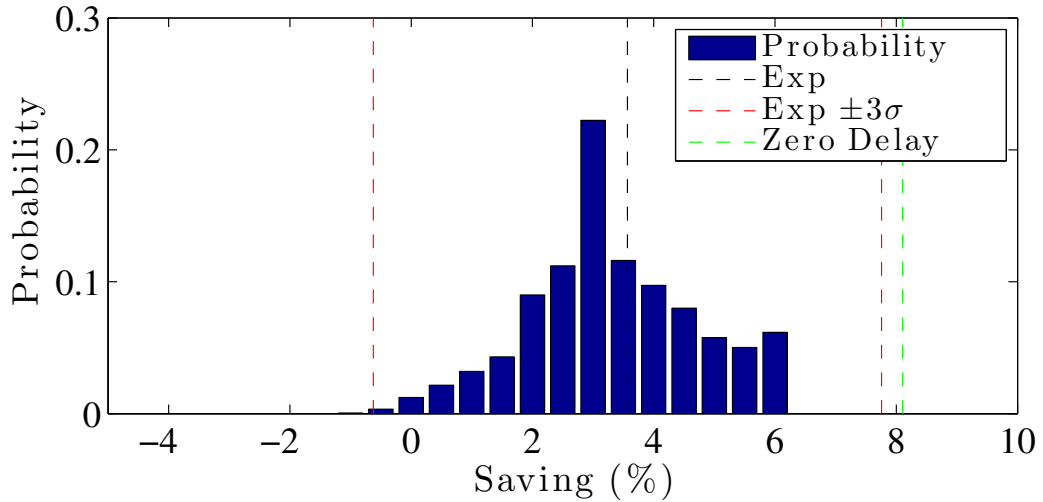


Figure 4.22: Probability distribution of possible savings for ATL-MAN and IAD-LHR

i.e. the sum of every possible outcome multiplied by the probability of it occurring, is 3.6%. This is just under half of the original 8.1% saving that was possible in a delay-free scenario, but in fact represents a very reasonable saving given delay. If you were to simulate this formation’s policies n times, then the average saving would converge to the expected value as n tends to infinity. However as achievable results are spread out no saving can be guaranteed and therefore a notion of risk and robustness must be introduced to help quantify what constitutes a ‘good’ solution.

4.9 Robust Planning

In the presence of uncertainty an issue arises in that the cost function can take a number of different values as a direct response to a specific realisation of uncertainty [89]. That is, an expected value is only a theoretical average, in fact, it is more likely that the formation achieves a different saving all together. For example if we had a 50% chance of a saving of -2% and a 50% chance of saving 4% our expected saving would be 1% , which on the surface seems reasonable, but in fact half of the time this formation would be actually be more expensive than just flying solo. What is most important, therefore, is the probabilistic ‘spread’ (also known as variability, scatter or dispersion) of the possible savings. That is, how far are we likely to deviate from the expected value.

The most common measure of spread is the standard deviation, σ , a lower σ would indicate the data points lie close to the expected value, while a higher σ would mean they are more spread out. For a normally distributed data set the percentage of

Table 4.2: The percentage of values expected to lie within Confidence Intervals $CI = (\bar{x} - \tau\sigma, \bar{x} + \tau\sigma)$

$\tau\sigma$	% within CI	% outside CI
1σ	68.27%	31.73%
2σ	95.45%	4.55%
3σ	99.73%	0.27%
4σ	99.99%	0.01%

values expected to lie within the Confidence Interval (CI) $(\bar{x} - \tau\sigma, \bar{x} + \tau\sigma)$ is shown in Table 4.2. This means that for a formation with an expected value \bar{x} and standard deviation σ , there is a 95% chance the formation will realise a solution within the interval $(\bar{x} - 2\sigma, \bar{x} + 2\sigma)$ and a 99% chance it lies within $(\bar{x} - 3\sigma, \bar{x} + 3\sigma)$. For the example of Section 4.8 the interval for $\pm 3\sigma$ is shown in Figure 4.22 and shows that there is only a small chance that this formation will result in a negative saving.

It is clear from the example of Section 4.8 that a single number, such as the expected value, cannot adequately describe how ‘good’ a formation is likely to be. Instead the level of variation from this expected value needs to also be considered, *i.e.* the standard deviation. Formations with high levels of variability (a high σ) have a greater level of risk associated with them, as their possible results are more spread out. It is important, when deciding on which formations to allocate, to have a notion of the risk associated with each one. In order to get an optimal yet robust allocation of formations one needs to be able to maximise the potential saving whilst simultaneously minimising this variability.

4.9.1 Robust Formation Assignment Using a MILP

Using the same MILP assignment problem of Equation (2.20) as outlined in Chapter 2 we look to minimise the total cost over the entire fleet of aircraft. The approach of this section will follow a similar idea to that outlined by Bertuccelli et al. [104], whereby a penalty, associated with the level of uncertainty, is imposed. In the basic MILP the costs c_j are chosen to be the estimated fuel burn for a formation x_j (although any metric can be used). For N_f total possible formations, the total cost is simply $\sum_{j=1}^{N_f} c_j x_j$, subject to each aircraft only belonging to one formation and x_j being binary.

In order to include this idea of robustness we choose to also optimise for uncertainty, making this a multi-objective optimization. To include this extra objective

into the original MILP we can simply adjust the cost by penalising the associated variability, *i.e.* the standard deviation, of each formation σ_j . That is, we add a penalty term $\tau\sigma$ into the cost function which is proportional, by a factor of τ , to the standard deviation. Then for a given penalty weighting τ on σ the assignment problem becomes

$$\begin{aligned}
& \underset{x}{\text{minimize}} && \sum_{j=1}^{N_f} (c_j + \tau\sigma_j)x_j, \\
& \text{subject to} && \sum_{j=1}^{N_f} p_{j,i}x_j = 1, \quad \forall i \in \{1, \dots, N_a\}, \\
& && x_j \text{ binary}, \quad \forall i \in \{1, \dots, N_a\}.
\end{aligned} \tag{4.19}$$

With this in place τ can be thought of as a ‘risk-tuning factor’ [104]. Increasing τ will put a greater penalty on each formation’s standard deviation, σ , and therefore begin to make those formations with a higher variability less desirable. This τ essentially corresponds to that of Table 4.2 and tuning this factor will simply cost the formations based on their confidence interval. More importantly though is that this τ reflects the desired level of risk aversion. A smaller τ will lead to greater expected savings but would be riskier than a larger τ .

Finally if we let c_{solo_i} denote the solo cost of a formation i , then for each formation i then we define the ‘risk level’ as

$$r_\sigma = 100 \times \sigma_i / c_{\text{solo}_i},$$

that is, the standard deviation as a percentage of the solo cost. A risk level of $r_\sigma = 2$ would correspond to the standard deviation being equivalent to 2% of the solo cost. Very roughly speaking, assuming a normal distribution, for an expected formation cost \bar{x} with the corresponding percentage saving denoted $ps_{\bar{x}}$, the 99% CI $(\bar{x} - 3\sigma, \bar{x} + 3\sigma)$ would then be equivalent to $(ps_{\bar{x}} - 3r_\sigma, ps_{\bar{x}} + 3r_\sigma)$. Thus, for a risk level of $r_\sigma = 2$ then with 99% confidence the solution will lie in $(ps_{\bar{x}} - 6, ps_{\bar{x}} + 6)$. Therefore for $\tau = 3$ (the 99% CI) the expected saving percentage $ps_{\bar{x}}$ would need to be at least 6% for this formation to be almost always favourable, anything less and the formation might often end up costing more than solo flight. Thus, the risk levels, τ values and the confidence intervals of Table 4.2 are all closely linked.

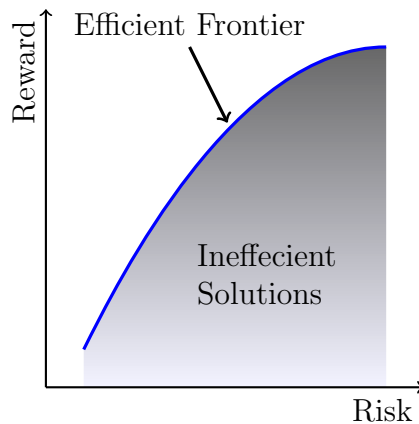


Figure 4.23: Efficient frontier solution

4.9.2 Portfolio Optimization

One of the main ideas behind modern portfolio theory is to choose a selection of investments, a portfolio, in an attempt to maximize the expected return for a given amount of risk. Therefore we consider each formation to be an investment with the return on this investment being the fuel saving. A portfolio would thus be an selection of these investments, that is, an allocation of formation flights. For a given level of risk, r_σ , the robust MILP of Section 4.9.1 can therefore be used to calculate the optimal portfolio. Altering one's level of risk aversion is then similar to tuning the τ penalty factor on the standard deviation.

Optimal portfolios can be calculated for any level of τ and thus any level of risk. Now since instead of just trying to minimize for cost we are also trying to minimize risk this leads to a multi-objective optimization. There is no intrinsic equivalence between a cost and a level of risk, instead there is a trade off between risk and reward. One could maximise reward whilst ignoring risk, minimize risk whilst ignoring reward or some combination of the two. Instead one looks for what is called 'pareto efficiency', whereby one objective cannot be improved without making the other worse off. In portfolio theory this is known as an 'efficient frontier' (Figure 4.23). Essentially, instead of a one dimensional solution (a point) there are two-dimensions (a curve) of points which are all equally 'efficient'. By choosing the value for one variable results in the optimal value for the other. Thus for a set level of risk, there is an associated return which is optimal and equivalently for a set level of return there is a level of risk which is optimal. Any point which does not lie on this frontier is known as 'inefficient', and can be improved by moving it onto the efficient frontier, either the same reward for less risk, a greater reward for the same risk or a combination of both.

A solution to the Robust assignment problem, therefore, is the calculation of the efficient frontier for the allocation of flights. With this, one can pick their level of risk aversion and simultaneously get their optimal expected return. Similarly if a set expected return is desired then there is then a corresponding minimum risk allocation.

4.10 Transatlantic Case Study

This case study continues with the set of Transatlantic flights of Chapter 2. Using the methodology of the two approaches outlined in this chapter we look to make robust assignments of formations. Each method has been implemented in Matlab and then run for all 21,945 possible formations.

4.10.1 Holding Pattern Approach

In order to calculate a solution for the holding pattern approach, for a given cutoff time t_{co} , it suffices to simply calculate Equation (4.2) for every possible relative delay. Every relative delay has a probability associated with it, while there is no inherent stochasticity in the cost function there is in the realisation of a delay. Therefore each solution is essentially a list of outcomes, its probability of occurring and the corresponding costs. For each formation for a given cutoff time t_{co} it takes roughly 2 seconds to calculate the entire solution (including the routing of Chapter 2). Therefore for all possible combinations it takes just over 12 hours of computation time to calculate all the necessary costs needed for use within the assignment stage.

Recall from Section 4.9.1 the introduction of the risk penalty τ for use within the MILP of Equation (4.19). For each formation c_i is the expected cost, σ_i is the standard deviation and x_i is the binary allocation of that formation to the solution. Using different values for τ , along with this MILP, can then be used to calculate the efficient frontier of Section 4.9.2.

As discussed in Section 4.3.1 an important variable within the hold approach is to specify a cutoff time t_{co} . Therefore it is important to show the results for a number of different t_{co} values. Allocating formations based on their expected values alone, *i.e.* when $\tau = 0$, for the cutoff times $t_{co} = 15, 60, 120$ and 240 minutes results in an overall expected savings of 2.8%, 4.7%, 5.3% and 5.5% respectively. By gradually increasing τ the ‘riskier’ formations are increasingly penalised resulting in less risky allocations. As shown in Figure 4.24, by increasing τ and removing the risky formations, results in a decreased overall expected saving. At relatively low

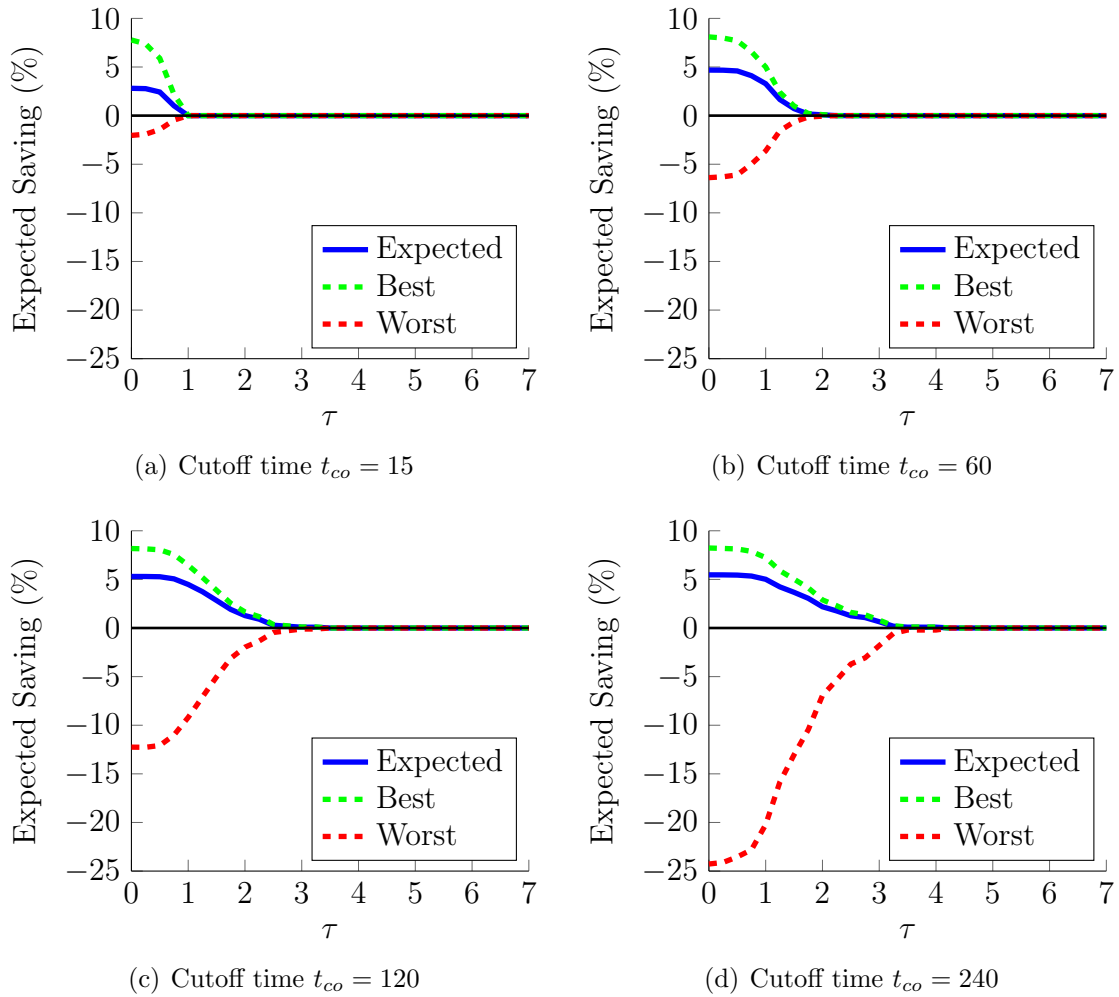


Figure 4.24: Expected values for tuning τ

levels of τ it becomes unfavourable to fly in formation when using this hold pattern approach.

What is clear from Figure 4.24 is that without a reasonable penalty for uncertainty and an increasing cutoff value t_{co} the allocations can have a large range of achievable values. At the absolute worst, for the unpenalised case, with a cutoff of 240 minutes the entire fleet could average a loss of almost 25%. However, by calculating the efficient frontiers, as shown in Figure 4.25, for the four t_{co} values of 15, 60, 120 and 240 minutes one can quickly compare the risk versus rewards of potential cutoff times. The efficient frontier described in Section 4.9.2 and shown in Figure 4.25 depicts the optimal formation allocation (or portfolio) for a given level of risk. Lower-risk portfolios are achieved by increasing τ , while higher-risk portfolios are a result of lower τ values. There may be formation allocations which lie below or to the right of

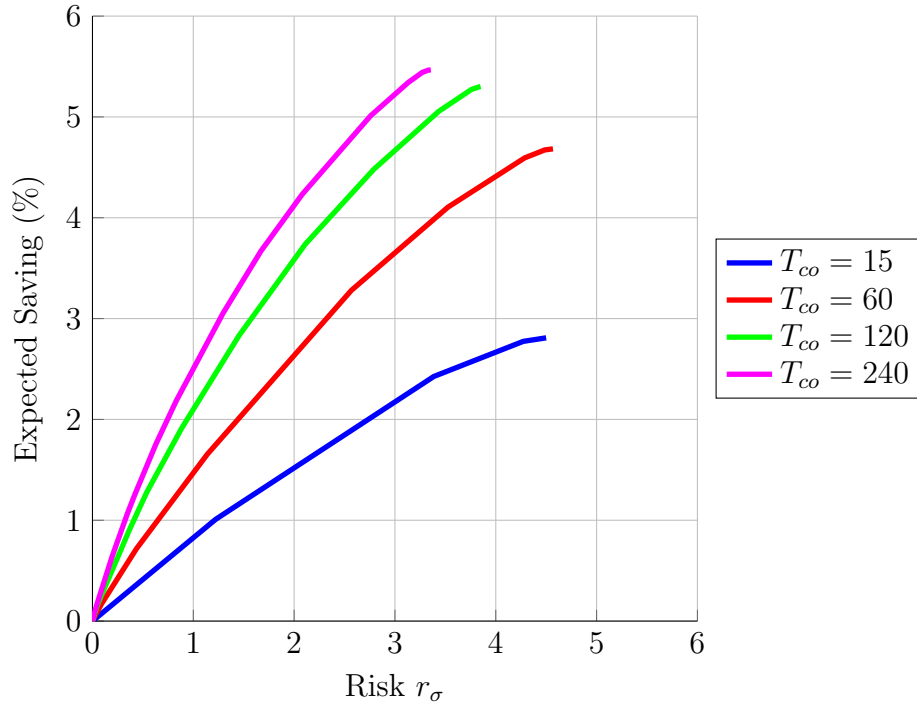


Figure 4.25: Efficient frontier of formation allocations

this efficient frontier, but they are inefficient and can therefore always be improved.

For a risk level of 1, one could expect to achieve anywhere from 0.8% for $t_{co} = 15$ to 2.5% for $t_{co} = 240$. Similarly for a greater risk level of 2, one could expect to achieve between 1.5% for $t_{co} = 15$ to 4.1% for $t_{co} = 240$. However, as discussed in Section 4.3.1, for a number of reasons high values of t_{co} would be very unlikely. Therefore t_{co} acts more as a theoretical maximum, rather than a likely achievable one. Cutoff levels up to around an hour may be possible and thus the corresponding line in Figure 4.25 acts as a reasonable guide.

4.10.2 State Space Approach

For each of the 21,945 possible formations the DP and SDP are run, with each one taking roughly 30 seconds. This means, to calculate all combinations around 180 hours of computation are required. The act of calculating all combinations, however, is itself embarrassingly parallelisable so can therefore be run on a number of cores or a computer cluster to reduce the realisable time to the user (*e.g.* a quad-core machine would take roughly $180/4 = 45$ hours).

For each combination, there is a list of possible outcomes, the probability of them occurring and the corresponding costs. From this the expected values and standard

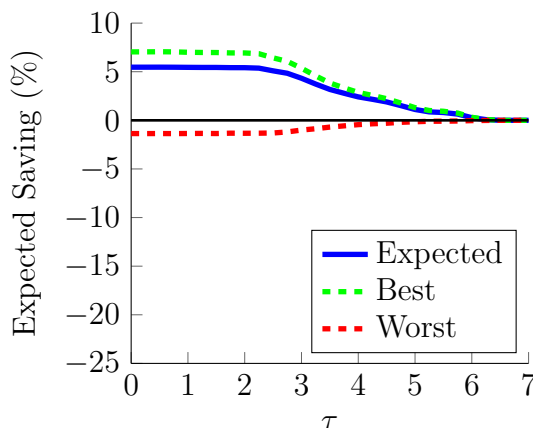


Figure 4.26: Tuning penalty and the corresponding expected savings

deviations are calculated for use within the allocation problem.

Again the idea is to tune τ to penalise the risky formations and achieve more robust solutions. For the unpenalised case, $\tau = 0$, the allocation results in an overall expected saving of about 5.5%. As the value for τ is gradually increased the average expected saving begins to decrease as shown in Figure 4.26. As τ reaches roughly 6 the penalty on uncertainty is so high that it is no longer favourable to fly in formation.

The efficient frontier of Figure 4.27 shows the tradeoff between a given level of risk-aversion and the expected reward. A maximum risk level of just under 1.5 leads to the unpenalised (maximum) expected saving of 5.5% while a level of 1 would yield about 4%.

4.10.3 Comparison of Methods

The hold approach provides a reasonably realistic and straight forward approach to mitigating the effect ground delay has on formation flight. However, the large jump between the cost function of Equation (4.2) for varying cutoffs values t_{co} results in a greater spread of possible results. The large proportion of good results partly outweighs the small proportion of extremely bad ones. However, those bad results could lay anywhere between -5% and -25% yielding a greater standard deviation σ and thus riskier results. The State Space approach provides results which are closer together. The cost of following a policy outlined by the State Space approach is a small reduction in the ‘best-case’ outcome. The ‘worst-case’ is also relatively contained, at most causing a 1% loss.

What is most importantly clear is that the State Space approach results in an efficient frontier far superior to that of the Hold approach. The same range of expected

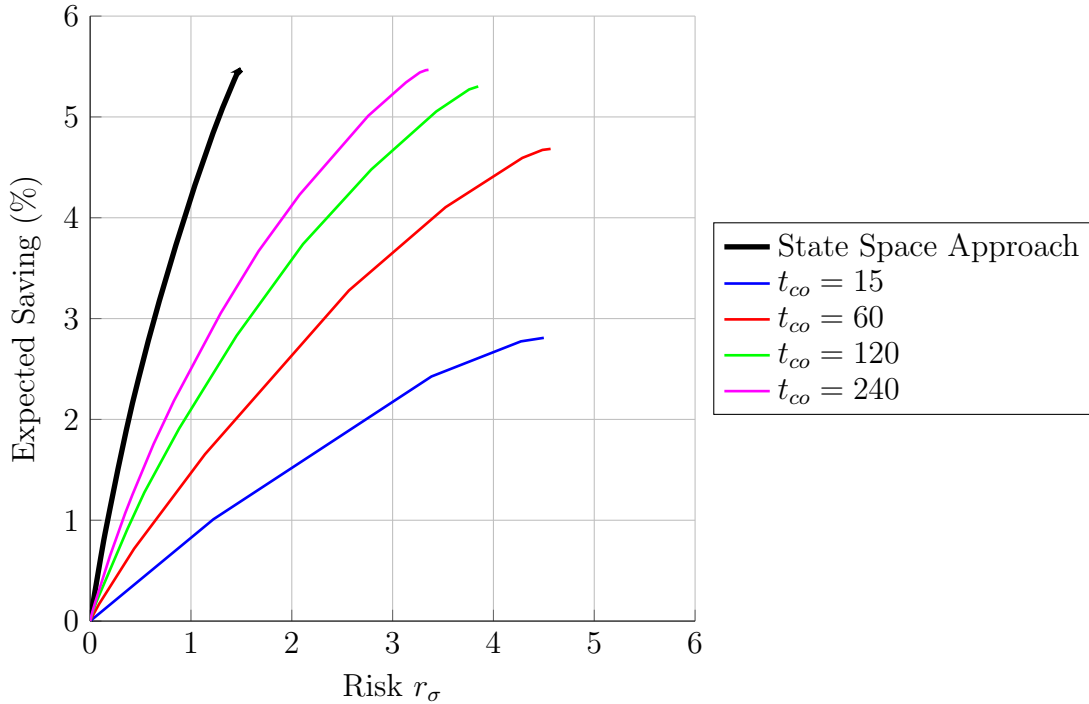


Figure 4.27: Efficient frontier of formation allocations

savings 0 – 5.5% can be achieved for a considerably lower level of risk when using the State Space approach. The main drawback of the State Space approach is that, currently, it takes roughly 30 seconds per formation compared to the Hold approach’s 2 seconds. However, as both methods are embarrassingly parallelisable this can be compensated for by using multiple cores and multiple computers.

4.10.4 Using a Nominal Speed Policy

The results of the preceding sections have shown that the Dynamic Programming approach produces superior results to the hold approach. However, the main difference between the two approaches is whether or not flights should continue along their flight path once they reach the rendezvous location. The Dynamic Programming approach has the added subproblem of calculating a proactive policy to follow until a delay is realised. It is therefore reasonable to question how much of the delay is mitigated by the deterministic part of the Dynamic Programming policy.

With this in mind we therefore assess a third approach (or at least an alternative use of the state space approach), using a nominal speed policy. We first calculate the DP solution, but then instead of assigning an SDP policy we apply a fixed control for the whole of the stochastic problem (as if the policy was the blue line of Figure 4.21).

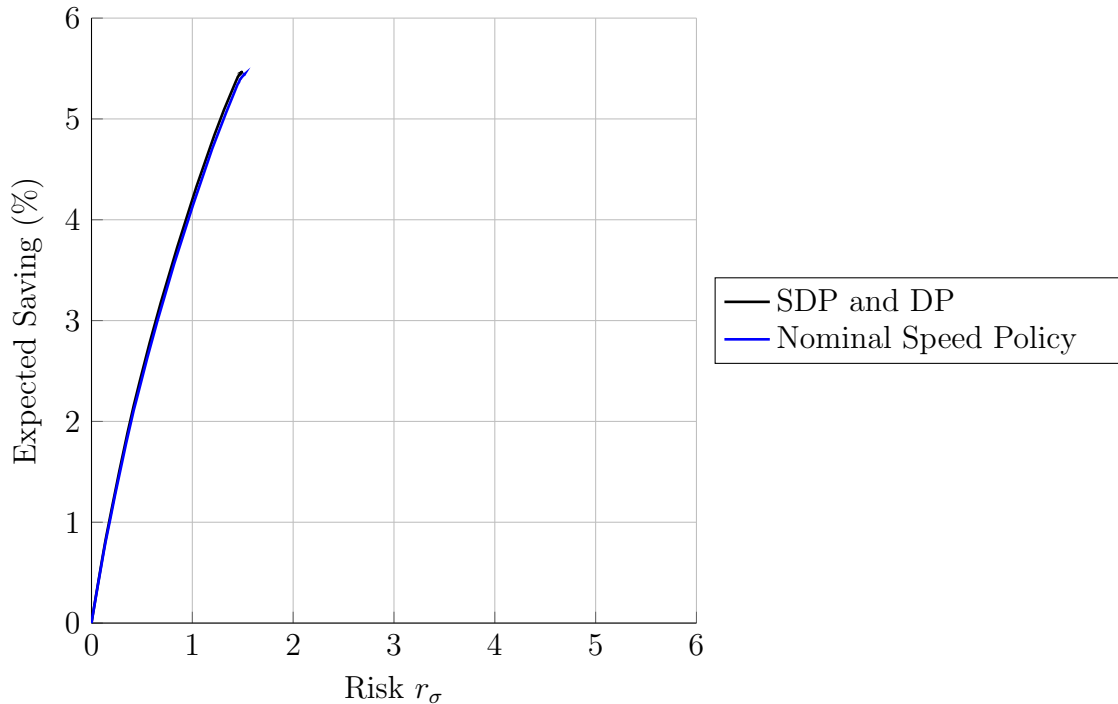


Figure 4.28: Efficient frontiers using full SDP and DP compared to a nominal speed policy and DP

That is, the aircraft simply flies its nominal cruise speed along its flight path until a delay is realised, then both flights follow their deterministic policy.

Calculating the efficient frontier for this approach yields an extremely interesting result and is plotted in Figure 4.28. What can be seen is that by using the nominal speed policy results in an almost identical efficient frontier to that of when the SDP is used. Clearly due to how uncertain the takeoff time is, it seems easier to just fly at a fixed nominal speed until the delay is realised. Therefore the majority of the improvement between the Hold approach and the State Space approach lies within the deterministic en-route policies. Interestingly, that means that there is not a great deal that can be done proactively to mitigate ground delay, instead a good reactive policy is more important.

4.11 Summary

With an introduction of uncertainty into the problem this chapter has investigated two possible and distinct strategies for mitigating the impact ground delay has on formation flight.

The first was a simplistic approach, instructing aircraft to enter a holding pattern

at the point of rendezvous to absorb any delay to the other aircraft. An important aspect of this approach relied on setting a suitable cutoff time, defining the maximum allowable time for one aircraft to hold and wait for the other to arrive. Any delay past the cutoff time produced losses corresponding to the cutoff time. High cutoff times produced good expected results but at the cost of the potential of large losses if a delay was still higher than the cutoff. Issues of practicality would limit cutoff times, likely up to about 60 minutes.

The second approach was a more in depth method, and the main focus of this chapter, using a State Space approach to model the ground delay problem. The approach involved splitting the problem into two closely linked subproblems. The deterministic problem, when both aircraft have taken off, was set up as Dynamic Program to be solved via Value Iteration. The solution to this deterministic problem was used within stochastic stage.

In order to account for issues arising from the discretisation of this state space a method of interpolation was added to the standard grid-based approach. This allowed for smoother cost functions and in turn smoother controls. Additionally, in order to increase the resolution of the grid around areas of greater non-linearity without being computationally prohibited, a quadtree based decomposition was implemented. This ‘shifted’ resolution to areas of greater non-linearity and allowed the linear approximation needed for interpolation.

The stochasticity was defined by airport-specific probability distributions modelled on historical data. For each flight the stochastic problem, when the other flight has not taken off, was modelled using a Stochastic Dynamic Program and solved using Value Iteration. Optimal Policies were calculated for any possible realisation of delay.

A notion of risk was introduced corresponding to the standard deviation of the results. The traditional MILP of previous chapters was adjusted to include a weighted penalty function on a formation’s risk. This allowed the risk tolerance to be tuned to filter out formations which were less robust. The idea of portfolio optimization was also introduced to produce efficient frontiers defining the reward corresponding to any level of risk-aversion.

Finally a Transatlantic case study compared the two methods against a list of 210 Transatlantic flights. Although similar expected savings were possible from both the Hold approach and the State Space approach, comparison of efficient frontiers clearly shows that the State Space approach produces significantly less-risky results for any level of reward. Interestingly it was shown that the reactive en-route policies

of the deterministic DP were more important. Using a fixed-speed-control for the uncertain part of the problem yielded almost identical efficient frontiers, prompting the conclusion that complex proactive policies may not be entirely necessary.

Chapter 5

Formation Flight Case Studies

5.1 Introduction

For formation flight to become a reality within the commercial flight sector, significant potential needs to be shown. It is therefore necessary to investigate a number of differing scenarios where formation flight may be of benefit. This chapter presents a comparison of three distinct data sets where formation flight could feasibly take place.

The objective of this chapter is to observe patterns and influencing factors affecting formations and the associated fuel savings. Firstly in Section 5.2, the three case studies will be presented, with Section 5.3 attempting to categorise them using a simple graph-theoretic approach. The results of applying the methodologies of Chapter 2 to each of the case studies will then be discussed. The results of using a 10% formation discount factor, for varying restrictions on the schedule is presented in Section 5.4. The correlations between a number of key values are measured and the results outlined in Section 5.5, whereby some of the features common amongst better-performing formations are discussed. Finally, in Section 5.6 we assess the impact the formation discount factor has, examining results for formation discount factors between 1-20%.

5.2 Case Study Data Sets

Firstly a set of 210 Transatlantic flights represents a region with great potential, with all flights travelling far enough and in similar enough directions to produce very favourable formations. Moving to a larger, yet more diverse, set of EasyJet flights

provides a different look at possible savings, with flights greatly ranging in distance and directions. Finally a look at a list of Singapore Airlines flights allows us to assess the potential when the airline acts more like a hub and spoke network.

5.2.1 Transatlantic Flights

The Transatlantic data set consists of 210 flights flying between the United States and Europe. Each route is based on a real flight including a specific aircraft type and scheduled departure time. As discussed in Section 2.6, these flights are taken from the OAG dataset for September 2001. While the flights have no specific company assigned to them, they are assumed to act like a single company, trying to optimize savings for the entire fleet rather than individual gain.

This list of Transatlantic flights provides a very likely scenario where formation flight might be employed. It has therefore been used throughout this thesis as a baseline for which to compare methods and results. Additionally 210 flights is a data set which is large enough to produce interesting results, whilst remaining small enough to not result in computational intractability. All 210 flights are distinct and result in 21,945 different formation combinations.

5.2.2 EasyJet European Flights

In the event of the adoption of formation flight within commercial aviation, it is likely that initially individual companies will join formations amongst their own fleet. As a result this would simplify the economic problem of how to allocate and share any formation savings.

Easyjet is the second largest European low cost carrier airline [105] and therefore makes an excellent company case study for the potential of formation flight. The entire data set consists of 8,750 easyjet flights over 7 days. As some flights run on multiple days the number of unique flights was 4238. Furthermore many routes were flown by multiple flights at different times of the day and so some flights were filtered and restricted from flying together.

For each of the 7 days there were between 1100 – 1400 flights, resulting in between 600,000 – 850,000 possible combinations needing evaluation. For each of these formation combinations the methods of Chapter 2 were used to calculate the optimal formation route and the corresponding cost.

Preliminary results indicated that there is not a significant difference between each of the seven days. Therefore, for clarity, only flights on a Monday are considered

in this chapter. This leaves 1313 flights to consider and results in around 860,000 formation combinations to consider, making this the largest of the three case studies.

5.2.3 Singapore Airlines Flights

Singapore Airlines is a major airline company serving flights from Southeast, East and South Asia to many domestic and international destinations. It acts as a good case study for a wide range of flight distances, ranging mostly between medium haul to super-long haul flights. The main difference between these flights and ones seen in the other case studies is that the vast majority of the routes are either flying from or to Singapore, acting like the hub of a hub-and-spoke network (discussed in Section 5.3), thus not all routes are natural candidates for formation flight.

The data set contains 417 different Singapore Airlines flights running over a 7 day period. The number of flights per day ranges between 228 and 242 resulting in a respective range of formation combinations of between 25,000 and 29,000. As with the EasyJet flights, only those flights on a Monday are considered. As a result there are 232 flights, equating to just under 27,000 combinations needing evaluation.

5.3 Comparing Airline Network Design

When investigating the potential for formation flight within commercial aviation it is important to analyse current airline network structures. A graph-theoretic approach will hopefully outline network features suitable for formation flight. Some simple analysis of the flight network design has been depicted in Figures 5.1, 5.2 and 5.3. Airports are represented as nodes of a graph, while flights are the arcs between any two nodes. The ‘degree’ of a node is defined as the number of arcs incident to the node. Therefore the degree of an airport is the total number of flights which fly to, or from that airport.

With this in mind, Figure 5.1 outlines the flight network as a representational graph-like structure. Then using Figure 5.1(b) as an example, node 1 has degree 3, while node 5 has degree 6. The three different graph structures are backed up by the circular connectivity plots of Figure 5.2. The airports (nodes) are represented as equally spaced points on a circle, ordered by the degree of each airport. Finally Figure 5.3 shows the sparsity of the entire network. The airports are first ordered by their longitude from lowest to highest, then the points represent where there is an arc leaving the first airport (the vertical location with lowest longitude at the top) and

travelling to the second airport (the horizontal location with the lowest longitude on the left) and thus every point corresponds to a flight.

It is widely observed [106,107] that the deregulation of the US domestic passenger aviation in 1978 resulted in a trend away from direct flights, to having larger airports (hubs) act as transfer points for flights, creating what is known as a hub-and-spoke network. Regional carrier airlines, such as Singapore Airlines, are an extreme example of this, whereby the main hub is Singapore Changi Airport (SIN), with the majority of their flights in some way involving SIN airport. An example representation of this kind of network is in Figure 5.1(c), which resembles a ‘hub-and-spoke’ topology with node 1 representing SIN airport.

A hub-and-spoke network is generally distinguishable by having a single high-degree node, the hub, connected to other nodes of much lower degree. While the Singapore Airlines flight list also has a handful of other cross-connections between other airports, as a distinguishing characteristic, the majority of the flights more closely resemble this kind of hub-and-spoke network. This is backed up by the ordered connectivity plot of Figure 5.2(c) and the network sparsity plot of Figure 5.3(c), showing that the entire network is focused around one node (SIN airport), with minimal interaction between other airports.

Conversely, EasyJet flights are mostly short and medium-haul and therefore choose to fly more point-to-point between airports. This creates what more closely resembles a ‘connected-network’ (as depicted in Figure 5.1(b)), with flights going in all directions to connect up a large geographical area (Europe in this case). Although in reality, networks like this are really just a number of hub-and-spoke networks connected together. The connectivity and sparsity of the network, shown in Figures 5.2(b) and 5.3(b) respectively, outlines a reasonably spread out and interconnected network vastly different from Singapore Airlines.

The 210 Transatlantic flights are between 26 US and 42 European airports, flown from west to east. As a result, the airports can be split into two disjoint sets, the US departure airports and the European destination airports. In graph theory, a network of this kind is referred to as a bipartite graph (Figure 5.1(a)). This type of airline network is not common, rather it is a result of the Transatlantic flights being a fictitious airline company. If it were a real airline company, this network would likely be interconnected with many domestic flights and a range of different transcontinental flights. As can be seen from Figure 5.3(a) the connectivity of the flight network is very sparse. As flights are only going in one direction, there is none of the symmetry apparent in Figure 5.3(b). As they are ordered by the airport’s longitude, only the

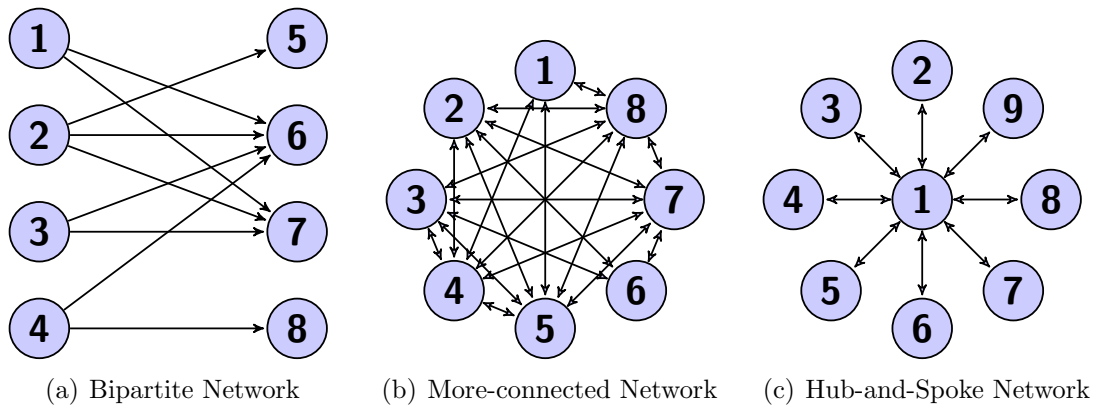


Figure 5.1: Representative graph structures of flight networks

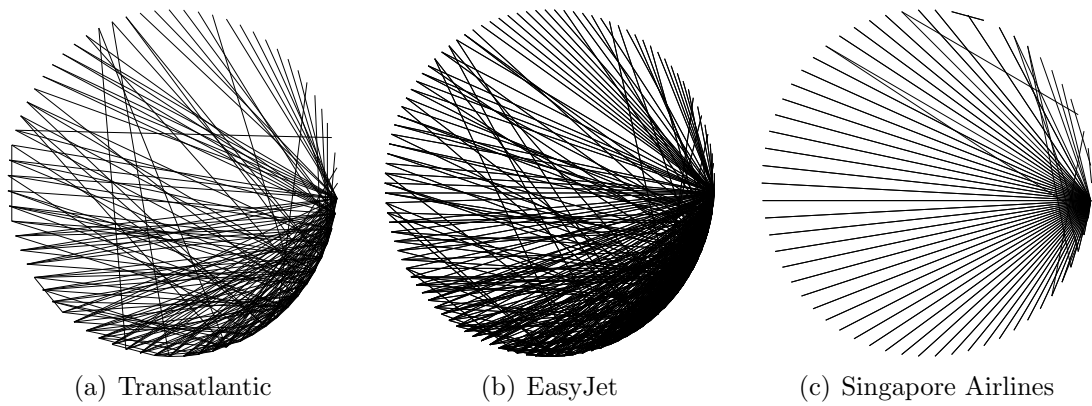


Figure 5.2: Circular connectivity ordered by airport degrees

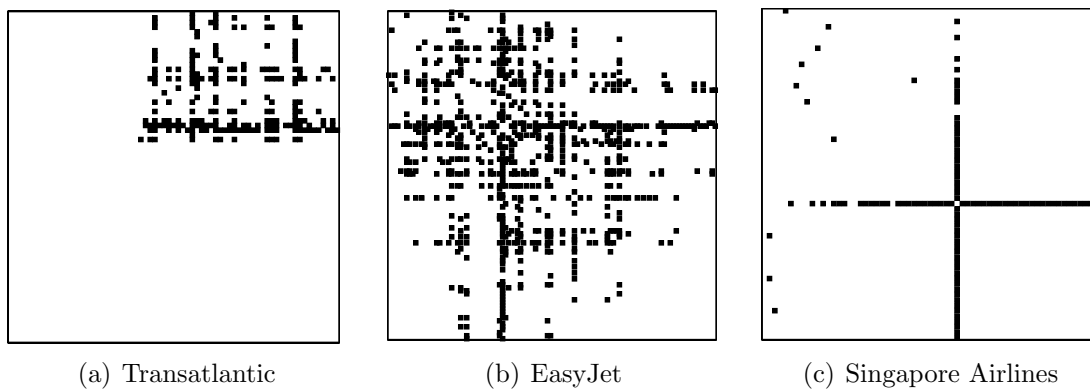


Figure 5.3: Flight network sparsity ordered by longitude

top right (corresponding to links between US and Europe) is populated.

Therefore it is clear that the three case studies in question represent a reasonable representation of today's flights while having significant distinction to make any comparisons interesting.

5.4 Results for Formation Discount Factor of 10%

The motivation behind the use of a geometric approach for routing, as outlined in Chapter 2, is to enable us to quickly calculate a cost for a potential formation. The speed of this calculation, for formations of size two, enables over 100,000 combinations to be assessed in under a minute. As explored in Section 2.5.2, the number of combinations requiring consideration increases according to the binomial coefficients, larger route lists will result in exponentially more combinations to consider. The Transatlantic route list and the Singapore Airlines route list are of a similar length and result in roughly 22,000 and 27,000 combinations respectively. The longer list of EasyJet flights results in a much larger number of combinations of roughly 870,000.

In order to impart some restrictions on scheduling a maximum permitted total change in takeoff times, Δt , is chosen, as outlined in Section 2.6.4. This restriction dictates the flexibility of a formation pair to change their takeoff times relative to one another. If a formation pair needs to change their takeoff times, in order to rendezvous at the required time, by a total time greater than Δt , then that formation combination will not be considered in the assignment stage.

With this in mind the results of this chapter follow the following solution process:

Enumeration For all possible combinations calculate the formation routes and corresponding costs of flying them.

Pre-Process Eliminate combinations exceeding scheduling constraint.

Assignment Given the costs of all combinations, assign a final fleet of formations to fly in order to minimise total cost.

In line with Chapter 2 for formations of size two, a fixed formation discount factor of 10% will be used throughout this section, while later, in Section 5.6, results will be presented for varying the discount factors between 1 and 20%.

The results for each of the case studies are now presented with the results shown for an increasing value of Δt (in minutes), representing the maximum total-allowable change a formation can make to their takeoff times. For lower values of Δt , there are

fewer possible formations to choose from, less choice results in lower overall savings. The objective function of the MILP is to minimise total cost (fuel burn), thus the only incentive is to minimise other values, such as deviation, because they are intrinsically linked to lower costs. The focus of these results, is to observe what happens to different values as the restrictions on the schedule are gradually lifted and the amount of choice of formations increases, therefore converging towards the unconstrained solution.

5.4.1 Overall Average Formation Saving

The metric which is minimised for in the MILP is the aircraft fuel burn, therefore the ideal measure of how ‘good’ a formation assignment is, is to measure the fuel saved as a result of allowing aircraft to fly in formation. This saving, as a percentage, is shown in Figure 5.4 and Tables 5.4 to 5.6. As this is the objective function of the entire optimisation, all flights are aiming for a better percentage saving. For $\Delta t = \infty$ (*i.e.* the unconstrained problem) the overall average saving for the entire fleets are 8.89%, 1.89% and 6.15% for Transatlantic, EasyJet and Singapore Airlines respectively.

This shows a clear distinction between the three case studies with Transatlantic flights saving the most on average and EasyJet saving the least. As an entire fleet EasyJet’s performance is fairly low, partly due to there being many more flights with many of them not being ideal for formation. Only around 45% of the flights in the final assignment are in formation (compared to 80 – 100% for the other two) and thus reducing the overall averages slightly.

What is more interesting is how well the saving responds to altering Δt . There are obviously very few suitable formations for a zero change in takeoff times, however by only allowing even just a Δt of 30 minutes, the assignments can achieve very reasonable savings. At around 120 minutes allowable change almost all of the unconstrained formation saving is achievable. Therefore, simply from an overall saving view point there needs to be some, but not necessarily a large amount, of flexibility in the takeoff times in order to get most of the fuel saving available from formation flight.

The spread of formation savings for all possible formations combinations, for each case study can be seen in Figure 5.5, whereby the savings are partitioned into intervals of 0.5% and the proportion which lie within that partition are counted. There is a clear shape for each case study, with Transatlantic flight savings ranging all the way up to 9.5% with the majority lying between 5 – 8% and very few with low savings, creating a clear negative skew. Conversely, EasyJet flights peak between 0 – 1% and

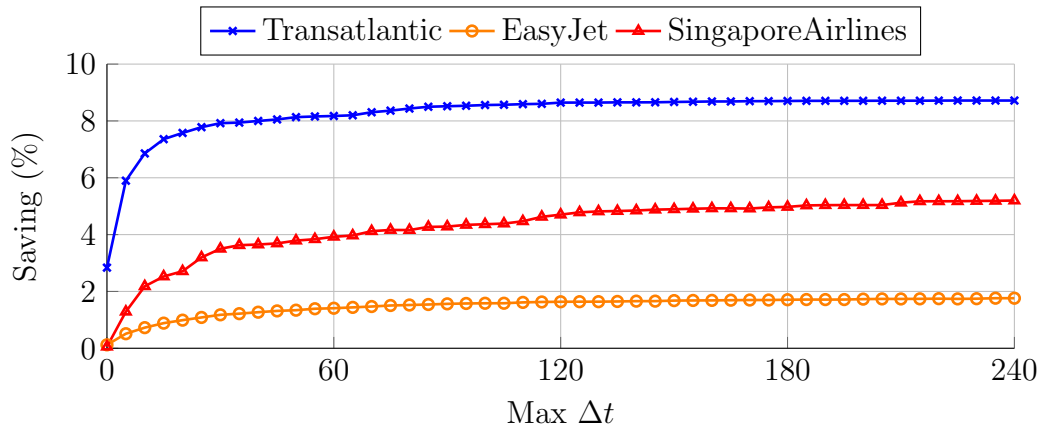


Figure 5.4: Total average saving (%)

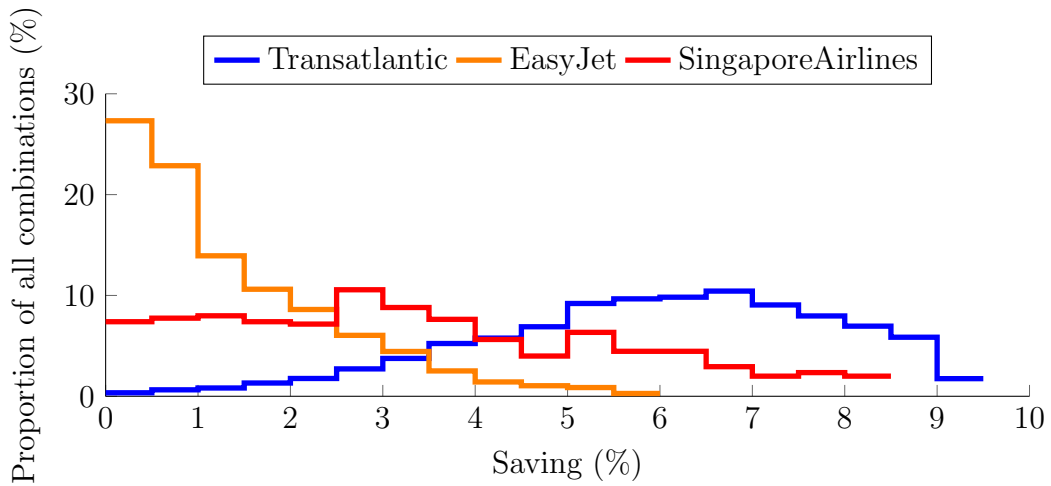


Figure 5.5: Distribution of formation saving (%)

quickly trail off, reaching maximums of 5 – 6% resulting in a clear positive skew. The Singapore Airlines flights, lie somewhere in the middle, with results spread reasonably evenly across all of 0 – 8.5%. Therefore what is clear from Section 5.3 and the results of Figure 5.5 is that we have three reasonably distinct case studies with a range of possible results.

5.4.2 Proportion of Flights Joining Formation

For a given assignment, a rough guide of how well the list of flights is suited to formation flight is the proportion of formations to solo flights. The results for an increasing Δt are plotted in Figure 5.6, whereby as an example, a proportion of 75% means three quarters of the flights are assigned into formation, while the remaining quarter fly solo. For the Transatlantic flights there is a very rapid trend towards all

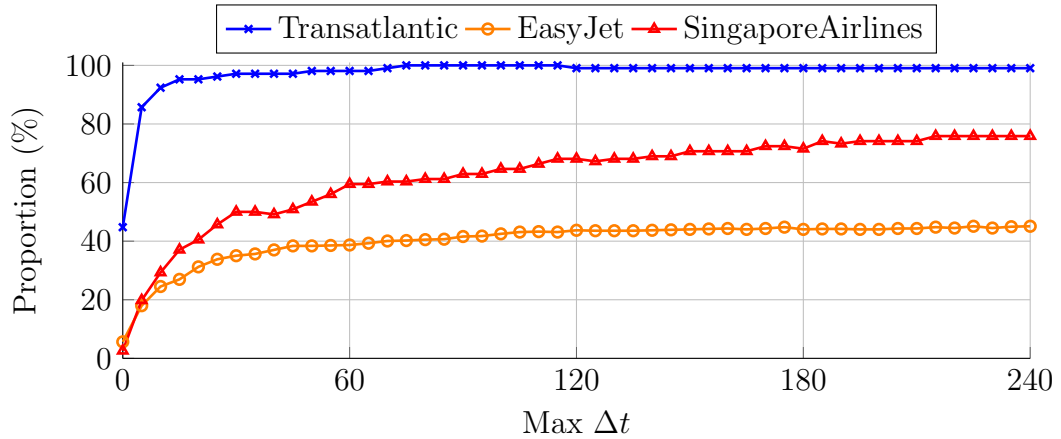


Figure 5.6: Proportion of aircraft assigned into a formation

flights being part of a formation (*i.e.* 100%). For Singapore Airlines the approach is more gradual, eventually reaching just over 75%. EasyJet have the lowest of the three, but levels off fairly quickly at about 45%, however the EasyJet route list is much larger than the other two at about 1,300 flights. Therefore, although this overall proportion really reflects the route list as a whole, with the many hundreds of short-haul flights within the EasyJet flight list likely bringing down the overall percentage value, what is more important is the response to the changing Δt .

5.4.3 Flight Deviations

One of the fundamental drawbacks of formation flight is that aircraft will need to deviate from their solo routes, flying out of their way in order to rendezvous with other formation members. The aim being, that any deviation in distance flown will be compensated for by the drag-reduction formation flight offers. Intuitively, formations will want to reduce this deviation as it directly effects their overall cost.

The deviation results for the Transatlantic, EasyJet and Singapore Airlines are shown in Tables 5.1, 5.2 and 5.3 respectively. These tables outline the average and maximum deviation values for both the total formation and for each individual flight within a formation. These results are also plotted in Figures 5.7(a) and 5.7(b), showing both the total average deviation in km and as a proportion of the solo distance.

What can be seen is that as the problem becomes less constrained, *i.e.* as Δt increases, the Transatlantic flights quickly move towards lower levels of deviation. Conversely, EasyJet and Singapore airlines decrease slightly, but mostly remain fairly constant. Furthermore it can be seen from the values outlined in Tables 5.1, 5.2 and 5.3, that these levels of deviation are overall very low. Low average deviations

Table 5.1: Transatlantic: Deviation (km) in route distance between formations and their solo routes.

Δt	No. Forms	Common airport	Formation total		Per aircraft	
			Avg	Max	Avg	Max
0	47 (45%)	11 (23%)	229 (3.3%)	1125 (17.2%)	114 (3.3%)	898 (14.5%)
5	90 (86%)	27 (30%)	183 (2.7%)	838 (14.5%)	91 (2.7%)	838 (14.5%)
10	97 (92%)	28 (29%)	121 (1.8%)	691 (11.3%)	61 (1.8%)	592 (9.6%)
15	100 (95%)	36 (36%)	112 (1.6%)	702 (9.7%)	56 (1.6%)	481 (6.0%)
30	102 (97%)	48 (47%)	86 (1.2%)	485 (6.3%)	43 (1.2%)	424 (6.3%)
60	103 (98%)	50 (49%)	76 (1.1%)	386 (5.3%)	38 (1.1%)	291 (4.6%)
120	104 (99%)	53 (51%)	51 (0.7%)	352 (4.3%)	26 (0.7%)	235 (3.1%)
240	104 (99%)	58 (56%)	40 (0.5%)	451 (5.3%)	20 (0.5%)	356 (3.9%)
Inf	105 (100%)	73 (70%)	27 (0.4%)	317 (3.8%)	14 (0.4%)	184 (2.6%)

Table 5.2: EasyJet: Deviation (km) in route distance between formations and their solo routes.

Δt	No. Forms	Common airport	Formation total		Per aircraft	
			Avg	Max	Avg	Max
0	37 (6%)	15 (41%)	52 (3.1%)	125 (8.4%)	26 (3.1%)	102 (4.8%)
5	118 (18%)	42 (36%)	54 (3.2%)	239 (10.4%)	27 (3.2%)	152 (6.0%)
10	161 (25%)	63 (39%)	47 (2.8%)	195 (10.7%)	23 (2.8%)	122 (8.6%)
15	177 (27%)	76 (43%)	48 (2.9%)	195 (10.7%)	24 (2.9%)	122 (8.6%)
30	230 (35%)	91 (40%)	47 (2.9%)	195 (10.7%)	23 (2.9%)	143 (8.6%)
60	254 (39%)	106 (42%)	42 (2.6%)	239 (10.7%)	21 (2.6%)	152 (8.6%)
120	287 (44%)	119 (41%)	36 (2.3%)	151 (8.3%)	18 (2.3%)	143 (5.6%)
240	296 (45%)	129 (44%)	34 (2.2%)	143 (6.3%)	17 (2.2%)	143 (5.6%)
Inf	308 (47%)	141 (46%)	31 (2.1%)	109 (5.4%)	16 (2.1%)	73 (3.5%)

Table 5.3: Singapore Airlines: Deviation (km) in route distance between formations and their solo routes.

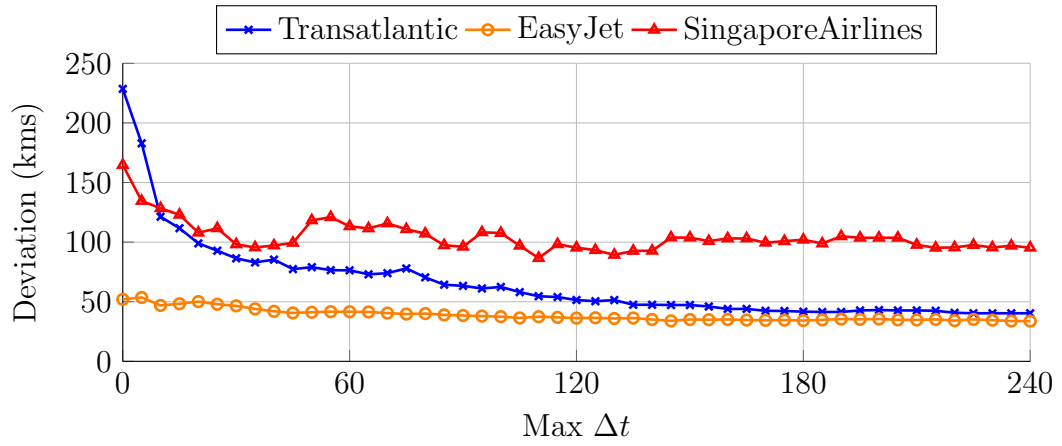
Δt	No. Forms	Common airport	Formation total		Per aircraft	
			Avg	Max	Avg	Max
0	3 (3%)	3 (100%)	165 (2.7%)	281 (4.7%)	82 (2.7%)	164 (3.7%)
5	23 (20%)	22 (96%)	135 (1.8%)	560 (5.4%)	67 (1.8%)	560 (4.9%)
10	34 (29%)	33 (97%)	128 (1.9%)	560 (8.5%)	64 (1.9%)	560 (8.5%)
15	43 (37%)	42 (98%)	123 (2.0%)	560 (8.5%)	61 (2.0%)	560 (8.5%)
30	58 (50%)	57 (98%)	98 (1.7%)	515 (8.5%)	49 (1.7%)	515 (8.5%)
60	69 (59%)	68 (99%)	113 (2.0%)	770 (12.5%)	57 (2.0%)	770 (12.5%)
120	79 (68%)	78 (99%)	95 (1.6%)	1082 (12.5%)	48 (1.6%)	1082 (12.5%)
240	88 (76%)	86 (98%)	95 (1.5%)	1289 (10.1%)	48 (1.5%)	1082 (10.1%)
Inf	96 (83%)	93 (97%)	49 (0.7%)	1140 (8.1%)	24 (0.7%)	868 (5.7%)

of 1 – 2% mean that most flights are not going very far out of their way in order to join in formation. The maximum deviation values really only correspond to a single formation and thus the results tend to fluctuate. It is therefore worth noting that even with low average savings, there may still be a few formations which have to deviate a larger amount.

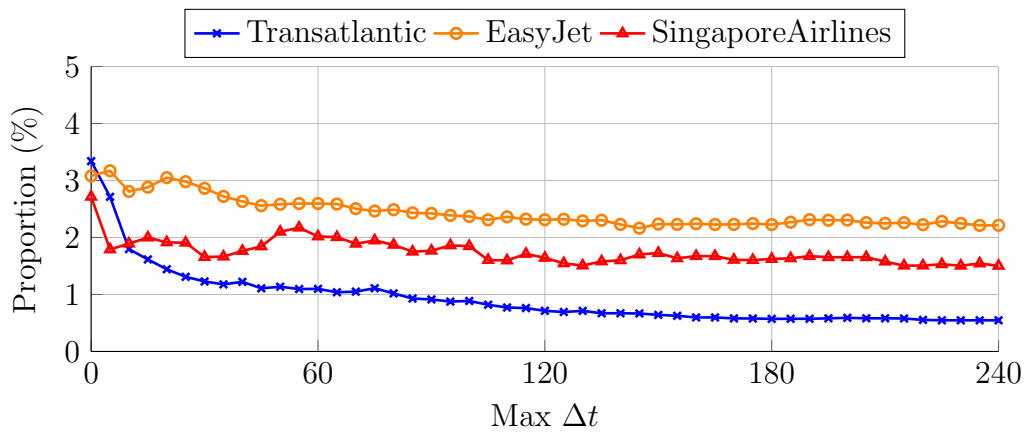
A linked factor to the level of deviation is whether the flights making up a formation have a common airport. That is, if they either depart from, or arrive at the same airport as the other formation member. Clearly if an airport is shared then there is likely to be a lower level of deviation. The proportion of the assigned formations who share a common airport are in Tables 5.1, 5.2 and 5.3 with values plotted in Figure 5.8. Due to the type of network, explored in Section 5.3, Singapore Airlines flights almost all fly to or from Singapore Changi Airport. EasyJet flights are also more or less constant at a level of about 40%, while the Transatlantic flights trend towards about 55%. Finally, while the consistency of the Singapore Airline and EasyJet flights are reflected in the same consistency in the deviation plots of Figure 5.7(b), the Transatlantic flights move towards lower deviation is matched by the increase in proportion of common airports.

5.4.4 Utilisation

It is of interest to assess how well an assignment performs when compared to the maximum it could theoretically achieve. This metric of ‘Utilisation’ was introduced in Section 2.6.5 and gives an overall value for a list of formations suitability to formation



(a) Total Average Deviation Distance



(b) Total Average Deviation as a proportion of solo distance

Figure 5.7: Deviation in distance between solo and formation flight

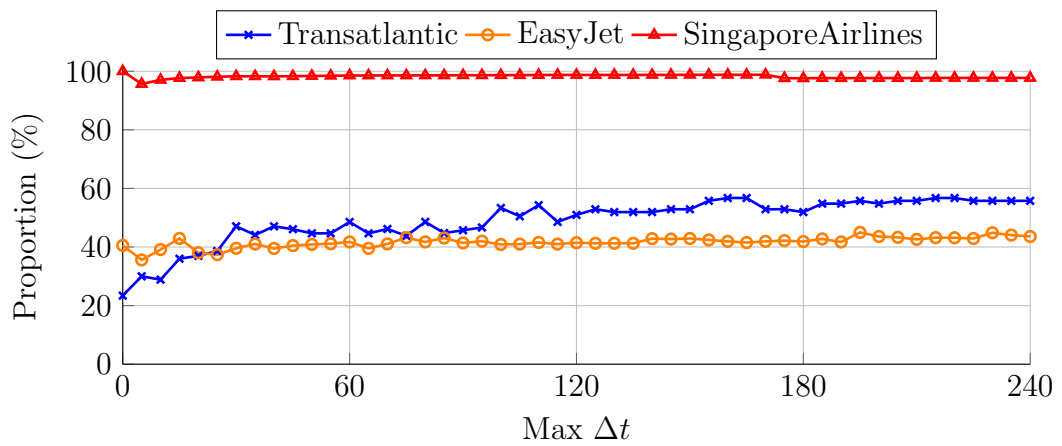


Figure 5.8: Proportion of formations who share a common airport

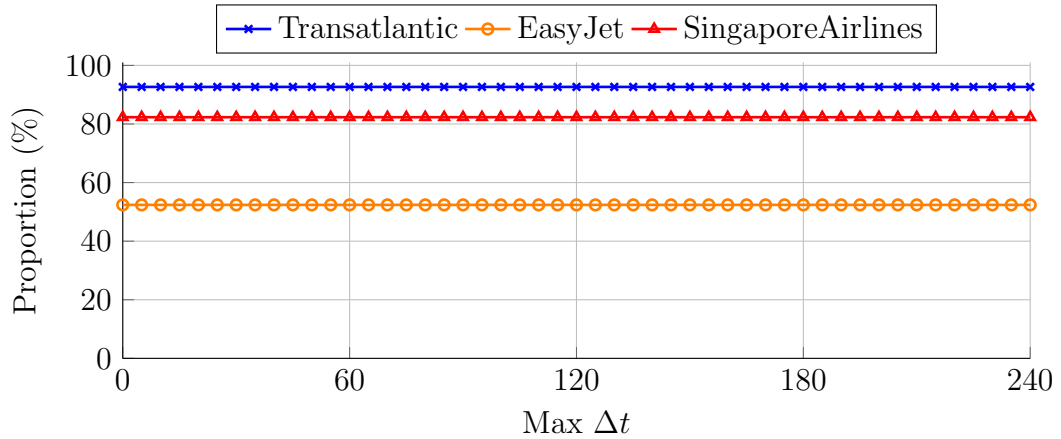


Figure 5.9: Average proportion of cruise to non-cruise flight

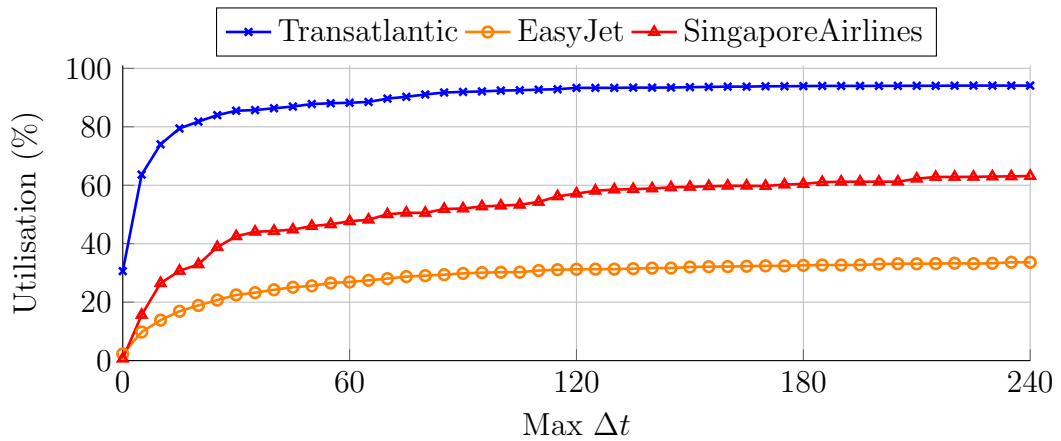


Figure 5.10: Utilisation of potential formation saving

flight. The values in Tables 5.4, 5.5 and 5.6, plotted in Figure 5.10, show a clear distinction between the three case studies. The Transatlantic flights can achieve anywhere up to a very impressive 96% utilisation for $\Delta t = \infty$, that is, out of the 9.3% theoretical-maximum saving achievable, roughly 8.9% was realised. The EasyJet and Singapore Airlines flights respectively can achieve levels of utilisation of up to roughly 35% and 75%.

The utilisation factor is a result of a number of different components, such as geographical suitability and flight distance but importantly dictated by the theoretical maximum. This theoretical max is calculated based on the entire cruise proportion of each flight is being in formation. Therefore a higher proportion of cruise to non-cruise flight, equates to a greater theoretical maximum saving. This proportion plotted in Figure 5.9 and covered in greater detail in Tables 5.4, 5.5 and 5.6, is a predefined constant, based on the solo flight list and so does not change based on assignment.

Table 5.4: Transatlantic: Cruise proportions and corresponding utilisation factors

Δt	Aircraft Cruise %				Fleet Utilisation	
	Min	Avg	Max	Saving	Theoretical Max	Utilisation (%)
0	84	93	95	2.83	9.3	30.6
5	84	93	95	5.89	9.3	63.6
10	84	93	95	6.85	9.3	74.0
15	84	93	95	7.35	9.3	79.4
30	84	93	95	7.92	9.3	85.5
60	84	93	95	8.17	9.3	88.2
120	84	93	95	8.64	9.3	93.3
240	84	93	95	8.71	9.3	94.1
Inf	84	93	95	8.89	9.3	96.0

Table 5.5: EasyJet: Cruise proportions and corresponding utilisation factors

Δt	Aircraft Cruise %				Fleet Utilisation	
	Min	Avg	Max	Saving	Theoretical Max	Utilisation (%)
0	0	52	90	0.12	5.2	2.3
5	0	52	90	0.51	5.2	9.8
10	0	52	90	0.72	5.2	13.9
15	0	52	90	0.88	5.2	16.9
30	0	52	90	1.17	5.2	22.5
60	0	52	90	1.40	5.2	26.9
120	0	52	90	1.63	5.2	31.2
240	0	52	90	1.76	5.2	33.6
Inf	0	52	90	1.89	5.2	36.1

Table 5.6: Singapore Airlines: Cruise proportions and corresponding utilisation factors

Δt	Theoretical Max (%)	Saving Achieved (%)	Utilisation (%)
0	8.20	0.05	0.7
5	8.20	1.28	15.6
10	8.20	2.18	26.5
15	8.20	2.52	30.7
30	8.20	3.50	42.5
60	8.20	3.92	47.7
120	8.20	4.70	57.1
240	8.20	5.19	63.2
Inf	8.20	6.15	74.8

5.4.5 Summary

The results of this section show a range of values and their interaction between the allowable change in takeoff times Δt . As the constraint is gradually lifted, values tend towards their unconstrained ideal. It is clear from Section 5.4.1 that even with a relatively low Δt of 30 minutes, formations can still achieve very reasonable savings. While Section 5.4.3 shows a general tendency to move towards lower levels of deviation (with Transatlantic flights being the most effected), this corresponds to more formations sharing a common airport. Finally the Utilisation factors of Section 5.4.4 show that the Transatlantic flights are the best suited to formation, followed by Singapore Airlines and then by the EasyJet flights. With the proportion of cruise to non-cruise flight being a major contributor.

5.5 Spread of Results

The results of Section 5.4 look at summarising values for a collection of results, *i.e.* averages, maximums and minimums for an assignment of formations. Moving from performance analysis to correlations, trying to understand why results are as they are, and in turn identify factors that affect formation flight performance. For the optimal assignments, based on a Δt of 30 minutes, for the Transatlantic, EasyJet and Singapore Airlines some results are plotted in Figures 5.11 to 5.13. Each point on these graphs corresponds to an assigned formation, with the location representing the relationship between the saving percentage and some other chosen variable (the correlation level is also noted and will be outlined fully in Section 5.5.1).

Looking solely at the x -locations, *i.e.* the formation saving, one sees the same pattern that is observable in Figure 5.5. That is, the Transatlantic flights are negatively skewed towards the higher percentage saving, the EasyJet skew positively towards the lower savings and the Singapore Airlines flights spread more evenly across the entire range. The results of this section aim to explore what is the main impacting factors on these formation savings.

First observing the relationship between cruise proportion (discussed in Section 5.4.4) and saving, plotted in Figure 5.11 clearly shows a positive trend towards higher saving through a higher proportion of cruise flight. While, for the higher levels of saving, proportions of 80 – 90% are required, there are formations with that level of cruise proportion achieving much lower savings.

Secondly Figure 5.12 shows percentage saving against the solo formation distance.

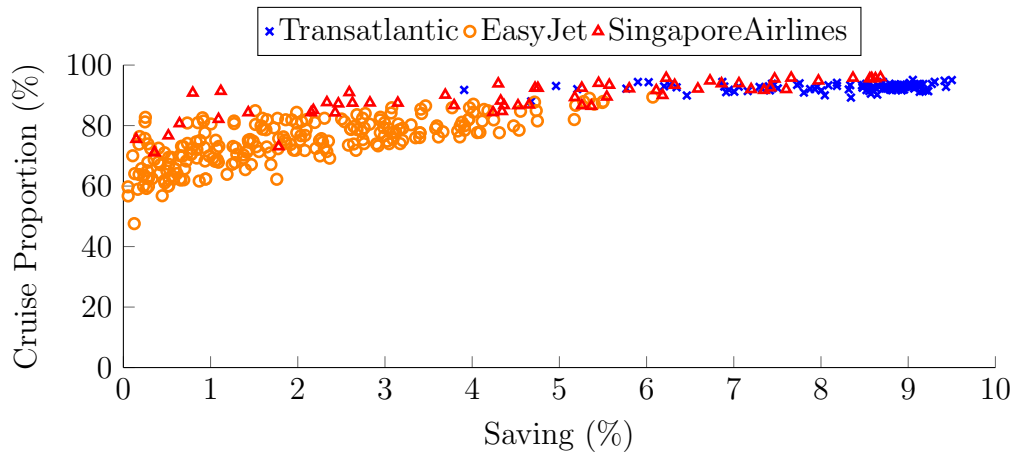


Figure 5.11: MILP $\Delta t = 30$: Cruise proportion against saving. Correlation: Transatlantic = 0.67, EasyJet = 0.74, Singapore Airlines = 0.82

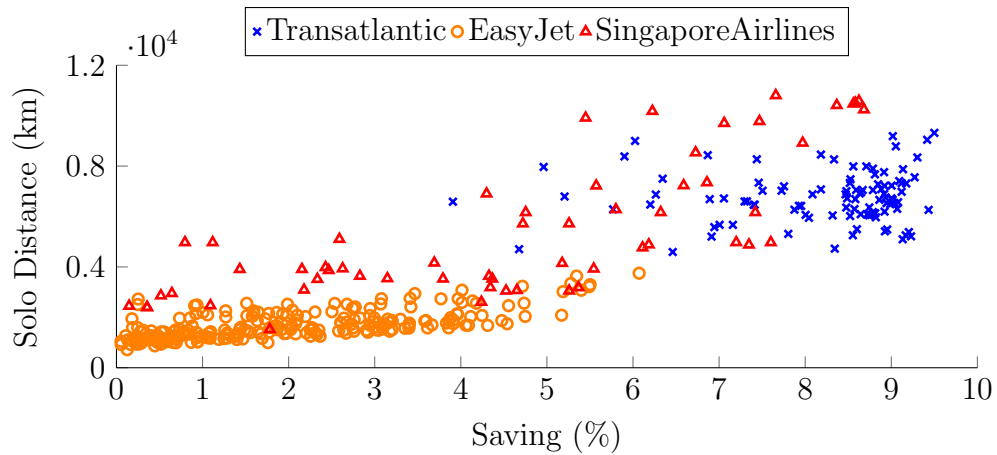


Figure 5.12: MILP $\Delta t = 30$: Solo distance against saving. Correlation: Transatlantic = 0.07, EasyJet = 0.70, Singapore Airlines = 0.70

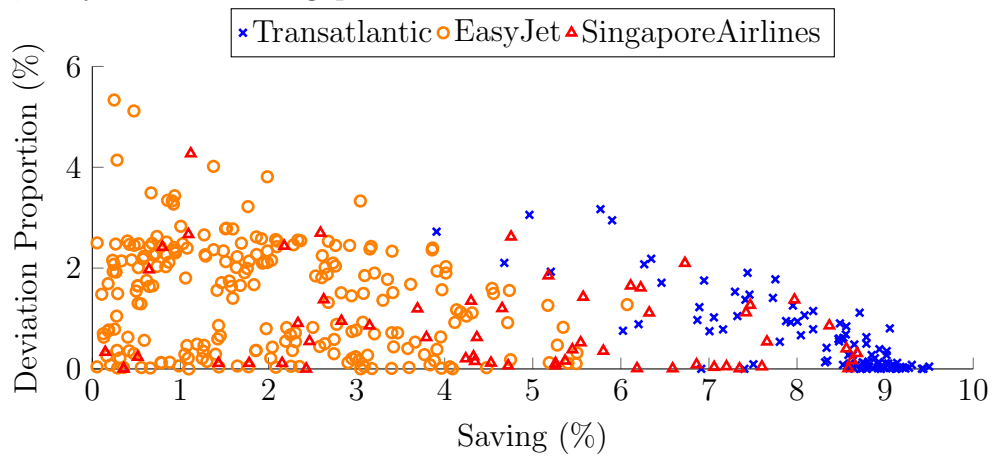


Figure 5.13: MILP $\Delta t = 30$: Deviation proportion against saving. Correlation: Transatlantic = -0.79, EasyJet = 0.01, Singapore Airlines = -0.16

Solo distance has not yet been directly discussed in this chapter as it is more an input than a result, however, it has been an aspect which has direct effect on the cruise proportion and deviation. While this spread of results, shows a general trend towards higher saving along with a higher solo distance, what it really shows is that there is more of a barrier to higher percentage savings. Formations likely need to be longer than 4000 km to achieve savings over 6%, which in turn would mean a cruise proportion likely above 80%. Conversely, longer flights do not guarantee higher savings percentages, as there are many flights achieving 6–7% which are substantially longer than those with 9%.

Finally the relationship between the proportion of deviation and the saving percentage is plotted in Figure 5.13. There is a clear spread, showing that formations with a high proportion of deviation likely result in lower savings, however there is less of an overall trend.

What will be shown in Sections 5.5.1 and 5.5.2 is that these spreads can be quantified using correlation coefficients in order to assess which variables have more impact on saving percentages.

5.5.1 Correlations

We now consider the implications of dependence on seemingly random variables through the comparison of correlation coefficients. Using the same values mentioned throughout this chapter, such as saving, deviation and flight distance, we aim to measure the statistical relationships between them.

First the correlation coefficients are defined by the Pearson product-moment correlation coefficient for a population (*i.e.* this is not a sample as all the values are known). For two variables X and Y , with covariance $\text{Cov}(X, Y)$ and standard deviations σ_X and σ_Y this is defined as:

$$\rho_{X,Y} = \frac{\text{Cov}(X, Y)}{\sigma_X \sigma_Y}. \quad (5.1)$$

This value ranges between +1 and –1 inclusive, with 1 meaning a total positive correlation, 0 no correlation and -1 total negative correlation. The stronger a correlation (*i.e.* the higher the absolute value), the stronger the dependence between the two variables. That is, a 0 correlation means two variables are more or less independent (or at least act randomly). A strong positive correlation means that an increase in X will likely result in an increase in Y , while a strong negative correlation would mean an increase in X would likely result in a decrease in Y .

5.5.2 Cross-correlation

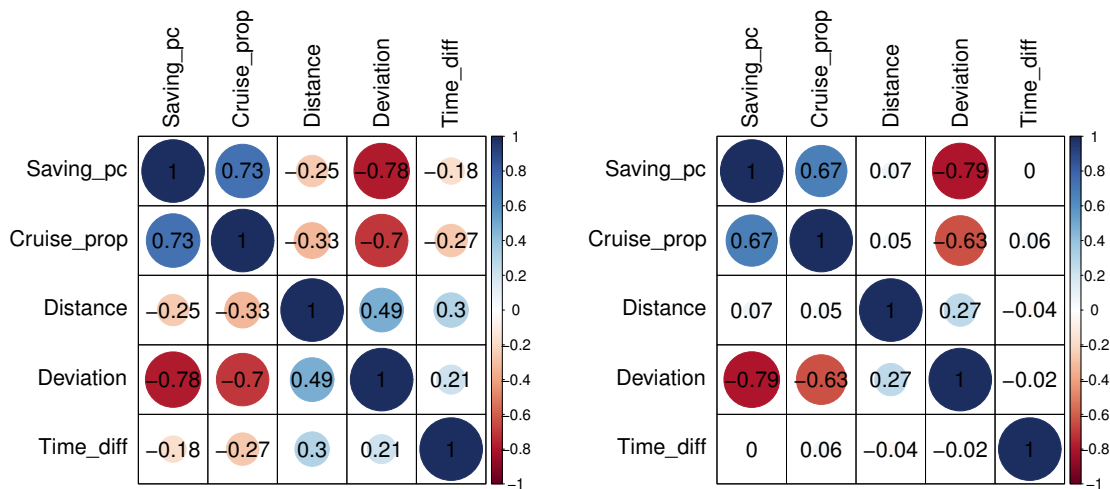
With this idea in mind it is interesting to observe how particular variables correlate to each other, this is accomplished using a ‘cross-correlation plot’. This kind of plot is a grid of cells, where each column and each row corresponds to a particular variable, then each cell contains the correlation value between each row and column variable. For additional clarity, a circle, with radius proportional to the correlation, is plotted. The plots of Figures 5.14, 5.15 and 5.16 depict the cross-correlation of a number of chosen variables. There are five variables considered for comparison:

- Saving_pc - The percentage saving against solo flight;
- Cruise_prop - The proportion of the flight at cruise;
- Distance - The solo flight distance;
- Deviation - The deviation in distance between solo and formation flight;
- Time_diff - The total difference in takeoff time required.

5.5.3 Results

The cross-correlations are calculated for each of the three case studies and for two sets of formations (Figures 5.14, 5.15 and 5.16). Firstly for the optimally-assigned solution formations for a Δt of 30 minutes; secondly for the list of all possible formation combinations (*e.g.* the unassigned list of formations) which are favourable (*i.e.* those that produce a saving). The reason for including results for both is to see if formations chosen by the assignment process differ greatly from the overall ‘population’.

There are clear similarities observable between all three of the case studies. Firstly there is almost no correlation between the time difference required for scheduling the formations and any other parameter. Therefore the Δt constraint used is essentially a random filter on the possible formations. This is an interesting conclusion, as discussed in Section 5.4, limiting the allowable change to the schedule is really just restricting ‘choice’ and results in lower overall savings. What this shows is that while this restriction is random and makes sense to include it, there is little that can be done to plan for it. Secondly the correlation between saving and cruise proportion for the assigned formations, is 0.67, 0.74 and 0.82 for the Transatlantic, EasyJet and Singapore Airlines respectively. This implies that flights which spend a greater proportion of their flight in cruise will likely result in a greater percentage saving.



(a) All formation combinations Unassigned

(b) MILP assignment $\Delta t = 30$

Figure 5.14: Cross-correlation of Transatlantic Case Study

This intuitively makes sense, from the discussion in Section 5.4.4, that in order to get a better theoretical maximum, a larger proportion of cruise is needed.

Finally when comparing within each case study, the unassigned formations to those which are assigned for a Δt of 30 minutes, there is minimal change. The same conclusions about the correlations can be made for both the assigned and the unassigned results. At most, the changes are around 0.2 in either direction, mainly shifting those from minimally-correlated to totally-uncorrelated and vice versa. Importantly, what this implies is that these correlations are mostly a feature of the case study and not the assignment process.

Looking just at the Transatlantic flights (Figure 5.14) what is initially clear, but can also be observed in the spread of Figure 5.13, is the strong negative correlation between percentage saving and deviation at -0.79 (-0.78 for the unassigned). Thus, flights with a greater deviation likely result in a lower saving as there is strong negative correlation. Conversely, what is also observable is that there is very little correlation (0.01) between the saving and the solo distance for the Transatlantic flights. As all the flights are long-haul, distance does not act as a distinguishing feature between the formations. Therefore, given the list of the unassigned Transatlantic flights' formations one could predict the formations with the best saving percentage using the other four variables using these correlations. One should pick those formations with low deviations and high cruise proportion and can effectively disregard the other variables. Interestingly what we will see is that this is not the case for EasyJet and

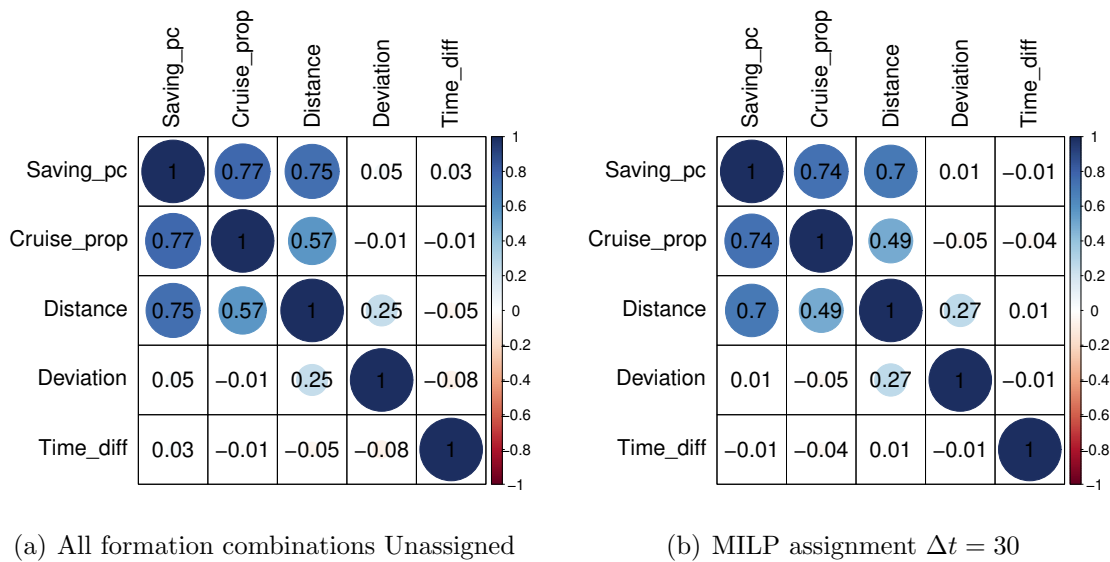


Figure 5.15: Cross-correlation of EasyJet Case Study

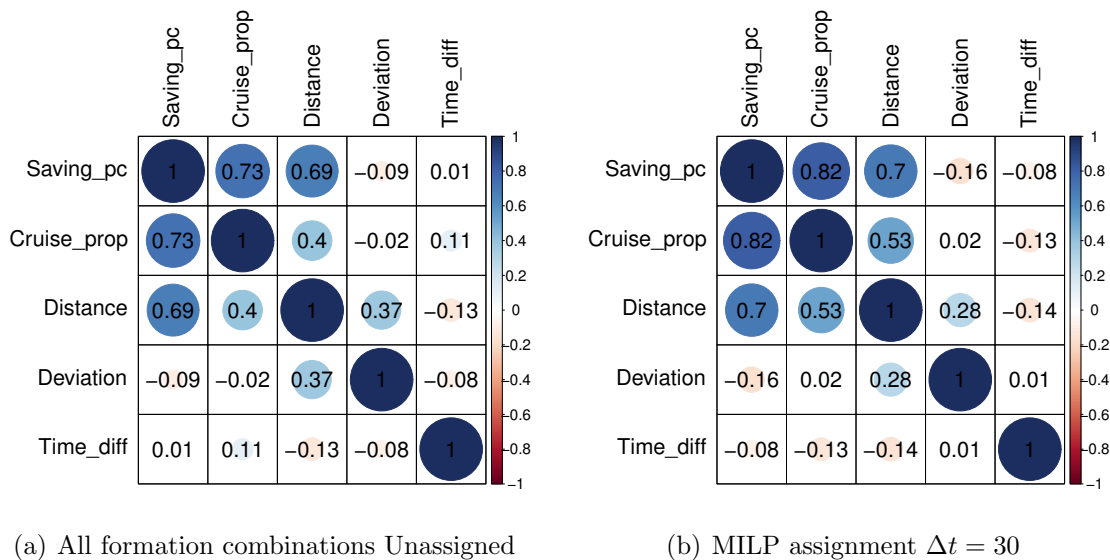


Figure 5.16: Cross-correlation of Singapore Airlines Case Study

Singapore Airlines.

The EasyJet and Singapore Airlines correlations present in very similar ways (with some values almost identical) but there is clear and drastic change from the Transatlantic. Firstly formation distance, which has had almost no correlation for the Transatlantic flights (0.07), has a strong positive correlation to saving percentage for EasyJet (0.7) and Singapore Airlines (0.7), which is also observable in Figure 5.12. This is likely due to the larger variety of flight distances within the flight list compared to the Transatlantic flights. EasyJet have shorter flights, both short and medium-haul, linking back to the discussion on flight distance and utilisation in Section 5.4.4. Shorter flights, inherently mean a lower proportion of cruise flight available for formation and thus less savings achievable. Therefore flight distance becomes more of a distinguishing factor within a diverse list of formations.

There is also a change in the correlation between saving and deviation. Deviation, which has a strong negative correlation for the Transatlantic flights (-0.79), now has almost no correlation for the EasyJet (0.01) and Singapore Airlines (-0.16) flights. As can be seen in Figure 5.7 and Tables 5.1, 5.2 and 5.3 the EasyJet and Singapore Airlines flights have a much higher level of deviation present than Transatlantic. Thus, as most flights have a reasonable amount of deviation, this becomes less of a differentiating feature between formations.

Therefore if given a more diverse range of flights lists such as the EasyJet and Singapore Airline flights the indicators of formations with potentially high percentage savings are different from that of the Transatlantic flights. One should aim for flights which have a longer solo distance, as they will be able to offset deviations, achieve higher cruise proportions and thus more savings.

5.6 Sensitivity to Formation Discount

Throughout this thesis, for formations of size two, a fuel saving discount of 10% has been assumed when flying in formation. However, due to this value only being an estimate, based on a variety of aerodynamic models and flight tests [21, 26, 27, 40], it is of interest to observe what effect this value has on the overall saving.

Therefore, we now look at the routing and assignment results for the three flight case studies, for the problem with no scheduled constraints (*i.e.* $\Delta t = \infty$). The results for an increasing formation discount factor, as a percentage, between 1 and 20%, are now presented. For clarification, let us first define the formation discount, $\hat{\lambda}_f$, as the percentage of fuel-saving applied during the formation stage of flight. This

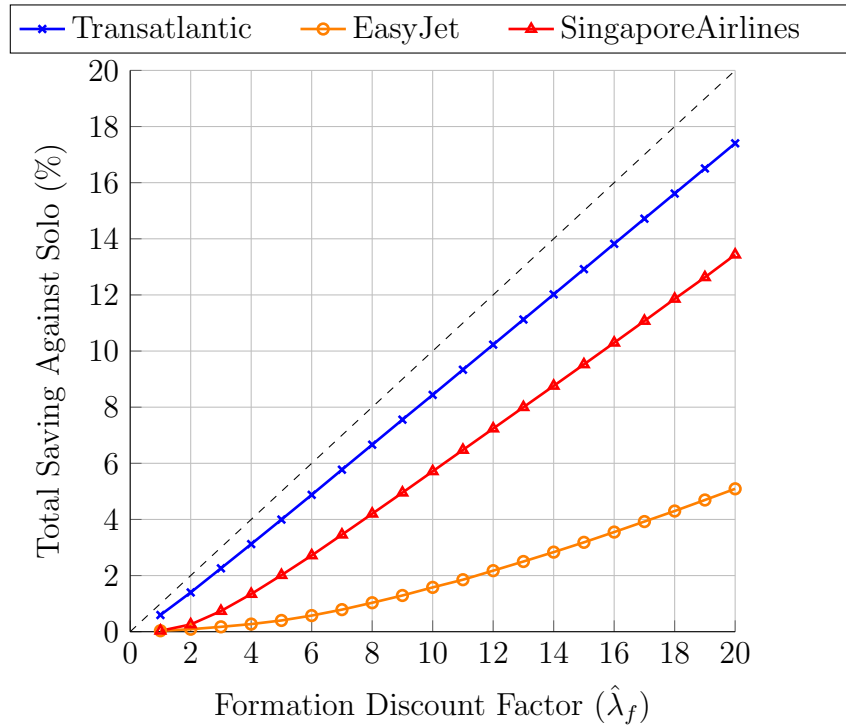


Figure 5.17: Relation between discount factor and overall saving achieved

value relates to the value λ_f of Chapter 2 by $\hat{\lambda}_f = 100 \times (1 - \lambda_f)$, and is essentially the same value expressed as a percentage.

Lower values of $\hat{\lambda}_f$, mean a lower level of incentive for flying in formation, while higher levels increase this incentive. The focus of these results, therefore, is to observe what happens to different values as incentive for flying in formation is increased or decreased, with the same measures, introduced in Section 5.4 being discussed.

5.6.1 Overall Average Formation Saving

The overall objective is to minimise cost, which is analogous to maximising saving, therefore all flights are aiming for the best percentage saving they can achieve. The overall saving, as a percentage, against solo flight is shown in Figure 5.17. A range of possible saving are present in the results for differing $\hat{\lambda}_f$, with the same general ordering of Transatlantic doing ‘best’ followed by Singapore Airlines and the EasyJet. What is important to observe in these results is the shape of the curve, and the response to an increase/decrease in $\hat{\lambda}_f$. The results for the Transatlantic show a stronger linear-relationship between $\hat{\lambda}_f$ and overall saving, while the other two have more of an offset with less of a direct response.

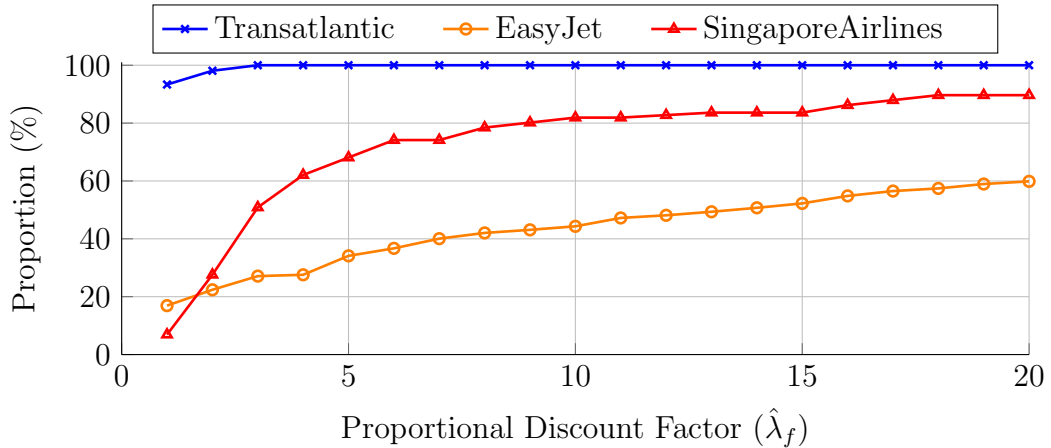


Figure 5.18: Proportion of aircraft assigned into a formation

5.6.2 Proportion of Flights Joining Formation

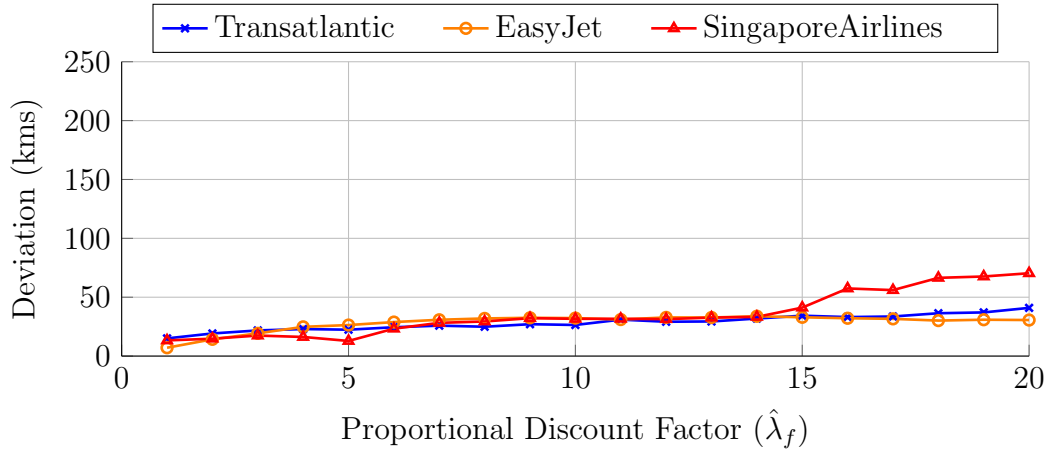
The suitability of a list of flights for formation flight can be shown by the proportion of formations to solo flights. The results of Figure 5.18, shows that the Transatlantic flights are highly suited to formation flight, even for very low discount factors. While the Singapore Airlines grows to around 80%, the proportion begins very low and shows a level of unsuitability for $\hat{\lambda}_f$ less than about 5%. Finally, the growth for the EasyJet flights is very steady, almost linear, between 1 and 20% but for much lower proportions, with many unsuited to formation flight.

5.6.3 Flight Deviations

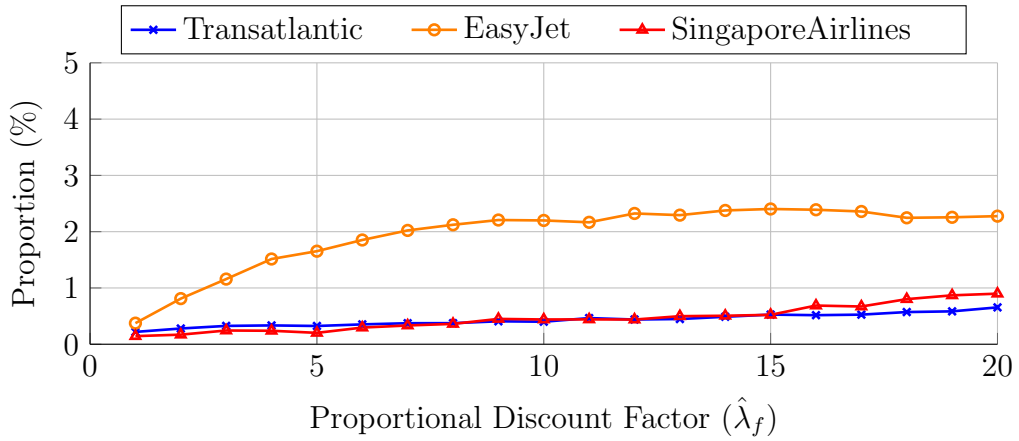
The deviation results for the Transatlantic, EasyJet and Singapore Airlines are plotted in Figures 5.19(a) and 5.19(b), showing both the total average deviation in km and as a proportion of the solo distance.

What can be seen is that while deviation should be incentivised by greater levels of $\hat{\lambda}_f$, the levels of deviation do not change by much. The results, in Figure 5.19(b), for EasyJet are more interesting, where low $\hat{\lambda}_f$ means only a small deviations are economical, but as it increases so too does the ‘freedom’ to change course and fly in formation, as greater savings are available.

Strongly linked to the level of deviation is proportion of flights with a common airport. The proportion of the assigned formations who share a common airport are plotted in Figure 5.20. While Singapore Airlines flights almost all fly to or from Singapore Changi Airport. EasyJet and Transatlantic flights move towards a greater proportion of shared airports as $\hat{\lambda}_f$ increases. As a result, with the number of shared



(a) Total Average Deviation Distance



(b) Total Average Deviation as a proportion of solo distance

Figure 5.19: Deviation in distance between solo and formation flight

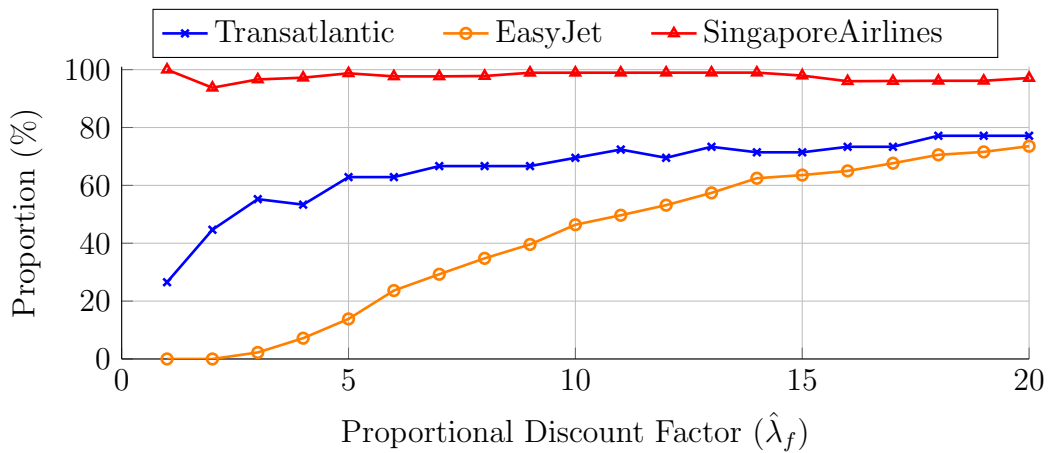


Figure 5.20: Proportion of formations who share a common airport

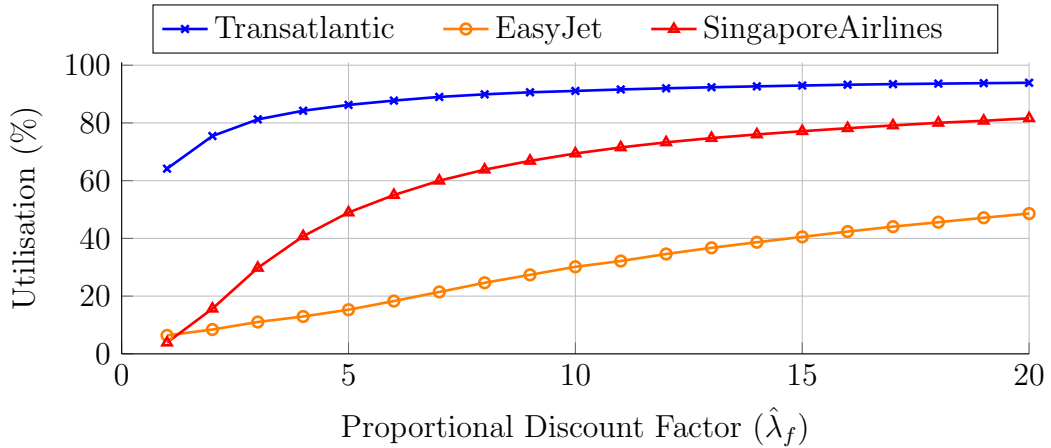


Figure 5.21: Utilisation

airports increasing and the level of deviation remaining more constant, the deviations must therefore be shifted to the joining or the breaking section of the flight. That is, formations taking off from the same airport likely have destinations which are further apart (and vice-versa)

5.6.4 Utilisation

Finally the level of utilisation depicts the efficiency of the overall formation process. The results plotted in Figure 5.21, between the three case studies, show a general tendency towards higher utilisation for higher levels of $\hat{\lambda}_f$. The Transatlantic flights quickly achieve very high utilisation levels. The Singapore Airlines require a $\hat{\lambda}_f$ of roughly 18% before they reach a level of 80% utilisation. While, EasyJet flights more gradually approach more modest levels of up to 50% for $\hat{\lambda}_f = 20$.

The results of this section show the effect of altering the formation discount factor $\hat{\lambda}_f$. While it is clear that greater discount factors generally result in higher overall savings, the relationship is not entirely one-to-one. While the value for $\hat{\lambda}_f$ has been estimated anywhere between 1 and 20% the results of this section show the direct result on formation savings and further that the overall routing and assignment methods can be used to assess these kinds of changes.

5.7 Summary

This chapter has explored the results of applying the methods outlined in Chapter 2 to three distinct case studies consisting of a list of solo flights. The Transatlantic

routes consisted entirely of long-haul flights flying east from the US to Europe, the EasyJet routes were based in Europe and either short or medium-haul, while the Singapore Airline routes ranged from medium to super-long-haul flights, all across south-east Asia.

Firstly a graph-theoretic approach was used to compare the network design of the flight lists. Where respectively the Transatlantic, EasyJet and Singapore Airline flights resembled bipartite, more-connected and hub-and-spoke networks. This along with a discussion on the geographic locations and typical flight lengths of the routes demonstrated the large difference apparent within commercial flight networks.

The results of running the routing and assignment of all formation combinations for each case study were then explored. Firstly for a scheduling restriction, placed on the total allowable difference in takeoff time required to meet in formation, gradually lifted showed how values tended towards the globally unconstrained optimum. Overall average percentage savings were promising for all the flight lists, with Transatlantic performing best at close to 9%, followed by Singapore airlines at just over 6%, while EasyJet flights produced a savings just under 2%, showing that there is real potential even for the short-haul flights.

A general measure of ‘suitability for formation flight’ called utilisation was also used, to give a real indication as to those flight lists which are much better suited for formation flight. Particularly high levels (95%) of utilisation were shown for the Transatlantic routes; slightly less (75%) for Singapore Airlines; and reasonably low levels (35%) for the EasyJet flights.

An analysis of the correlations between a number of key values showed what features were needed to produce ‘good’ formations. While for the Transatlantic flights low levels of deviation were important, generally indicating better savings. Conversely for the EasyJet and Singapore Airlines flights it was important to have a longer solo flight distance with the shorter flights performing worse than those which were longer. Interestingly it was shown that the scheduling constraint was entirely uncorrelated to formation saving, acting more or less like a random filter, removing potential choices within the assignment stage.

Finally the results for a range of possible formation discount factors between 1 and 20% was explored. Results showed that while the relationship between fuel saving and different values of $\hat{\lambda}_f$ was close, it is not strictly linear. Furthermore, while deviation was more incentivised for higher levels $\hat{\lambda}_f$, the average deviations did not greatly increase, instead formations shifted towards sharing airports and leaving all their deviations to one end of the formation route.

Chapter 6

Door-to-Door Routing For Road and Air

6.1 Introduction

One of the main goals set in 2011 by the Advisory Council for Aeronautics Research in Europe (ACARE) is for 90% of travellers within Europe to be able to complete their journey, door-to-door, within 4 hours [4]. The work of this chapter is to develop software capable of calculating door-to-door journeys, including air and road travel within Europe, with the aim of assessing how feasible reaching this 4 hour target is.

Given the current road and air transport network it is important to observe how it can be improved in order to best serve an ever growing population. Thus, for a particular start and finish location, and some cost to minimise such as time or distance, what does the full door-to-door journey look like. Additionally it is important to assess what aspects of the overall journeys should be altered, such as reducing check in/out times, to improve connectivity within Europe.

The following sections outline the work undertaken by the author during a six month Airbus placement between April and September 2013. The focus of the work is to build a framework to calculate the best door-to-door journeys between a road, airport and flight network. This chapter will begin by first outlining the data available and the data required, in Section 6.2. A description of the Routino routing software and how it can be used for this problem will then be explored in Section 6.3, with methods for first creating a road, airport and flight network discussed Section 6.3.2, where open-source routing software is adapted in order to route through the entire network. In Section 6.4 possible user-oriented workflows for generating results are

explored. The results of the European door-to-door journeys from six cities to the other 500 most populated cities is then presented in Section 6.5, where time ‘wasted’ at airports is shown to be a large factor in reaching the 4 hour ACARE targets.

Finally it is important to note that while scheduling is always important when undertaking any journey, the work of this chapter does not attempt to deal with any scheduling aspects of the driving and flying, that is, it is assumed that a flight departs a fixed time after the passenger arrives at the airport regardless of the time of day.

6.2 Developing a Road, Airport and Flight Network

In order to sufficiently understand how door-to-door routes work on a global scale, a vast amount of data is required. This involves sourcing and utilising a number of different sets of data into one unified network in order to calculate optimal routes. This section explores the range of relevant data needed, along with some of the methods required to adapt it for use within the door-to-door routing problem.

6.2.1 The Road Data

An ongoing open-source project called Open Street Map [108] enables us to freely access a large amount of the data necessary for the road network. The data is crowd-sourced, allowing for individual users to update and correct information for their geographical location. Thanks to a large online community of these dedicated ‘mappers’ there is a huge amount of information freely and easily available. The open-source nature of the project also helps to incentivize these people to develop a wide array of software and tools to help harness the power of this data. In particular a piece of software called Routino [109], has been used, with a large proportion of the work of this chapter is directed at adapting this software to allow for the calculation of door-to-door routes.

OpenStreetMap (OSM) data comes in a bespoke XML format, which is lightweight yet powerful. Every single point of data on an OSM map is comprised of a node, with a unique id, latitude and longitude for example

```
<node id="354" lat="-0.2204" lon="-0.5159"/>
```

This node could be part of road, a phone-box or even a windsock on an airstrip. In addition to the basic node information, tags are used, which can be used to add

details such as what the node represents, or other important attributes. The tag is comprised of a key k and value that key takes, v , for example

```
<tag k="barrier" v="bollard"/>
<tag k="junction" v="roundabout"/>
```

There are hundreds of different types of tags, all of which are well documented within the OSM wiki [108]. These nodes are then further classified into ‘relations’ and ‘ways’. A relation consists of one or more tags alongside an ordered list of one or more nodes or ways, their purpose is to define geographical or logical relationships between other elements. A way is defined by an ordered list of nodes, typically including at least one tag or inclusion within a relation. A way can comprise anything between 2 and 2000 nodes and can be open or closed loops. Ways are mainly things like roads and pathways, but can also be used to define an area, such as a park or grounds. For example a simple residential street called ‘Clipstone Street’ is represented as follows

```
<way id="3205">
  <nd ref="3155" />
  <nd ref="3162" />
  ...
  <nd ref="3222" />
  <tag k="highway" v="residential"/>
  <tag k="name" v="Clipstone Street"/>
</way>
```

where the ellipsis represents where a number of other nodes have been omitted for brevity. This way, is therefore simply a number of nodes, in order, representing points on the ‘residential highway’ known as ‘Clipstone Street’. Each included node, is defined elsewhere and can be easily looked up to find its details such as latitude and longitude location.

All the data Open Street Map holds is publicly available to download. It can be downloaded in a few different file types and for varying parts of the world. The largest file is ‘planet.osm’ which is more or less all the data available, it is around 40 GB compressed and over 550 GB uncompressed. Moreover there are update files published daily which contain any changes made so as to keep the data as up to date as possible.

6.2.2 The Flight and Airport Data

Flight data is readily available but in an entirely different format to the OSM data. Although lists of common flight routes can be easily sourced it is important to use accurate and up to date data. Therefore OAG data is chosen to represent all the flight routes for door-to-door routing. This data contains information on a huge list of flights, including times, dates and geographical routes flown. In total the flight data consists of 2,776 airports, with 12,839 direct flights between, roughly two thirds international and one third domestic. Additionally the only real data required for the airports is their geographic locations, which can be taken directly from the OAG datasets.

Although not as in depth or complex as the OSM data, this list of routes along with the corresponding airports is sufficient for creating the framework for door-to-door routing. In Section 6.3.2 this data will be converted into a format which is compatible with the OSM data, and thus suitable for the routing software of Section 6.3.

6.3 Routino Routing Software

Open source projects such as OSM have a tendency to encourage open-source community made software. OSM has a large range of tools and software built to try and make sense of such a vast amount of data. In particular there is a number of software packages designed for optimally routing, running on a variety of platforms and often tailored to tackling a particular problem. The software which is the focus of this work is one particular project called Routino.

6.3.1 Routino Base-Software

Routino is an application written primarily in C, designed to route between two locations using a dataset of topological information contained within OSM. The router uses a network routing algorithm, taking OSM data as input and calculating either the shortest or quickest route between two (or more) points. At the core of this is a pre-processing stage which attempts to create a data structure of ‘cleaned’ data along with a number of ‘super-nodes’ to create a two-tiered routing process in order to route more efficiently. This process is necessary because even short roads are made up of a huge number of nodes and so without adequate ‘cleaning’ the routing would be extremely time consuming.

To find the route within the super-node network an A^* algorithm is used [110].

This uses a score-based calculation to find the route between super-nodes with the lowest score, which corresponds to either the shortest or fastest path. The score for the remaining path, between the super-nodes and the destination is then calculated using Dijkstra algorithm [111] and these results are combined, finding the route with the best score to create the final route.

6.3.1.1 Routing Profiles

One of the key features of Routino is the ability to route for a number of different transport types such as car, motorcycle, HGV, horse, foot and even wheelchair. It is able to do this using the notion of a profile. Each transport type has an individual profile which defines both speed and preference for every type of highway. Part of an example profile for a car is as follows:

```
<speed highway="motorway" kph="112" />
<speed highway="residential" kph="48" />
<speed highway="path" kph="0" />
<speed highway="steps" kph="0" />
```

So a car can travel 112 km/h on a motorway, 48 km/h on a residential road, but not at all on a path or on steps. Conversely, if traveling by foot, one can reach a modest 4 km/h on a path or steps but are not allowed on a motorway. In a similar way preferences are defined, in an attempt to order a passengers ‘desire’ to take a certain type of highway. For example for a cyclist:

```
<preference highway="cycleway" percent="100" />
<preference highway="path" percent="90" />
<preference highway="steps" percent="0" />
```

The notion of percent here adjusts the ‘score’ of a particular route and means that the additional score for every 1 km of travelling on a path is the same as that achieved by travelling $1 \times 100/90 = 1.11$ km on a cycleway, thus one would be happy to travel a little farther on a cycleway if it meant avoiding a path. Furthermore there is an option to allow the forcing of one-way roads, cars must obey one-way, but someone on foot would not. The real bonus to having profiles is that these are not ‘hard-coded’ and can be changed rapidly, as an input to the software. Therefore you can quickly get a number of different routes, for example one that tries to avoid motorways or one that opts for shortest rather than quickest. Route changes may then be observed by perturbing certain inputs, such as how a route differs if the speed on secondary roads was increased.

6.3.2 Adapting Routino for Road and Air

The Routino software is an ideal solution for routing for a global road network, but it first must be adapted to account for the ability to take flights. An initial exploration of possible separate routing methods indicated that a centralized approach was needed. Thus the idea is to build upon the Routino software to allow calculation of routes which include the choice of flights and road simultaneously. With Routino taking the OSM XML data as input, it is therefore necessary to create the airport and flight network in the same way. While the flight network itself will be similar to the road network, the airport network acts more like ‘barrier’ between the road and air.

The XML data scheme discussed in Section 6.2.1 provides an insight into the format the input data for Routino must to be constructed. Therefore the entire network needs to be defined as a series of either nodes, ways or relations. For the road network these are already defined in the OSM file and so do not need any adjustments. The airport and flight network must be converted into these types. For example one can imagine for a flight, the ‘way’ is the great circle flight path, made up by a number of ‘nodes’ placed along the path. With this in mind the following sections outline the construction of this data formation for use within Routino.

6.3.2.1 Design Considerations

Having the geographical location of a flight route is only part of the process. You cannot just arrive at an airport and take off straight away, flights also have a number of time-penalties and restrictions inherent to them. So it is important to not only account for a variety of these penalties and constraints but also be able to adjust them on the fly to see how they may effect how we route. Therefore the following need to be taken into account:

- Flight routes (departure and destination airport);
- Flight types (short, medium, long haul);
- Flight speeds for each flight type;
- Climb profile and associated speed;
- Descent profile and associated speed;
- Check in/out times both domestic and international;
- Taxi in/out times;

- Transfer times between combinations of domestic and international flights;

The above need full consideration to more accurately model the true way a road and flight journey interact. To accomplish this we now create an airport network that can facilitate not only different types of flow through it but also incorporate the times required to carry out each particular flow. That is a ‘path’ where you are forced to flow in the correct way, for example coming from the road network → check in → taxi out → climb → cruise → descend → taxi in → check out → join back up with the road network. Or perhaps instead of rejoining the road network you descend → taxi in → transfer → taxi out → climb etc. It is very important to make sure that the way people can flow through this network is constrained in order to adequately model the route.

6.3.2.2 New Highway Types

In compliance with the OSM and Routino scheme we create a set of new highway types to go alongside the existing ones. Each of these new highway types will have a default speed and preference assigned to it for each profile. The new highway types for the flight network, outlined in Table 6.3, include short/medium/long haul flights, which are defined by Table 6.1, with the three main phases of climbing, cruising and descending. The airport network, which will be covered in greater detail in Section 6.3.2.3, is less like the typical road network and is really used to artificially impart time penalties when switching from the road network to the flight network. Therefore the highway types shown in Table 6.2 represent all the major time considerations when transitioning from road to air. This includes checking in and out, taxiing and transferring between flights.

Table 6.1: Definition of flight types

Flight Type	Distance
Short	Less than 1200 km
Medium	Between 1200 and 4000 km
Long	Greater than 4000 km

6.3.2.3 Airport Network Design

For each of the 2276 airports, contained in the OAG data, an artificial airport layout must be created. With one road path in, the entrance, and a number of flight paths in and out, either international or domestic. With the new highway types defined,

Table 6.2: Flight network highway types

Highway Name	Description	Example Speed
Flight Short: Climb	Takeoff and climb	400 km/h
Flight Short: Cruise	Cruise	550 km/h
Flight Short: Descend	Descend and land	350 km/h
Flight Medium: Climb	Takeoff and climb	450 km/h
Flight Medium: Cruise	Cruise	650 km/h
Flight Medium: Descend	Descend and land	400 km/h
Flight Long: Climb	Takeoff and climb	500 km/h
Flight Long: Cruise	Cruise	750 km/h
Flight Long: Descend	Descend and land	500 km/h

Table 6.3: Airport network highway types

Highway Name	Description	Example Time
Check In	Overhead to get to check in	15 min
Check In: Dom.	Bag check/security	60 min
Check In: Int.	Bag check/security	120 min
Check Out: Dom.	Collect bag/security	30 min
Check Out: Int.	Collect bag/security	60 min
Taxi Out	Gate to runway	15 min
Taxi In	Gate to runway	15 min
Transfer: Int. → Int.	Make a connection	30 min
Transfer: Int. → Dom.	Make a connection	60 min
Transfer: Dom. → Int.	Make a connection	60 min
Transfer: Dom. → Dom.	Make a connection	30 min

we simply construct an airport layout which incorporates all these types creating the realistic flow through the airport.

A diagram representing how passengers should flow within an airport is shown in Figure 6.1. The arrows represent the way passengers are permitted to travel around the airport. The airport is divided by the domestic and international flights, connected by possible transfers, with arrival flows in and departure flows out.

This creates a minimal, yet robust method for routing through an airport. Each dot represents a node, while each arc represents a way. Importantly, each arc section of the graph has a predefined distance, so, given a required duration to traverse each arc, the corresponding speed can also be calculated. For example if the taxi out arc is 100 meters (0.1 km) long and we require it to take 15 minutes (0.25 hours), to simulate this, we set the speed across the taxi out arc to be $0.1/0.25 = 0.4$ km/h. Therefore, given the arc distances, the times outlined in Table 6.3 are translated into speeds for use within the routing software.

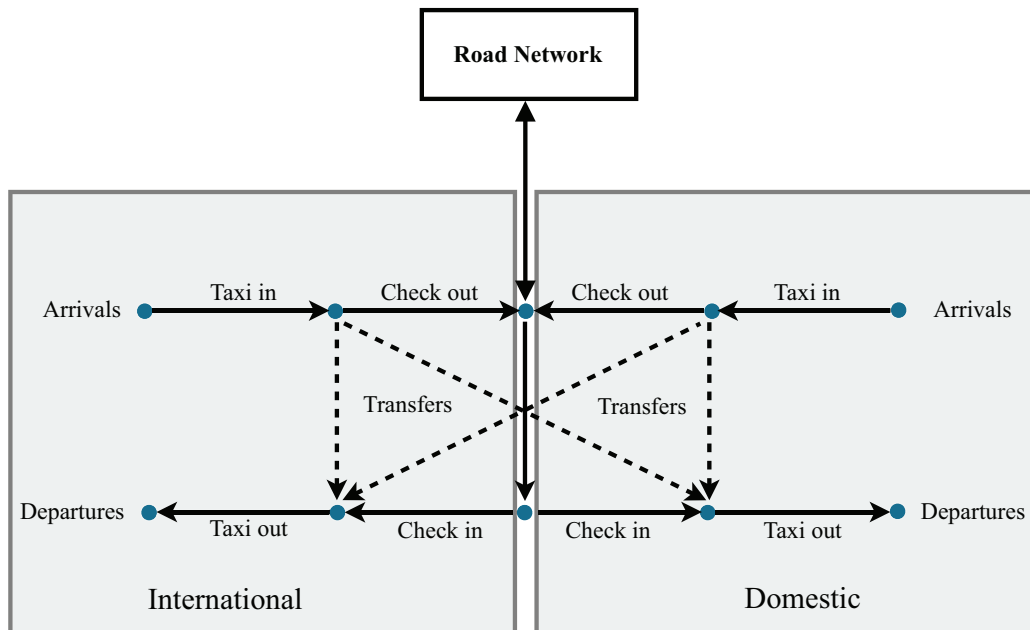


Figure 6.1: Airport network and possible flows

Once all the airports have been created, the flight routes between them can also be added between airports' arrival and departure nodes of Figure 6.1. While these values can be adjusted, for use within this chapter we define the climb distance to be 200 km and descent distance to be 250 km. For flights which are less than 450 km we define the climb to be 40%, the cruise 30% and the descent 30% of the total great circle distance.

6.3.2.4 Combining Road and Air Data

With the general methods in place for creating airports and flight routes, they can now be combined so they can be utilised within the routing software. A Matlab script is used to loop through the OAG dataset and create the appropriate XML file containing all the airports and flight routes in the OSM format. This script assigns unique ids to the new nodes it creates and joins them together into ways. The node placement is important, as the distance between arcs must be consistent across the entire globe. Therefore an inverse distance function is used to determine the appropriate longitude and latitude point of every new airport node based on the location of the last.

Flight routes are then created by creating a selection of sample points along the great circle route as the flight's nodes. Although many flights will not directly fly great circle, for the purposes of this chapter it acts as a reasonable estimate. The method does, however, leave room in future to create and calculate actual flight paths.

The last part of combining the data involves joining the two distinct data files together. Not simply just computationally, as we must also physically join these airports to the road network so the router can go between them. Therefore it is necessary to lookup, for each airport, and connect it to the nearest accessible road network. A web service called `overpass-api` [112] is sufficient, which allows simple queries to collect specific OSM data from the entire dataset. Importantly we require the node id of a nearby major road which is adequately connected to the main network (as some residential and service roads are often used within airport boundaries but do not connect with the outside road network). This process is roughly as follows:

1. Input: location of the airport
2. Create a bounding box around the airport;
3. Call `overpass-api` for the major roads in that box;
4. Remove unnecessary data;
5. Find the closest road node;
6. Return node id

The returned node id is then assigned to a new ‘airport connection’ way and then the road, airport and flight network are all connected and we can successfully route from the road into the airport and beyond. The resulting network data is then three separate OSM files, the road data, the airport data and the flight data. These files will be used within the Routino software in Section 6.3.3, enabling door-to-door routing for road and air.

6.3.3 The Router

The road and flight data has now been created into a useable and connected OSM format, for use within the Routino routing software. However the routing software needs one more pre-processing step to create a more efficient network for the routing method to work with.

6.3.3.1 PlanetSplitter

An executable called ‘PlanetSplitter’ is included with the Routino software, this is used to parse the ‘raw’ OSM data into a cleaned and structured format for the router

to understand and efficiently make use of. This is run, calling in all the required OSM files as inputs where it is parsed and processed into a final data structure. For example, running this for the road data for Europe and the flight network is done as follows:

```
./planetsplitter --parse-only '/Airport_Network.osm'
./planetsplitter --parse-only --append '/Flight_Network.osm'
./planetsplitter --parse-only --append '/Europe.osm.pbf'
./planetsplitter --process-only
```

This then saves the parsed and processed OSM files into a folder consisting of four files: Nodes, Relations, Segments and Ways. These are structured and will be called when using the router. This process also includes the calculation of the ‘super-nodes’ for use within the two-tiered routing of A^* followed by Dijkstra’s shortest path algorithm.

6.3.3.2 Routing

The main routing program performs the actual calculation of optimal routes, using the structured data generated by the PlanetSplitter program. It can take a number of different inputs but essentially it takes the input of two (or more) locations and the profile and returns the optimal route. There are a wide range of inputs, in addition to the processed OSM data, that can be supplied to the routing program, with a selection of them presented in Table 6.4. These can be anything from making adjustments to the profile and preferences, setting the language for the step-by-step instructions, or setting the output files to create for the finished route.

Table 6.4: Example of Routino input flags

Input Command	Description
profiles={filename}	The routing profile file.
translations={filename}	Translations file for directions
output-html	Create HTML output
output-kml-route	Create KML output
output-csv-route	Create CSV output
output-text	Create text output summary
output-text-all	Create text output of all nodes
transport={transport}	HGV, car, foot, etc
shortest	Find shortest path
quickest	Find quickest path
lon{i}={longitude}	longitude of point i
lat{i}={latitude}	Latitude of point i
highway-{highway}={preference}	Set highway preference
speed-{highway}={speed}	Set highway speed

The router can take an input of a number of locations, that is the longitude and latitude of the waypoints that make up the start, middle and end points of the route. Up to 99 waypoints can be specified and the route will pass through each of the specified points in sequence. For each point given, the router will find the closest node or point within a segment that allows the specified traffic type.

Then given the choice of either shortest route, *i.e.* the minimum distance route, or the quickest route, *i.e.* minimum time route, the router can be run. On completion of finding the best route the router can output this result into a number of useful file formats such as HTML, KML, CSV or plain text.

6.3.3.3 Example Usage

Given the variety of inputs defined by Section 6.3.3.2, the following describes a number of scenarios where the inputs to the router may vary. Many of the possible inputs have either a default value or are specified in a profile file unless otherwise stated.

Example 1

A car journey between Bristol and Glasgow, that takes a scenic route (*i.e.* avoids major roads) and only travels at a reasonably slow speed of 80 km/h.

```
router --transport=motorcar --quickest \  
  --lat1=51.48 --lon1=-2.57 --lat2=55.86 --lon2=-4.23 \  
  --highway-motorway=0 --highway-trunk=0 \  
  --speed-primary=80 --speed-secondary=80
```

Example 2

A journey between Bristol and Glasgow, with a very low 20% preference for driving

```
router --transport=motorcar --quickest \  
  --lat1=51.48 --lon1=-2.57 --lat2=55.86 --lon2=-4.23 \  
  --highway-motorway=20 --highway-trunk=20 --highway-primary=20 \  
  --speed-primary=80 --speed-secondary=80
```

Example 3

A journey between Bristol and Bucharest which avoids medium / long haul flights

```
router --transport=motorcar --quickest \  
      --lat1=51.48 --lon1=-2.57 --lat2=44.42 --lon2=26.091702 \  
      --highway-flight_medium_climb=0 --highway-flight_long_climb=0 \  
      --highway-flight_medium_cruise=0 --highway-flight_long_cruise=0 \  
      --highway-flight_medium_desc=0 --highway-flight_long_desc=0 \  

```

6.3.4 Final Route Solution

As described in Section 6.3.3.2, Routino is capable of producing a range of output files. Most of these are simply the main result's data, *i.e.* the route, but printed in a particular format. For example the CSV file is just the text-all file but separated by commas. While the size of the files depend greatly on the complexity of the route, driving routes have many more twists and turns and so need more location data. Airport movements will be only 10 or 11 different points. A sample of the results of routing from Bristol to Glasgow are shown in Section 6.3.4, whereby the section detailed is where we move from the road network to the airport and flight network. The following describes some of the output types that the result can be easily presented as.

Comma Separated Values (CSV)

The Comma Separated Value file is probably the most well known text structures and is therefore easily transferable. It is widely supported and can be used by a variety of different programs for analysis and further work.

Text

The text files, are essentially the CSV file but separated by tabs rather than commas. The Text-all file contains more points and information than just the normal text file. These include the node id, bearing and the type of node (intersections, junction etc). This is more useful as an output for use within more-verbal routing, given specific instructions to follow to carry out the route.

HTML

This again contains the same information as the text and CSV file, and uses bearing and junction data to give actual directions. It can work out which way the route is turning and therefore describe movements such as 'turn left' or 'go

Table 6.5: Sample of output for Bristol to Glasgow

Longitude	Latitude	Distance (km)	Duration (m)	Cumulative distance (km)	Cumulative duration (m)	Speed (km)	Description
-2.721	51.390	0.0440	0.0316	13.7040	9.5750	80	Downside Road
-2.722	51.390	0.0580	0.0433	13.7620	9.6183	80	Downside Road
-2.723	51.385	0.5370	0.5033	14.2990	10.1216	64	Cooks Bridle Path
-2.724	51.384	0.1100	0.1016	14.4090	10.2233	64	Winters Lane
-2.719	51.382	0.4180	0.2850	14.8270	10.5083	88	BRS: Airport connection
-2.718	51.382	0.0100	16.666	14.8370	27.1750	0.03600	BRS: Check in
-2.718	51.382	0.0100	60.000	14.8470	87.1750	0.01000	BRS: Check in (domestic flight)
-2.718	51.382	0.0090	15.000	14.8560	102.1750	0.03600	BRS: Taxi out
-3.518	53.599	252.5910	37.888	267.447	140.063	400	Domestic short haul flight climb: [BRS:GLA]
-3.711	54.099	57.152000	6.2333	324.5990	146.2966	550	Domestic short haul flight: [BRS:GLA]
-4.433	55.872	202.595	34.730	527.194	181.0266	350	Domestic short haul flight descent: [BRS:GLA]
-4.433	55.872	0.0100	16.666	527.20400	197.6933	0.036	GLA: Taxi in
-4.433	55.871	0.0090	13.500	527.21300	211.1933	0.040	GLA: Check out (domestic flight)
-4.435	55.874	0.3650	0.2483	527.57800	211.4416	88	GLA: Airport connection
-4.441	55.873	0.3660	0.3416	527.94400	211.7833	64	Walkinshaw Road

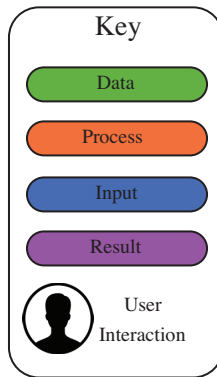


Figure 6.2: Workflow key

straight’, much like more modern satellite navigation services. Additionally the Routino software comes with a configurable web ‘front end’, so you can visualise the route on a map with directions alongside.

Keyhole Markup Language (KML)

The original software created a GPX file (common among GPS tracking) but was changed to the KML used within Google Earth software. It creates a file which can be opened within Google Earth/Maps and shows the route plotted over a map over the associated time range.

6.4 Workflow

With all the data and methods in place the aim of this section is to outline the different workflows to go from a user input to some meaningful output. Figure 6.2 describes the ‘color key’ used throughout the workflow diagrams. Green represents data, such as a list of locations or OSM file; orange is a process, such as a matlab script or running Routino; blue corresponds to an input, such as a routing command; purple corresponds to a meaningful output, such as a KML file; and finally the user icon corresponds to a user interaction, such as typing in a input or observing an output. Thus the following sections outline the four key stages of Figure 6.3.

6.4.1 Stage 1 - Precomputing

The first stage, depicted in Figure 6.4, involves the gathering and parsing of data, as described in detail in Section 6.3.2, with the output being used within stage 2 (Section 6.4.2), the routing process. This stage is important, as the quality of the

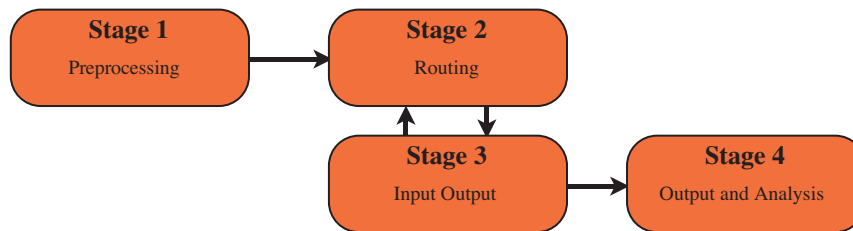


Figure 6.3: Overview of workflow

inputs and way they are processed impacts greatly on the quality (and accuracy) of the routing result.

The time needed to parse the OSM inputs in PlanetSplitter (Section 6.3.3.2) is proportional to the size of OSM files. Processing Europe, for example, takes a few hours, whereas parsing the Planet.osm take a few days. The Airport.osm file results in a relatively small size of 150 MB, taking only a few minutes to process. Therefore stage 1 is the most continually computationally-expensive part of the door-to-door routing process.

Once stage 1 is complete, it does not have to be repeated and so is a sort of sunk cost of the computational efficiency. Having said that, one large shortfall of this stage is that, presently, if you need to change the network, *i.e.* add/remove a new flight or airport or update the road network, you need to reprocess the entire OSM network. However this would likely be something that could be addressed in the future versions of the software.

6.4.2 Stage 2 - Routing

The second stage, as shown in Figure 6.4, involves the actual calculation of routes. Using the data structures from stage 1 along with a number of inputs, produces output files describing the optimal route. This stage encompasses the main functionality of the adapted Routino software. The Router application is just C code compiled to an executable and is therefore fairly low level. The router is run from the command line along with a number of inputs, as described in Section 6.3.3.2, to define the variables and constraints on the route that is calculated. It is therefore beneficial to automate stage 2, or at least abstract it from the user. The third stage is a collection of ways of achieving this, to gain results quickly without the need to type lots of long commands into the command line.

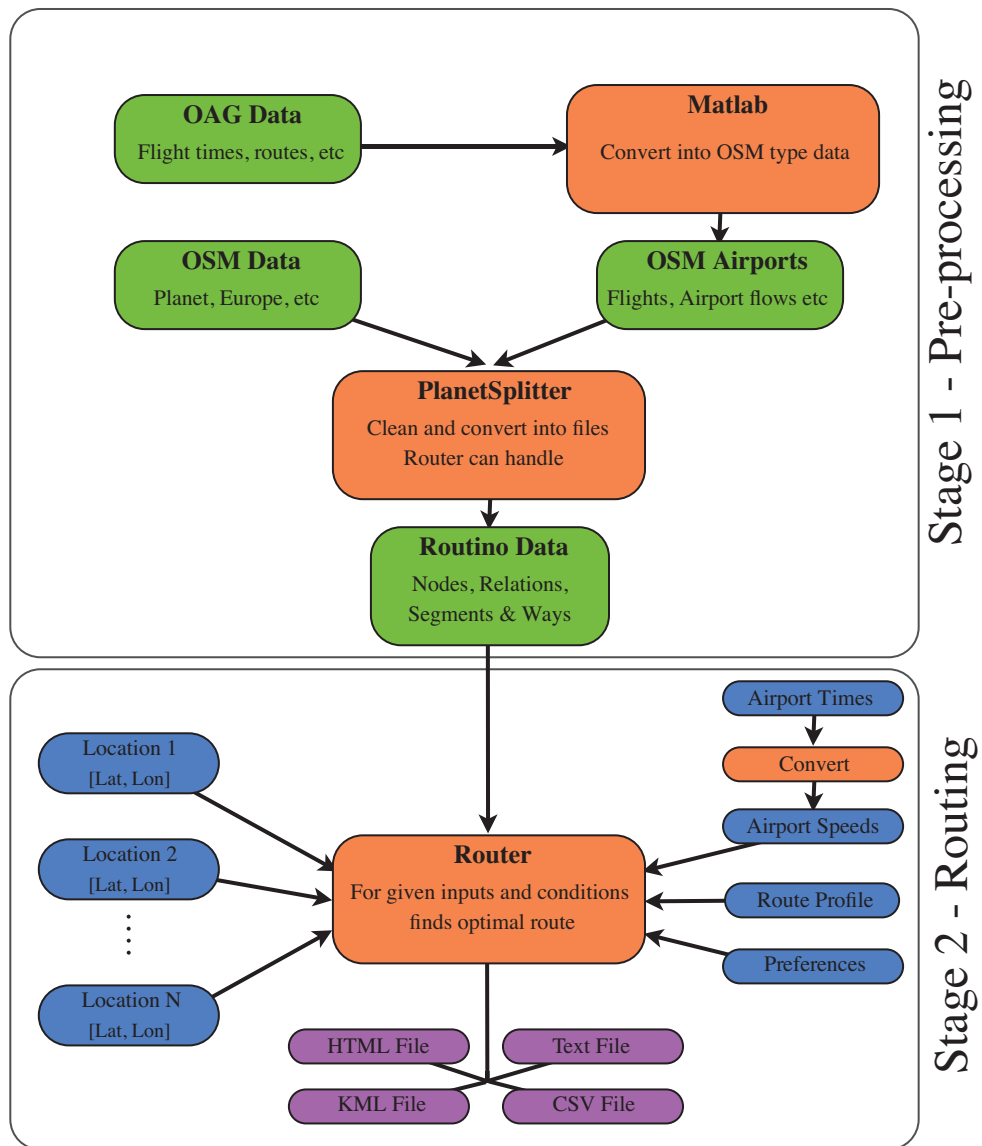


Figure 6.4: Pre-computing and routing stages

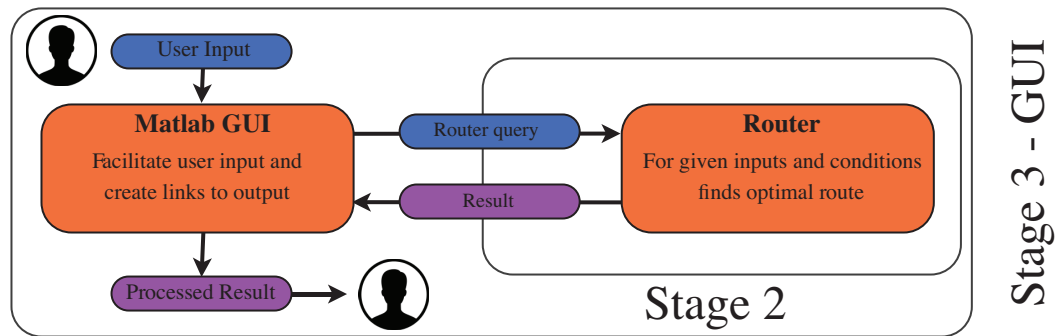


Figure 6.5: Stage 3 - Using a Graphical User Interface

6.4.3 Stage 3 - Input Output

The third stage is essentially an addition to stage 2. Due to the command line interface, stage 2 is ideal for use within a wrapper such a Graphical User Interface (GUI) or to be called upon from other programs for example Matlab or even Excel (through Visual Basic). This makes it very versatile and can be used to find very specific single routes or can be looped through hundreds of routes. Stage 3 is therefore designed to make it easier to perform stage 2 and gain results. The rest of this chapter chooses to use Matlab to call the Router and perform post-processes. It has been chosen for its usability and availability along with its ability to handle a large range of data formats (both input and output).

6.4.3.1 Stage 3 (a) - GUI

A GUI simplifies the calculation of inputs for the user and can be used to easily observe results and open output files directly. Matlab allows for fast prototyping of GUIs so can be easily changed to suit various needs, the current GUI is shown in Figure 6.6. It can be used to hide certain processes from the user as outlined in Figure 6.5. For example airport times cannot directly be inputted to the Router, instead the values are converted to speeds which represent the required time. Similarly an overall preference for Car or Aircraft can be used to quickly change preferences for all the highways associated with each transport type. Furthermore the GUI allows the input of an address with a geographical lookup, so the user does not need to know the exact longitude and latitude values. In future it would also be possible to have input from a map, clicking a departure location followed by a destination, with the resulting route plotted.

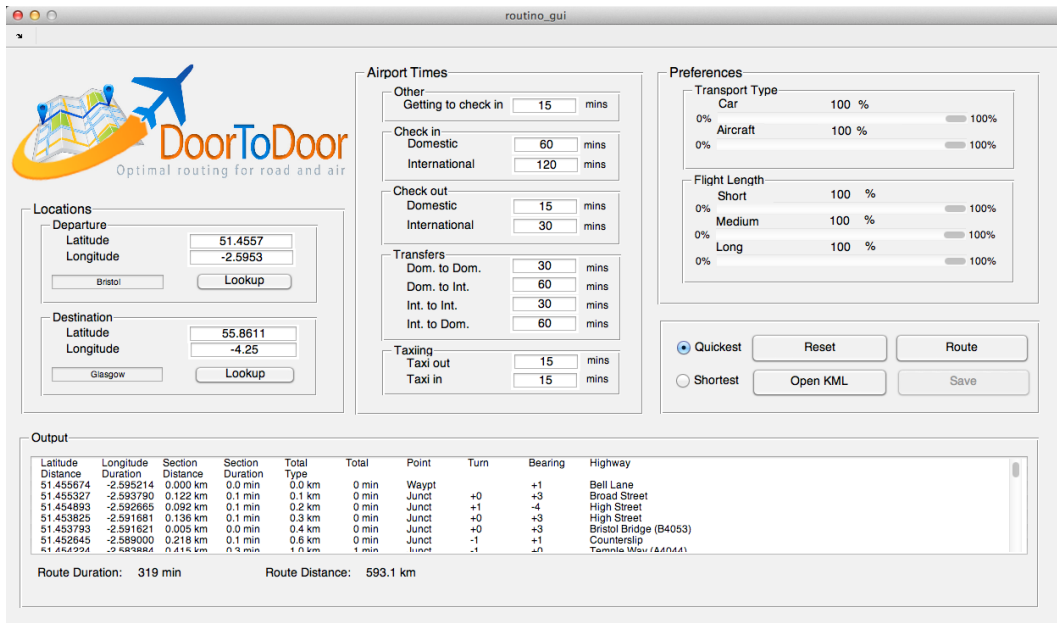


Figure 6.6: Screenshot of Matlab GUI for Door-to-Door routing

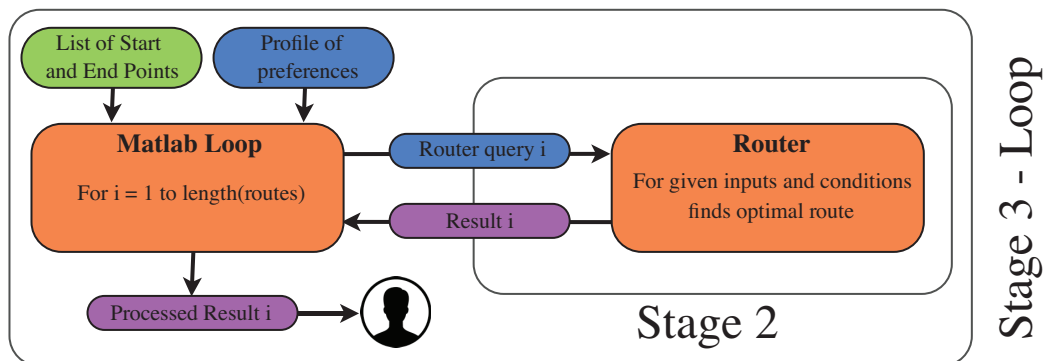


Figure 6.7: Stage 3 - Using a loop

6.4.3.2 Stage 3 (b) - Automation

For a large number of required routing tasks a more automated technique can be used. This enables the calculation of a large amount of results without having to create each router-query individually. For example a list of start and end points with a given profile, including speeds, preferences and airport time, can be used as input and would create a series of CSV and KML files for each optimal route as shown in Figure 6.7 as output. Moreover the output from the loop can be input directly into further Matlab processes to analyze the large set of results.

The use of this automated type system is incorporated into creating visual data in stage 4. The aim of the results of Section 6.5 is to take major cities, such as London,

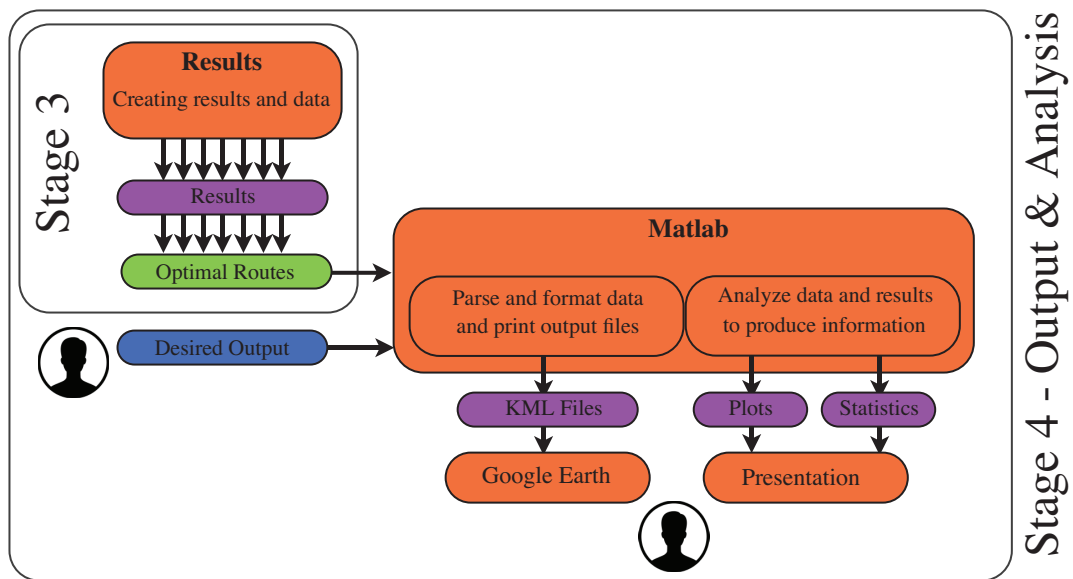


Figure 6.8: Stage 4 - Output and Analysis of results

and loop through a list of other major cities to see how long it takes to reach them. Running this for a number of different cities, gradually builds up a picture of the connectivity and ability to reach large portions of the population.

6.4.4 Stage 4 - Output and Analysis

The main reason for implementing the methods of this chapter are to facilitate the gathering of large amounts of data and sufficiently analyze and present the results. The optimal routes contain a large amount of data points, the challenge is translating this into information. The route results contain a number of interesting values such as distance and time, while additional city information can include factors such as GDP and population that might be of interest to the end user.

Creating useful output information is therefore a priority of the door-to-door routing software, it is stage 4 which takes the results from the previous three stages and produces output suitable for an end user to both visualize and interrogate, as presented in Figure 6.8. Google earth is a very powerful tool for visualizing aspects which arise for temporal and geographical data, so a KML can be created which animates a route's path over time. Many routes can be grouped together to see how a number of different routes propagate over time. These routes can also hold textual information that can be used to display progress and statistics, such as current and average speeds or population reached and additional city information can then be overlaid. However the scope of this chapter does not include detail on most of these output types as

these are mainly intended for an end user.

This completes the outline of the door-to-door routing software and how it used. With the workflows now discussed the methods of this chapter will now be used to gain some insight into the door-to-door journeys within Europe.

6.5 Results

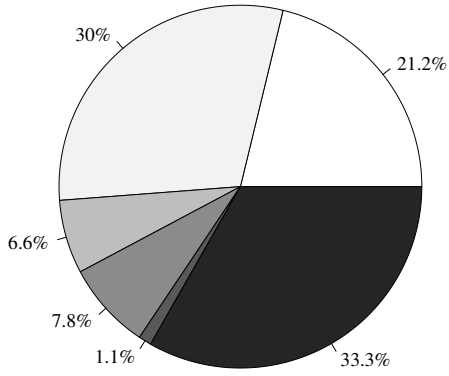
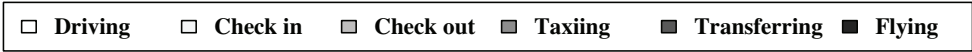
For any location in European, we can now use the door-to-door routing software to calculate the quickest routes to reach any other location. With that in mind, for a given starting point of a city we calculate the door-to-door journey to reach 500 of the most populated European cities, which has been automated via the methods of Section 6.4.3. The remainder of this chapter will focus on presenting and discussing some of the results of the door-to-door journeys starting from the six most populated cities in Europe: London; Berlin; Madrid; Kiev; Paris; and Rome.

While routing between one starting city to 500 others generates a huge amount of results, the aim is to assess the overall patterns rather than the specifics of individual routes. Although it is of interest to look at how individual routes can be improved, it is beyond the scope of this chapter. Instead we intend to assess the features of a ‘good’ journey and how one might considering improving journeys within Europe.

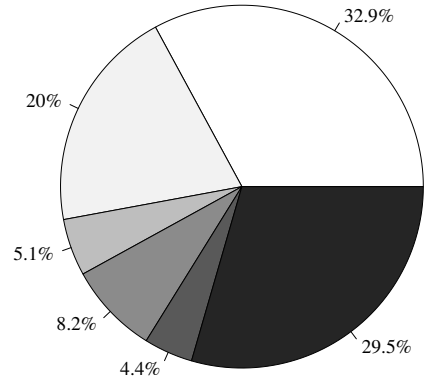
6.5.1 Where Total Journey Time is Spent

First we consider where, on average, the majority of time within a door-to-door journey is spent. Looking at the pie charts shown in Figure 6.9, we can quickly see, proportionally, for each of the six cities, how the time of all 500 journeys is spent between driving, checking in, checking out, taxiing, transferring and flying.

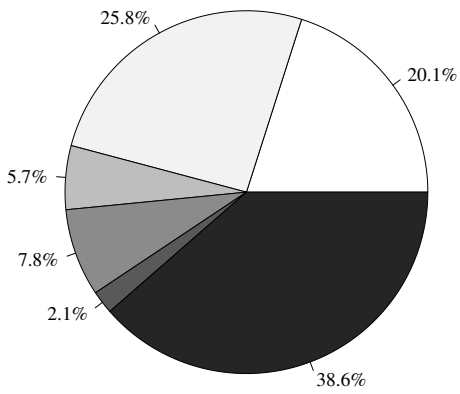
While each of the six cities present fairly similar results, there are observable differences. For example the proportion of driving is higher for Berlin and Paris (and arguably Rome) than for the other cities. This is likely due to their geography, being ‘inland’ and more central means that the 500 destination cities are more easily accessible by car. Moreover the box plot diagram of Figure 6.10 shows the spread of the great circle distances (*i.e.* as the crow flies) to the other cities. The red center line is the mean value, while the dotted black line corresponds to the mean plus or minus two standard deviations (as in Section 5.5.1). The average great circle distance from Berlin and Paris to the other 500 cities are lowest at just over 1000 km and are perhaps therefore better suited to reaching other parts of Europe.



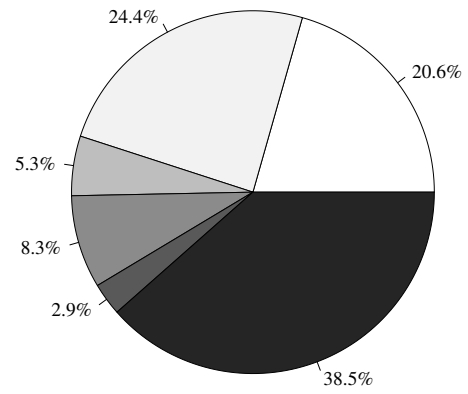
(a) London



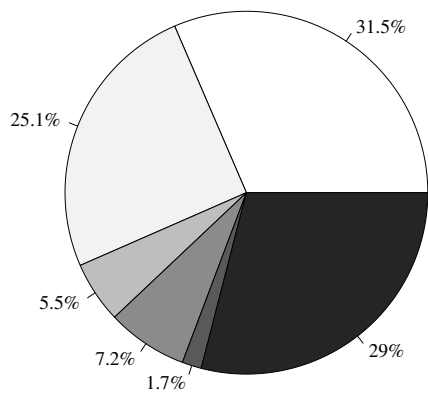
(b) Berlin



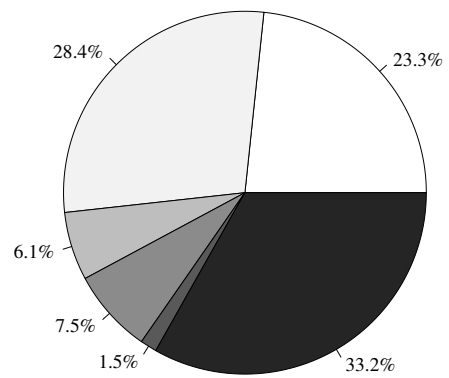
(c) Madrid



(d) Kiev



(e) Paris



(f) Rome

Figure 6.9: City connectivity results

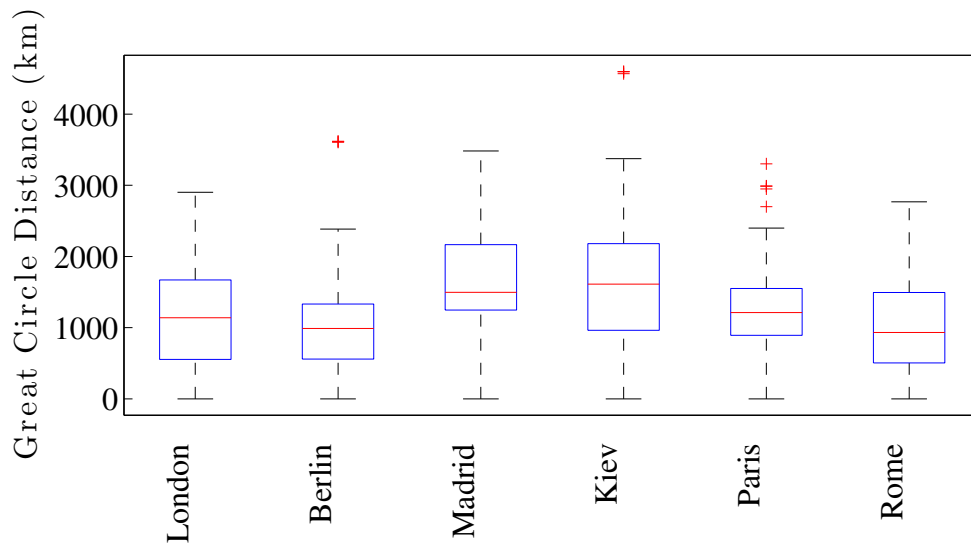


Figure 6.10: Spread of Great Circle Distance to other 500 European Cities

Finally what is clear for all of the cities is that a large proportion of total journey-time is spent ‘waiting’, *i.e.* stationary, whilst checking in and out, transferring and taxiing. Spending journey-time waiting, although necessary for taking a flight, is essentially a waste for the passenger. On average the proportion of time wasted ranges from 37% for Paris, to 43% for Rome. Therefore one of obvious ways of improving connectivity within Europe, is to reduce the amount of time spent waiting.

6.5.2 Journey Distance Compared to Journey Duration

In order to further compare the differences between the six cities, we now look at the spread of values for all the door-to-door routes. The scatter plots of Figure 6.11 show how the distance of the route travelled corresponds to the total duration of the journey.

Looking first at the shorter routes, *i.e.* distances less than about 500 km, what can be seen is that for all six cities there is a roughly-linear relationship between distance and duration. Now this is essentially all the short routes, which are too short to warrant a flight and so are thus driving-only routes. The gradient of the slope corresponds to the average speed, with a shallower slope meaning a higher average speed and a steeper slope meaning a lower average speed. As we move towards the longer routes then the results start to spread out more. As these routes include a flight, the average speed increases as the total distance increases, as more of the journey is spent at faster speeds.

As most flights tend to depart from the nearest possible airport, the spread on the

longer routes is likely due to the varied amount of additional driving required after the flight. Furthermore, if the waiting times at airports can be reduced then the more spread out points of the flight routes would shift towards a lower duration and would likely encourage some of the longer, driving-only routes to opt for flying.

6.5.3 Proportion of Wasted Time

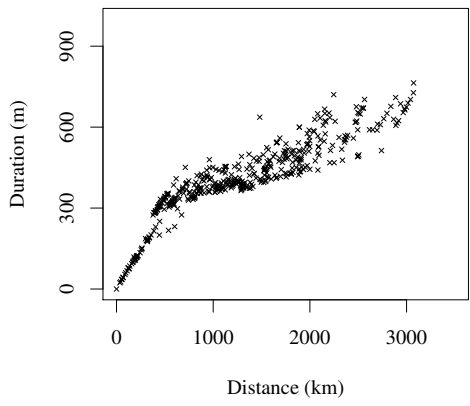
As discussed in Section 6.5.1 a large amount of the journeys spend a reasonable amount of their time waiting at an airport. The plots of Figure 6.12 show the spread of journey distance against the percentage of the journey spent waiting. Again there are a number of shorter routes which are driving-only and this results in essentially no time wasted. As the journey distance increases, and the routes begin to encounter airports then the proportion of the entire journey spent waiting grows dramatically. Those first few routes that warrant a flight can spend anywhere up to about 75% of the entire journey duration waiting. While, as would be expected, as the journey distance increases, proportionally less time is spent waiting, however it mostly remains accountable for between 25 and 50% of the total journey.

Therefore it is clear that one major bottleneck within the European transport network is the amount of time spent at airports. Reducing the wait times at airports will be crucial in reducing overall journey times and improving passenger efficiency. One can imagine that with reduced wait times more journeys would opt for flights and the average journey speed would increase.

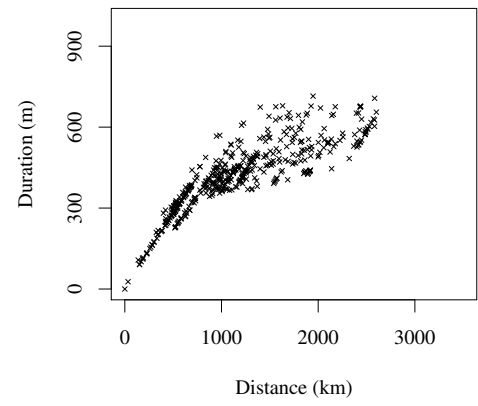
6.5.4 Isochrone Map

Finally, in line with the ACARE targets [4] of passengers being able to reach their destinations, door-to-door, within 4 hours we look at the results for the current situation. Isochrones are essentially plots, from a given starting point, representing where one can reach within a fixed amount of time. Isochrones for the six most populated European cities are depicted in Figure 6.13. A collection of points is first extracted from the 500 routes, which are the last points reached within a fixed time-frame. For an increasing time-frame, the isochrone regions can be estimated by calculating the convex hull of this collection of points. Each bordered region of Figure 6.13 represent a separation of one hours worth of journey-time, up until to a maximum of 6 hours.

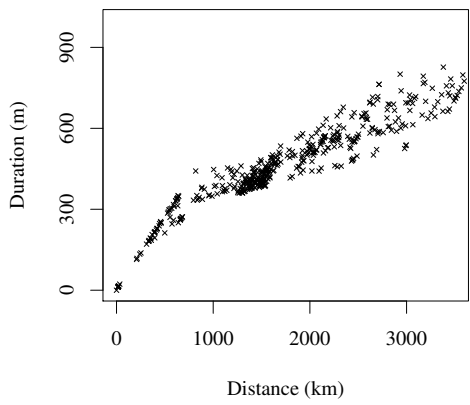
Due to the direction of many of the routes some isochrone regions are somewhat undersampled and result in large straight borders. Additionally the conical shape of the London, Madrid and Kiev isochrones are partly due to them being in a ‘corner’ of



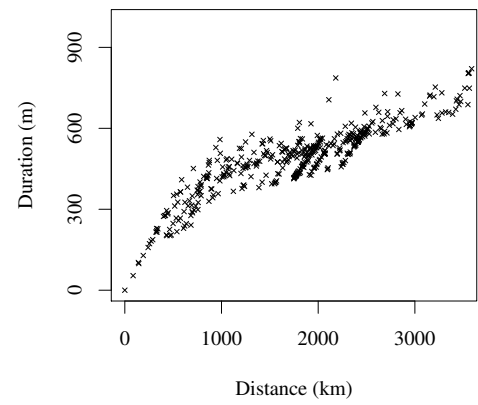
(a) London



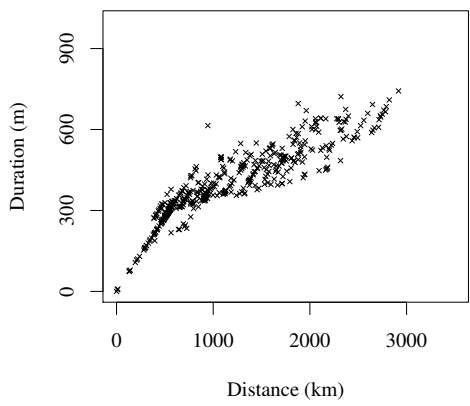
(b) Berlin



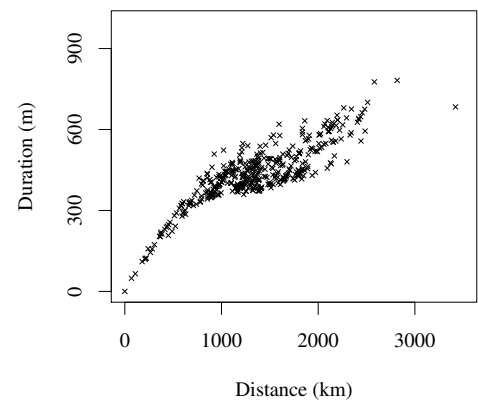
(c) Madrid



(d) Kiev

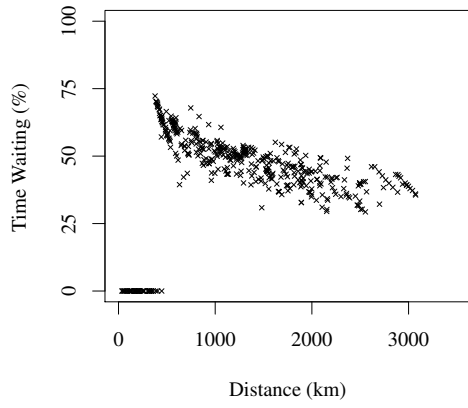


(e) Paris

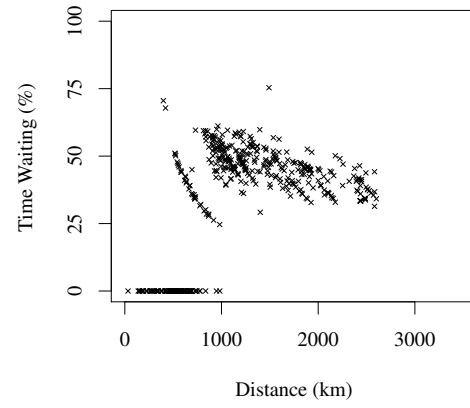


(f) Rome

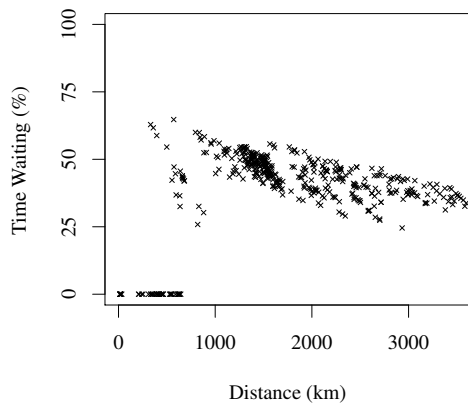
Figure 6.11: Duration vs distance of journey



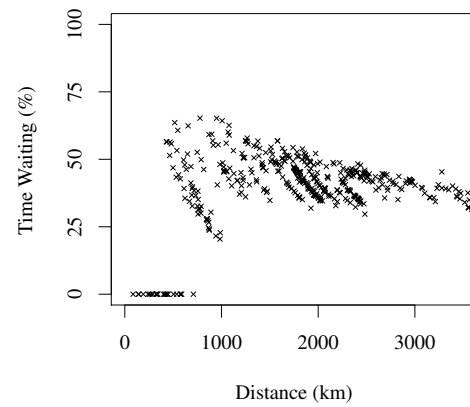
(a) London



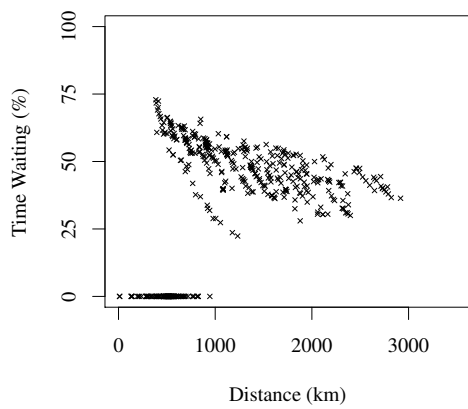
(b) Berlin



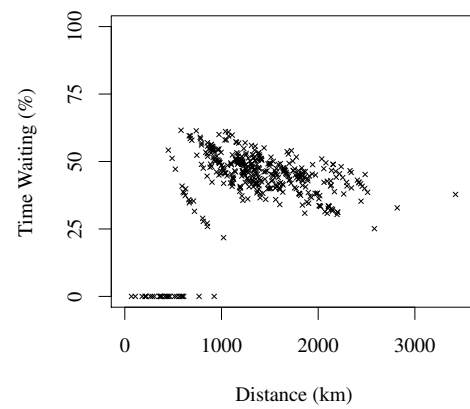
(c) Madrid



(d) Kiev



(e) Paris



(f) Rome

Figure 6.12: Percentage of journey spent waiting vs journey distance

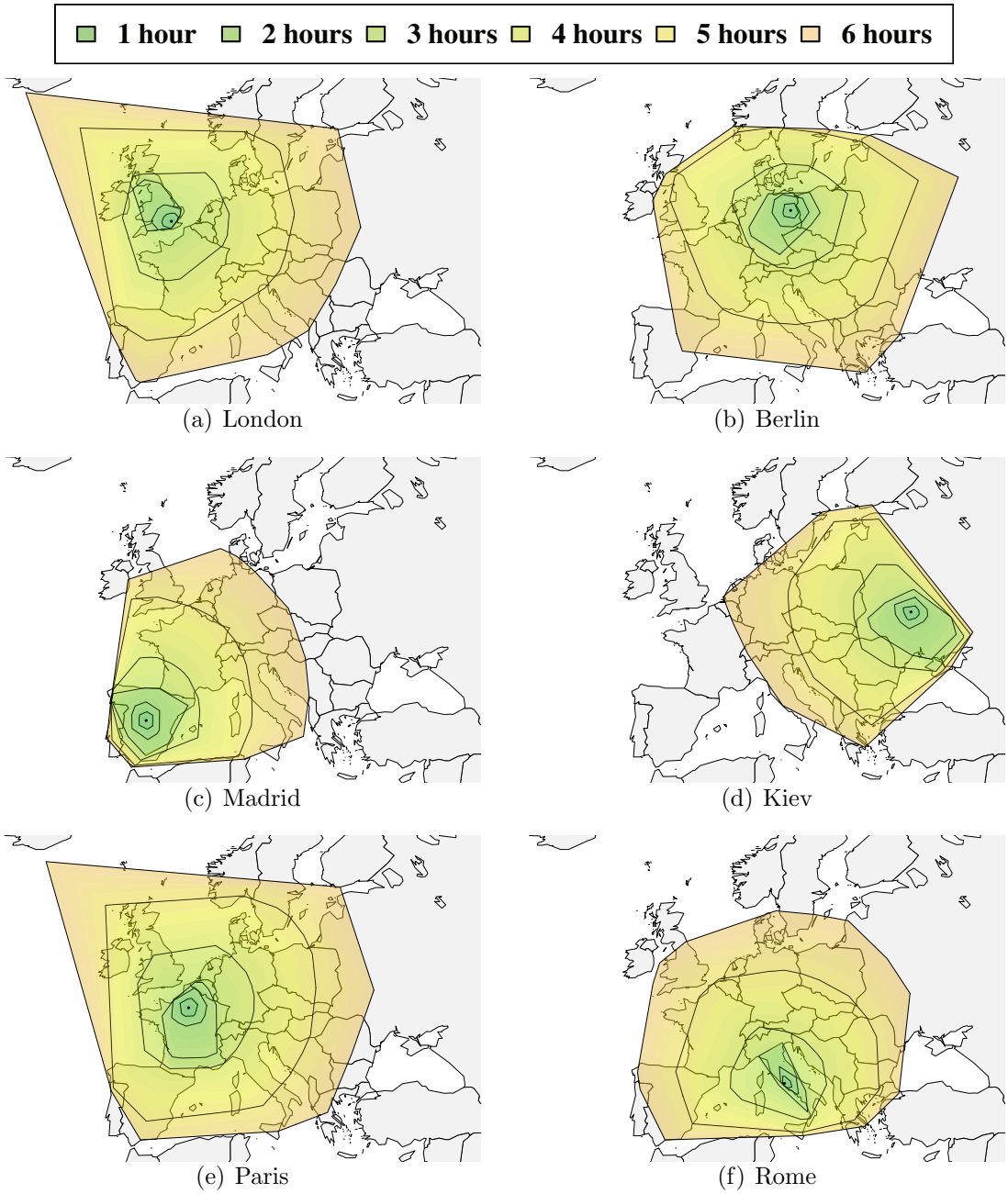


Figure 6.13: Isochrones of reachability

Europe, so all the journeys calculated are moving more towards central Europe. How uniformly the isochrone regions change as the time increases corresponds to easier flows within the network. Thus less-uniform changes represent possible bottle-necks, for example it is clear that the first few rings of the London isochrone are hindered by having to cross the sea and therefore a flight is required.

What is also clear is that if you look at each of the regions representing up to 4 hours worth of travel, then the 4 hour ACARE target seems a long way off. Given up to 5 hours, then a more reasonable amount of Europe can be reached. Given 6 hours then quite a large majority can be serviced and is likely closer to the ACARE target in terms of percentage of people connected. Therefore if wait times could be reduced by an hour then very roughly speaking we could reach the regions represented by the current 5-hour isochrone, within the 4 hour ACARE target. However, a single hour reduction seems unlikely to be enough to fully meet the target.

6.6 Summary

This chapter has investigated the use of adapted open-source data and software to analyse the door-to-door road and air journeys within Europe, in an attempt to assess the feasibility of the ACARE 4 hour targets.

The use of Open Street Map data allowed the road network to be accurately represented on a global scale. Additionally it defined a specific structure which would be required when including the airport and flight network. The open-source routing software known as Routino was introduced as the method for finding optimal routes within the OSM data. This included a two-tiered routing processes to efficiently calculate optimal paths even on a large scale. Furthermore a number of profiles could be created to account for different modes of transport and include the preferential treatment of road types.

With the structure of the OSM data and the Routino software prescribed, the airport and flight network was then appropriately modelled so the entire network could be combined. The airport network was designed to constrain flow and ensure aspects such as checking in and taxiing were included. The Routino routing software was then successfully adapted to be able to route for the entire, combined road, airport and flight network and thus calculate optimal door-to-door journeys between any two locations.

The workflow of using the door-to-door routing software was then outlined to go from the input of location pairs to a desired output. Importantly this considered

the end-user as a key component within the workflow and showed how either an automated method or a GUI can be used to simplify the process.

Finally the results of calculating the door-to-door routes from six major European cities to the 500 other most populated ones was explored. The results showed that some cities, such as Paris and Berlin were clearly better suited geographically for connecting to a large proportion of Europe more easily. Additionally, where journey-time was spent was analysed to show that a significant proportion of longer journeys durations are 'wasted', waiting at airports for overheads such as check ins. With the use of isochrones it was shown that while the 4-hour ACARE target is still some way off, reducing the 'wasted' time at airports would be key in improving overall connectivity within Europe.

Chapter 7

Conclusion

7.1 Thesis Summary

The recent and predicted future-growths in the demand for commercial flight motivates a real need for more efficient, economical and environmentally friendly approaches to aviation. While a number of potential future-concepts have been proposed over the years the use of formation flight for drag-reduction shows significant promise. Through a variety of flight tests, numerical simulation and analytic study researchers have demonstrated the ability to reduce drag by impressive amounts through the use of close and extended formation flight. The key research areas are broad, yet interconnected including: aerodynamics; automation and control; routing and assignment; and procedures and regulations. There is still a number of key problems that will first need addressing, from the automated micro-control problem of maintaining formation to the regulatory and procedural standards such as flight separation levels, however ongoing work encourages the idea that formation flight could one day become a reality.

While a handful studies have looked at the potential of deviating current flight routes, in order to fly in formation and save fuel, few have tackled the substantial and highly combinatorial fleet-assignment problem when routing for formation flight. The work of this thesis has therefore been to address a number of important questions needed to solve the problem of globally routing and assignment of commercial flights.

Accordingly, this problem has been broken down into the two overriding, interconnected questions, with the objective of minimising total fuel burn:

Objective

Optimise current routes to fly in formation to minimise total fuel burn.

Question 1

Given a set of flights, where should the aircraft meet in order to fly in formation?

Question 2

Given the cost of each formation, which of these formations should be used to minimise the total cost?

With this in mind the work outlined in Chapter 2 aims to provide a fundamental approach to answering these questions. A geometric technique to analytically calculating optimal formation routes is introduced. While this is used for finding formations of size two and three, the framework is shown to be extendable to theoretically any sized formation. The approach provides a fast method for routing hundred-of-thousands of formations (of size two) a minute allowing us to solve the global assignment problem. Given a list of solo flights, this fast routing method allows for every formation combination to be evaluated in a reasonable time, therefore the global assignment problem, of allocating flights into formations is made tractable. A Transatlantic case study is used to explore some of the savings attainable through this method, with possible average fuel-burn savings against solo flight of almost 9% for formations of 2 and just over 13% for formations of up to 3. Finally with one of the main assumptions being that solutions were time-free, it was important to also investigate the impact of current scheduling on formation flight. Restrictions to maximum scheduling changes was imposed during the assignment stage where it was shown that even with just a relatively small adjustment, of up to 30 minutes, to the current schedules meant formations of size two could achieve savings upwards of 8%.

With the basic methodology used throughout the thesis outlined in Chapter 2, Chapter 3 then looked at the problem of wind-optimal routing for formation flight. While the geometric approach assumed that flights would fly great-circles, in reality flights often do not, instead adjusting their routes to take into account factors such as wind. A numerical routing approach, using a finite set of variable way-points optimally placed by an active-set optimisation, was introduced to find optimal formation routes in the presence of wind. The numerical nature of the approach meant that this routing method required significantly more computational time than the geometric

approach of Chapter 2 making it far less practical for solving the global assignment problem. The aim of this chapter, was therefore used to compare, and ultimately benchmark, the use of the geometric approach to calculate an ‘approximate cost’ for use within the assignment stage.

With this in mind two separate work-flows were presented for assigning aircraft into formation fleets in the presence of wind. The first, required the calculation of the cost of the wind route for all combinations, followed by a MILP assignment. This produced a globally optimal solution but was a significantly more computationally intensive process. The second approach used the fast geometric method to estimate the highly-combinatorial assignment process and then the wind-routes were simply calculated as a post process. The results of this chapter showed that the geometric solution acts as a reasonable initial estimate to the assignment problem. Furthermore the use of a cost-estimating assignment process further improved upon this initial estimate in exchange for additional computation time. The results of the Estimated Assignment Method were extremely promising, creating assignments which differed from the global solution by as little as 0.2 of a percent while taking substantially less time to compute. In this sense, the wind routing method can be thought of as just one example of using higher-fidelity routing approaches within this problem, whereby the geometric method can provide a way of counteracting the combinatorial nature of the global assignment problem. While the more-complex routing could be feasibly left to a post-processing stage.

The work of Chapter 4 introduced the problem of uncertainty in aircraft takeoff times whereby negative binomial distributions were fitted to real world ground delay data. Two methods were then outlined to try and mitigate the impact this ground delay had on formations. The first method was a hold approach, instructing aircraft to enter a holding pattern at the point of rendezvous to absorb any delay to the other aircraft. The second method, and the main focus of this chapter, was a two-staged state-space approach was outlined using dynamic programming and value iteration to assign optimal speed-control policies for airborne aircraft to follow for any possible realisation of delay. Issues arising from discretisation of the state space were counteracted by two additions to the standard grid based approach. Firstly interpolation was used, allowing for both smoother cost functions and in turn smoother control choice. Secondly quadtree based sampling allowed the sampling resolution to be ‘shifted’ to areas of greater non-linearity improving the approximation of the linear interpolation.

A notion of risk was introduced, corresponding to the standard deviation of the achievable results for a given formation or formation assignment. The assignment

MILP of Chapter 2 was then adjusted to include a weighted penalty function on a formation's risk, allowing the risk tolerance to be tuned to filter out formations which were less robust. The concept of portfolio optimization was then also introduced to produce efficient frontiers defining the optimal level of reward corresponding to any level of risk-aversion. Finally the Transatlantic case study compared the two methods against and although both produced similar expected savings, by comparing efficient frontiers it was clear that the State Space approach produces significantly less-risky results for any level of reward. Interestingly it was also shown that very similar efficient frontiers could be achieved by simply flying at a fixed nominal speed until the delay is realised. This prompts the realisation that complex proactive policies may not be entirely necessary as long as a suitable en-route policies are used.

Three distinct case studies were presented in Chapter 5 to try and assess the potential of commercial formation flight on a global scale for formations of size two. This included the Transatlantic flights, used within Chapters 2, 3 and 4, a Singapore Airlines flight list and larger list of short/medium-haul EasyJet flights. Fleet average savings were shown to be around 8%, 6% and 2% for the Transatlantic, Singapore Airlines and EasyJet case studies respectively. While the results show clear potential for formation flight within these regions, their large difference between average savings highlighted how some sets of flights are clearly more suitable than others. With each of these case studies exhibiting a number of key characteristics, such as flight length, geographic location and differing flight topologies. This chapter explored the important features associated within a set of flight for them to achieve a good level of saving. The level of flight deviation was shown to be highly negatively-correlated against saving for the Transatlantic flights will having almost no correlation for the EasyJet and Singapore Airlines flights. Conversely, the flight distance was shown to be highly positively-correlated against savings for the Singapore Airlines and EasyJet flights, while remaining uncorrelated for the Transatlantic flights. The results of adjusting the formation discount factor outlined the range of possible savings achievable for different levels of drag-reduction.

Finally Chapter 6 outlines work undertaken during a six-month industrial placement developing door-to-door routing software to calculate shortest paths between any two locations. The aim was to assess the connectivity of major cities and how much is reachable from each city within a given time window. The routing software was developed to find the shortest-distance or shortest-time paths between any two locations using a bespoke road, airport and flight network. This included a number of key features, such as the ability to handle preferential treatment of road and flight

types and also include aspects of airport waiting times such as check-in, taxiing and domestic and international flight transfers. Open source data was used and combined with adapted open source software to develop a method capable of analysing a huge number of city pairs on a large global-network. Finally the results of calculating the door-to-door routes from six major European cities to the 500 other most populated ones was explored. Where journey time was spent was analysed, showing a significant proportion of longer journeys' durations were 'wasted', waiting at airports for overheads such as check ins. Additionally isochrone plots were used to show that reducing the 'wasted' time at airports would be key in improving overall connectivity and reaching the 4-hour ACARE target.

7.2 Summary of Contributions

The work explored throughout this thesis contributes a number of novel ideas in areas of routing and assignment for commercial formation flight. The following is a summary of the key contributions:

A geometric routing methodology for commercial formation flight

A unique adaption of a Fermat-Torricelli mathematical problem to model the formation flight problem. Allowing fast, analytical routing for theoretically any sized formation, whilst including aspects such as minimum climb/descend distances constraining and differential-fuel burn. This approach helps to tackle the impact of the combinatorial nature of enumerating all possible formation pairings.

Numerical method for formation routing through wind

Although wind-routing exists within single-aircraft routing and trajectory optimization the process has not yet been explored for commercial formation flight. This thesis uses an active-set approach with a numerical optimizer to find minimum-energy paths through a static wind-field.

An estimated assignment process for improving geometric allocation

A simple adapted assignment process to improve an initial geometric estimated MILP allocation of flights to formation fleets. A cost-estimating function, based on the geometric approach, is used to predict the potential cost for the significantly more complex wind-route.

Two-stage dynamic programming to mitigate effect of ground delay

Two previously unexplored methods are presented to try and mitigate the effect uncertainty in take-off time has on flights looking to join in formation. The focus is a two-stage dynamic programming formulation to calculate optimal-speed control policies to follow for any possible realisation of delay.

Portfolio analysis for choosing risk-based formation assignments

Taking methods used extensively in economics and finance and applying them to the formation flight assignment problem. Creating formation assignments to incorporate aspects of risk and uncertainty. Efficient frontiers are calculated to directly compare different assignments for the multi-objective optimization of minimising both cost and risk-level.

Analysis of three distinct case studies with formation flight potential

The largest assessment of the global assignment problem currently available. Three distinct flight lists of Transatlantic, EasyJet and Singapore Airlines flights are routed and assigned to minimize total fuel burn.

Door-to-Door routing for a road, airport and flight network

To assess the connectivity of major European cities door-to-door routing software was developed to calculate shortest paths between any two locations using a bespoke road, airport and flight network. This includes a number of key features, such as the ability to handle preferential treatment of road and flight types and also include aspects of airport waiting times such as check-in, taxiing and domestic and international flight transfers.

7.3 Future work

7.3.1 Extension of Current Methods and Work

Smarter approaches for assessing larger formations

The geometric approach has outlined a methodology which can be used to theoretically route for any sized formation. However, the added combinatorial problem of deciding the order of joining up and breaking away from formation would exponentially increase the time required for routing. Therefore a smarter, perhaps heuristic, approach should be explored for building upon the results of previous assignments of formations of size two and three.

Optimise for different metrics

While the objective of this thesis has been to minimise the total fuel-burn of a entire list of flights, the methods and framework outlined can be used for a number of different metrics. Metrics such as NO_x emissions; the overall duration of the flight; and direct operating costs are just a few of the many possible scenarios. For example perhaps instead of just saving money on fuel, the potential fuel-saved could be used to fly at an increased speed, resulting in shorter flights and therefore lower operating costs.

Decentralised assignment methods

This thesis has focused on a centralised approach to the formation routing and assignment problem, however a number of the approaches demonstrated could be used on a more decentralised level. Perhaps the formation routing could be applied at either the ATC tactical-phase or at a flow-management level routing airbourne flights together. This could potentially alleviate some of the uncertainty around take-off

Benchmark the assignment process for higher-fidelity routing methods

With the work of Chapter 3 showing that the post-processing stage is an ideal place for more complex routing, it would be of interest to extend this to other, higher-fidelity routing methods. A full trajectory-optimisation could be used to route for the formations to include a number of extra factors, producing more accurate potential routes.

Extend the state space approach to include en-route uncertainty

The state space approach of Chapter 4 was implemented in such a way as to include extra features at a later date. It would therefore be of interest to apply similar techniques of robust planning for the en-route portions of the flight, whereby uncertainties due to turbulence or weather could result in portions of the formation leg of the journey being unsuitable for drag-reduction. It would be of use to observe what kinds of policies, for mitigating this kind of uncertainty, arose from this problem.

7.3.2 Future Applications

Ad-hoc routing

While a decentralised, global assignment based approach would be ideal, it is likely that, due to a number of circumstances (such as bad weather and flight

cancellations), flights could be airbourne, wanting to join formation but have no other formation members. Therefore an ad-hoc approach, requiring minimal forward planning would be useful.

Air-to-air refuelling

A different ‘future-concept’ for aviation is for aircraft to be refuelled mid-air, by a large ‘tanker’ aircraft, allowing either for extended aircraft ranges or a reduction in fuel burn due to taking off with a much lower fuel weight. A number of the methods within this thesis could likely be extended to attempt to solve the routing problem for both the tanker and the regular aircraft. For example, the formation routing of Chapter 2, based on an assigned aircraft scalar weight, could be used, as the tanker would burn fuel at a much greater rate, it would be assigned much larger weight, incentivising the other aircraft to do most of the deviating.

Highways in the sky concepts

One more ‘future-concept’ for commercial flight involves the idea of creating virtual ‘Highways’ in the sky, where flights would fly to an ‘entry point’ and then fly along the more densely packed airways (for example along the North Atlantic Tracks), allowing for increased traffic while reducing ATC workload, but also leaving the potential for an area perfect for formation flight. The routing stages of this thesis could then be used to find optimal routes to the highways’ entry points, while on the highway formation fleets could combine to make larger formations.

7.3.3 Extensions of Door-to-Door Routing

Assess impact of formation flight might have on connectivity

A small connection between the formation flight problem and the door-to-door routing problem is that if formations opted to fly faster, with costs counteracted by the drag-reductions, then flight times could be shortened. This decrease in flight time could potentially increase the connectivity of major cities and go some way to reaching the 4-hour ACARE target.

Sensitivity analysis of airport waiting times

While the methods and framework have now been outlined, it would be of interest to undertake sensitivity analysis on a number of timing factors effecting

airports. For example, if check-in times could be reduced by 30 minutes how would that effect the connectivity of European cities.

Analysis of connectivity for industry and shipping

With the inclusion of different vehicle types such as Heavy Goods Vehicles (HGVs) it is possible to run a similar analysis to see how well cities are connected from a cargo and shipping point of view. With speed and road type restrictions for HGVs along with a whole number of additional time factors associated with cargo aircraft the connectivity results would be of great interest.

Additional transportation methods

The software adapted allows for a number of different transportation types this includes aircraft, cars, HGVs, motorbikes, bicycles and even walking. For increased realism it would be of interest to add additional transportation methods such as sea, rail and high-speed rail to see what impact this has.

Bibliography

- [1] R. Poli, L. Ravenel, and R. Besson, “IATA Annual Review 2014,” tech. rep., 2014.
- [2] “SESAR Definition phase 1: Deliverable 1,” tech. rep., 2007.
- [3] P. Forsyth, “The impacts of emerging aviation trends on airport infrastructure,” *Journal of Air Transport Management*, vol. 13, pp. 45–52, Jan. 2007.
- [4] M. Dareck, C. Edelstenn, and T. Ender, “Flightpath 2050 Europe’s Vision for Aviation,” tech. rep., European Commission, 2011.
- [5] N. E. Antoine and I. M. Kroo, “Aircraft Optimization for Minimal Environmental Impact,” *Journal of Aircraft*, vol. 41, no. 4, pp. 790–797, 2004.
- [6] J. I. Hileman, H. M. Wong, I. a. Waitz, D. S. Ortiz, J. T. Bartis, M. a. Weiss, and P. E. Donohoo, “Near-Term Feasibility of Alternative Jet Fuels,” *Aviation*, vol. 21, no. 6, pp. 1–150, 2009.
- [7] G. M. Forecast, “Flying on demand: Airbus Global Market Forecast 2014-2033,” tech. rep., 2014.
- [8] Air Transport Action Group (ATAG), “The Global Flightplan: Reducing Emission form the International Sector Through Carbon-Neutral Growth From 2020,” Tech. Rep. November, Air Transport Action Group, Warsaw, 2013.
- [9] L. Dray, A. Evans, T. Reynolds, and A. Schäfer, “Mitigation of Aviation Emissions of Carbon Dioxide,” 2010.
- [10] L. Dray, “An analysis of the impact of aircraft lifecycles on aviation emissions mitigation policies,” *Journal of Air Transport Management*, vol. 28, pp. 62–69, 2013.

- [11] C. Haissig, “Military formation flight as a model for increased capacity in civilian airspace,” in *The 23rd Digital Avionics Systems Conference*, no. IEEE Cat. No. 04CH37576, pp. 1.C.4–1.1–9, IEEE, Oct. 2004.
- [12] R. K. Nangia, “Greener civil aviation using air-to-air refuelling - relating aircraft design efficiency and tanker offload efficiency,” *The Aeronautical Journal*, vol. 2006, no. 3186, pp. 589–592, 2007.
- [13] R. K. Nangia, “Way Forward to a Step Jump for Highly Efficient & Greener Civil Aviation An Opportunity for the Present & a Vision for Future,” *Personal Publication RKN-SP-2008-120*, vol. 120, no. November, 2008.
- [14] N. Qin, A. Vavalle, A. Lemoigne, M. Laban, K. Hackett, and P. Weiserfetl, “Aerodynamic considerations of blended wing body aircraft,” *Progress in Aerospace Sciences*, vol. 40, pp. 321–343, Aug. 2004.
- [15] G. Bower, T. C. Flanzer, and I. Kroo, “Formation Geometries and Route Optimization for Commercial Formation Flight,” in *27th AIAA Applied Aerodynamics Conference*, no. AIAA Paper 2009-3615, (San Antonio, Texas), American Institute of Aeronautics and Astronautics, June 2009.
- [16] L. L. Gould and F. Heppner, “The Vee Formation of Canada Geese,” *The Auk*, vol. 91, pp. 494–506, July 1974.
- [17] F. R. Hainsworth, “Wing movements and positioning for aerodynamic benefit by Canada geese flying in formation,” *Canadian Journal of Zoology*, vol. 67, pp. 585–589, Mar. 1989.
- [18] C. J. Cutts and J. R. Speakman, “Energy savings in formation flight of pink-footed geese,” *Journal of Experimental Biology*, vol. 189, pp. 251–261, 1994.
- [19] H. Weimerskirch, J. Martin, Y. Clerquin, P. Alexandre, and S. Jiraskova, “Energy saving in flight formation.,” *Nature*, vol. 413, pp. 697–8, Oct. 2001.
- [20] S. J. Portugal, T. Y. Hubel, J. Fritz, S. Heese, D. Trobe, B. Voelkl, S. Hailes, A. M. Wilson, and J. R. Usherwood, “Upwash exploitation and downwash avoidance by flap phasing in ibis formation flight.,” *Nature*, vol. 505, no. 7483, pp. 399–402, 2014.
- [21] P. B. S. Lissaman and C. A. Shollenberger, “Formation flight of birds,” *Science*, vol. 168, no. 3934, pp. 1003–1005, 1970.

- [22] W. Okolo, A. Dogan, and W. Blake, “Effect of Trail Aircraft Trim on Optimum Location in Formation Flight,” *Journal of Aircraft*, pp. 1–13, 2014.
- [23] J. Pahle, D. Berger, M. Venti, C. Duggan, J. Faber, and K. Cardinal, “An Initial Flight Investigation of Formation Flight for Drag Reduction on the C-17 Aircraft,” in *AIAA Atmospheric Flight Mechanics Conference*, no. August, (Minneapolis, Minnesota), pp. 1–13, American Institute of Aeronautics and Astronautics, Aug. 2012.
- [24] S. A. Ning, *Aircraft drag reduction through extended formation flight*. PhD thesis, Stanford University, 2011.
- [25] Y. Liu, K. Risse, K. Franz, and E. Stumpf, “Assessment of Potential Benefit of Formation Flight at Preliminary Aircraft Design Level,” in *53rd AIAA Aerospace Sciences Meeting*, no. January, pp. 1–15, 2015.
- [26] W. Blake and D. R. Gingras, “Comparison of Predicted and Measured Formation Flight Interference Effects,” *Journal of Aircraft*, vol. 41, no. 2, pp. 201–207, 2004.
- [27] W. Blake and D. Multhopp, “Design, Performance and Modeling Considerations for Close Formation Flight,” in *23rd Atmospheric Flight Mechanics Conference*, vol. 2, pp. 476–486, 1998.
- [28] D. Jacques, M. Pachter, G. Wagner, and B. Blake, “An analytical study of drag reduction in tight formation flight,” in *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, no. August, (Reston, Virginia), American Institute of Aeronautics and Astronautics, Aug. 2001.
- [29] M. G. Wagner, L. D. Jacques, W. Blake, M. Pachter, and W. P. Afb, “Flight Test Results of Close Formation Flight for Fuel Savings,” *Test*, no. August, pp. 1–11, 2002.
- [30] M. J. Vachon, R. Ray, K. Walsh, and K. Ennix, “F/A-18 Aircraft Performance Benefits Measured During the Autonomous Formation Flight Project,” in *AIAA Atmospheric Flight Mechanics Conference*, no. AIAA Paper 2002-4491, (Monterey, California), American Institute of Aeronautics and Astronautics, Aug. 2002.
- [31] R. Ray, B. Cobleigh, M. J. Vachon, and C. St. John, “Flight Test Techniques used to Evaluate Performance Benefits During Formation Flight,” in *AIAA*

- Atmospheric Flight Mechanics Conference*, no. AIAA Paper 2002-4492, (Monterey, California), American Institute of Aeronautics and Astronautics, Aug. 2002.
- [32] S. A. Ning, T. C. Flanzer, and I. Kroo, “Aerodynamic Performance of Extended Formation Flight,” *Journal of Aircraft*, vol. 48, pp. 855–865, May 2011.
- [33] L. DeVries and D. A. Paley, “Wake Estimation and Optimal Control for Autonomous Aircraft in Formation Flight,” in *AIAA Guidance, Navigation, and Control (GNC) Conference*, no. AIAA Paper 2013-4705, (Boston, Massachusetts), American Institute of Aeronautics and Astronautics, Aug. 2013.
- [34] M. Brodecki and K. Subbarao, “Autonomous Formation Flight Control System Using In-Flight Sweet-Spot Estimation,” *Journal of Guidance, Control, and Dynamics*, pp. 1–14, Aug. 2014.
- [35] D. F. Chichka, J. L. Speyer, C. Fanti, and C. G. Park, “Peak-Seeking Control for Drag Reduction in Formation Flight,” *Journal of Guidance, Control, and Dynamics*, vol. 29, pp. 1221–1230, Sept. 2006.
- [36] M. Pachter, J. J. D’, and A. W. Proud, “Tight Formation Flight Control,” *Journal of Guidance, Control, and Dynamics*, vol. 24, pp. 246–254, Mar. 2001.
- [37] P. Binetti, K. Ariyur, M. Krstic, and F. Bernelli, “Control of formation flight via extremum seeking,” in *Proceedings of the 2002 American Control Conference*, no. IEEE Cat. No.CH37301, (Anchorage, Alaska), pp. 2848–2853, American Automatic Control Council, May 2002.
- [38] R. K. Nangia and M. E. Palmer, “Formation Flying of Commercial Aircraft, Variations in Relative Size/Spacing - Induced Effects & Control Induced Effects & Control,” in *25th AIAA Applied Aerodynamics Conference*, no. AIAA Paper 2007-4163, (Miami, Florida), pp. 25–28, American Institute of Aeronautics and Astronautics, June 2007.
- [39] J. Xu, S. Ning, G. Bower, and I. Kroo, “Aircraft Route Optimization for Heterogeneous Formation Flight,” in *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, no. AIAA Paper 2012-1524, (Honolulu, Hawaii), American Institute of Aeronautics and Astronautics, Apr. 2012.

- [40] R. King and A. Gopalathnam, “Ideal Aerodynamics of Ground Effect and Formation Flight,” *Journal of Aircraft*, vol. 42, no. 5, pp. 1188–1199, 2005.
- [41] G. Ribichini and E. Frazzoli, “Efficient coordination of multiple-aircraft systems,” in *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*, vol. 1, pp. 1035–1040, IEEE, 2003.
- [42] Eurocontrol, “User Manual for the Base of Aircraft Data (BADA) Revision 3.10,” Tech. Rep. 12/04/10-45, 2012.
- [43] D. Service, J. N. Hallock, and J. A. Volpe, “Aircraft Wake Vortices: An Assessment of the Current Situation,” tech. rep., DTIC Document, 1991.
- [44] W. Mason and S. Iglesias, “Optimum spanloads in formation flight,” in *40th AIAA Aerospace Sciences Meeting & Exhibit*, no. January, (Reston, Virginia), American Institute of Aeronautics and Astronautics, Jan. 2002.
- [45] H. P. Thien, M. A. Moelyadi, and H. Muhammad, “Effects of Leaders Position and Shape on Aerodynamic Performances of V Flight Formation,” *arXiv preprint*, p. 7, Apr. 2008.
- [46] S. R. Bieniawski, R. W. Clark, S. E. Rosenzweig, and W. B. Blake, “Summary of Flight Testing and Results for the Formation Flight for Aerodynamic Benefit Program,” in *52nd Aerospace Sciences Meeting*, no. January, 2014.
- [47] J. Wolfe, D. Chichka, and J. Speyer, “Decentralized controllers for unmanned aerial vehicle formation flight,” *San Diego, CA*, no. July, pp. 3833–3833, 1996.
- [48] D. Chichka, J. Speyer, and C. Park, “Peak-seeking control with application to formation flight,” in *Conference on Decision & Control*, no. December, (Phoenix, Arizona), pp. 2463–2470, 1999.
- [49] Y. Zou, P. R. Pagilla, and R. T. Ratliff, “Distributed Formation Flight Control Using Constraint Forces,” *Journal of Guidance, Control, and Dynamics*, vol. 32, pp. 112–120, Jan. 2009.
- [50] J. Moon and R. Sattigeri, “Adaptive guidance and control for autonomous formation flight,” *American Helicopter Society 63rd Annual Forum*, 2007.
- [51] W. Zhao, T. Go, and E. Low, “Formation Flight Control Using Model Predictive Approach,” in *47th AIAA Aerospace Sciences Meeting including The New*

- Horizons Forum and Aerospace Exposition*, no. January, (Reston, Virginia), pp. 1–8, American Institute of Aeronautics and Astronautics, Jan. 2009.
- [52] J. Xu, S. Andrew Ning, G. Bower, and I. Kroo, “Aircraft Route Optimization for Formation Flight,” *Journal of Aircraft*, vol. 51, pp. 490–501, Mar. 2014.
- [53] G. Duke, *Air Traffic Control*. Hersham: Midland Publishing, 10th ed., 2009.
- [54] P. Bonami, A. Olivares, M. Soler, and E. Staffetti, “Multiphase Mixed-Integer Optimal Control Approach to Aircraft Trajectory Optimization,” *Journal of Guidance, Control, and Dynamics*, vol. 36, pp. 1267–1277, Sept. 2013.
- [55] M. Soler, A. Olivares, and E. Staffetti, “Hybrid Optimal Control Approach to Commercial Aircraft Trajectory Planning,” *Journal of Guidance, Control, and Dynamics*, vol. 33, pp. 985–991, May 2010.
- [56] J. T. Betts, “Survey of Numerical Methods for Trajectory Optimization,” *Journal of Guidance, Control, and Dynamics*, vol. 21, pp. 193–207, Mar. 1998.
- [57] T. E. Kent and A. G. Richards, “On Optimal Routing For Commercial Formation Flight,” in *AIAA Guidance, Navigation, and Control (GNC) Conference*, no. AIAA Paper 2013-4889, (Boston, Massachusetts), pp. 1–11, American Institute of Aeronautics and Astronautics, Aug. 2013.
- [58] J. D. Anderson, *Introduction to Flight*. McGraw-Hill Series in Aeronautical and Aerospace Engineering, New York: McGraw-Hill, third ed., 2000.
- [59] S. Gueron and R. Tessler, “The Fermat-Steiner Problem,” *The American Mathematical Monthly*, vol. 109, p. 443, May 2002.
- [60] M. D. de Villiers, “A Generalisation of the Fermat-Torricelli Point,” *The Mathematical Gazette*, vol. 79, pp. 374–378, July 1995.
- [61] Y. Shen and J. Tolosa, “The Weighted Fermat Triangle Problem,” *International Journal of Mathematics and Mathematical Sciences*, vol. 2008, 2008.
- [62] A. Zachos and A. Cotsiolis, “The weighted Fermat Torricelli problem on a surface and an inverse problem,” *Journal of Mathematical Analysis and Applications*, vol. 373, pp. 44–58, Jan. 2011.
- [63] K. Ghalieh and M. Hajja, “The Fermat Point of a Spherical Triangle,” *The Mathematical Gazette*, vol. 80, p. 561, Nov. 1996.

- [64] C. Grob and T.-K. Stempel, “On Generalizations of Conics and on a Generalization of the Fermat-Torricelli Problem,” *The American Mathematical Monthly*, vol. 105, p. 732, Oct. 1998.
- [65] S. Abu-Saymeh and M. Hajja, “On the Fermat-Torricelli Points of Tetrahedra and of Higher Dimensional Simplexes,” *Mathematics Magazine*, vol. 70, pp. 372–378, Dec. 1997.
- [66] E. J. Cockayne, “On Fermat’s Problem on the Surface of a Sphere,” *Mathematics Magazine*, vol. 45, p. 216, Sept. 1972.
- [67] F. Eriksson, “The Fermat-Torricelli Problem Once More,” *The Mathematical Gazette*, vol. 81, Mar. 1997.
- [68] L. Chan, *Seismic Performance of Shear Walls Utilizing Cellular Material*. PhD thesis, The Catholic University of America, 2009.
- [69] J. E. Taylor, “The structure of singularities in soap-bubble-like and soap-film-like minimal surfaces,” *Annals of Mathematics*, vol. 103, no. 3, pp. 489–539, 1976.
- [70] T. H. Colding and W. P. Minicozzi, “Shapes of embedded minimal surfaces,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 103, pp. 11106–11, July 2006.
- [71] S. Fortuna, D. L. Cheung, and A. Troisi, “Hexagonal lattice model of the patterns formed by hydrogen-bonded molecules on the surface,” *The journal of physical chemistry. B*, vol. 114, pp. 1849–58, Feb. 2010.
- [72] M. R. Philpott and Y. Kawazoe, “Triplet states of zigzag edged hexagonal graphene molecules $C(6m \times 2)H(6m)$ ($m = 1, 2, 3, \dots, 10$) and carbon based magnetism,” *The Journal of chemical physics*, vol. 134, pp. 124706 1–9, Mar. 2011.
- [73] W. R. Tobler, “A Classification Of Map Projections,” *Annals of the Association of American Geographers*, vol. 52, pp. 167–175, June 1962.
- [74] M. Garey and D. Johnson, *Computers and Intractability: a guide to NP-completeness*. San Francisco: W.H. Freeman and Company, 1979.
- [75] J. T. Linderoth and A. Lodi, “MILP Software,” *Wiley Encyclopedia of Operations Research and Management Science*, 2011.

- [76] A. Bayen and C. Tomlin, “MILP formulation and polynomial time algorithm for an aircraft scheduling problem,” in *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*, no. December, pp. 5003–5010, IEEE, 2003.
- [77] F. Soumis, J. A. Ferland, and J.-m. Rousseau, “A model for large-scale aircraft routing and scheduling problems,” *Transportation Research Part B: Methodological*, vol. 14, pp. 191–201, Mar. 1980.
- [78] M. Lohatepanont and C. Barnhart, “Airline Schedule Planning: Integrated Models and Algorithms for Schedule Design and Fleet Assignment,” *Transportation Science*, vol. 38, pp. 19–32, Feb. 2004.
- [79] D. O. Fraser, “Optimum Flight Paths,” 1951.
- [80] W. Warntz, “Transatlantic flights and pressure patterns,” *Geographical Review*, vol. 51, no. 2, pp. 187–212, 1961.
- [81] M. R. Jardin and A. E. Bryson, “Neighboring Optimal Aircraft Guidance in Winds,” *Journal of Guidance, Control, and Dynamics*, vol. 24, pp. 710–715, July 2001.
- [82] Y. . Qi and Y. J. Zhao, “Energy-Efficient Trajectories of Unmanned Aerial Vehicles Flying through Thermals,” *Journal of Aerospace Engineering*, vol. 18, pp. 84–92, Apr. 2005.
- [83] T. McGee, S. Spry, and K. Hedrick, “Optimal path planning in a constant wind with a bounded turning rate,” in *AIAA Guidance, Navigation, and Control (GNC) Conference*, (San Francisco, California), pp. 1–11, American Institute of Aeronautics and Astronautics, Aug. 2005.
- [84] R. L. McNeely, R. V. Iyer, and P. R. Chandler, “Tour Planning for an Unmanned Air Vehicle Under Wind Conditions,” *Journal of Guidance, Control, and Dynamics*, vol. 30, pp. 1299–1306, Sept. 2007.
- [85] K. Nakai and K. Uchiyama, “Vector Fields for UAV Guidance Using Potential Function Method for Formation Flight,” in *AIAA Guidance, Navigation, and Control (GNC) Conference*, (Boston, Massachusetts), pp. 1–13, American Institute of Aeronautics and Astronautics, Aug. 2013.

- [86] W. Al-Sabban, “Wind-Energy based Path Planning For Unmanned Aerial Vehicles Using Markov Decision Processes,” *Robotics and Automation . . .*, 2013.
- [87] R. Esmaelzadeh and A. Naghash, “Rendezvous trajectory optimization using real genetic algorithm combined with gradient method.,” *WSEAS Transactions on Systems*, vol. 6, no. 12, pp. 1–6, 2006.
- [88] R. F. Patron, A. Kessaci, and R. M. Botez, “Flight trajectories optimization under the influence of winds using genetic algorithms,” in *AIAA Guidance, Navigation, and Control (GNC) Conference*, (Boston, Massachusetts), pp. 1–11, American Institute of Aeronautics and Astronautics, Aug. 2013.
- [89] P. Sengupta, M. D. Tandale, and P. K. Menon, “Risk-Hedged Traffic Flow Management under Airspace Capacity Uncertainties,” in *AIAA Guidance, Navigation, and Control (GNC) Conference*, (Boston, Massachusetts), pp. 1–17, American Institute of Aeronautics and Astronautics, Aug. 2013.
- [90] M. Matthews, M. Wolfson, R. DeLaura, J. Evans, C. Reiche, H. Balakrishnan, and D. Michalek, “Measuring the Uncertainty of Weather Forecasts Specific to Air Traffic Management Operations,,” 2009.
- [91] J. a. Fessler and B. P. Sutton, “Nonuniform fast Fourier transforms using min-max interpolation,” *IEEE Transactions on Signal Processing*, vol. 51, no. 2, pp. 560–574, 2003.
- [92] “BlueCrystal Phase 2.”
- [93] G. Cook, Andrew J and Tanner, “European airline delay cost reference values,” *EUROCONTROL Performance Review Unit*, no. March, 2011.
- [94] H. R. Idris, B. Delcaire, I. Anagnostakis, W. D. Hall, N. Pujet, E. Feron, R. J. Hansman, J.-P. Clarke, and A. Odoni, “Identification of flow constraint and control points in departure operations at airport systems,” in *AIAA Guidance, Navigation, and Control (GNC) Conference*, (Boston, Massachusetts), 1998.
- [95] “Flightstats website.”
- [96] D. P. Bertsekas, “Dynamic Programming and Optimal Control 3rd Edition , Volume II by Chapter 6 Approximate Dynamic Programming Approximate Dynamic Programming,” *Control*, vol. II, pp. 1–200, 2010.

- [97] A. Antos, C. Szepesvarf, and R. Munos, “Value-Iteration Based Fitted Policy Iteration: Learning with a Single Trajectory,” *IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning*, 2007.
- [98] R. Bellman, *Dynamic Programming*, vol. 11 of *RAND CORPORATION. Research studies*. Princeton University Press, 1957.
- [99] W. Uther and M. Veloso, “Tree based discretization for continuous state space reinforcement learning,” *Proceedings of the National Conference on Artificial Intelligence*, pp. 769–775, 1998.
- [100] K. Doya, “Reinforcement learning in continuous time and space.,” *Neural computation*, vol. 12, no. 1, pp. 219–245, 2000.
- [101] S. Davies, “Multidimensional triangulation and interpolation for reinforcement learning,” *Advances in neural information processing systems*, pp. 1005–1011, 1997.
- [102] D. Kim, J. Lee, K. E. Kim, and P. Poupart, “Point-based value iteration for constrained POMDPs,” *IJCAI International Joint Conference on Artificial Intelligence*, vol. 7, pp. 1968–1974, 2011.
- [103] R. Munos and A. Moore, “Variable Resolution Discretization in Optimal Control Variable Resolution Discretization in Optimal Control,” *Machine Learning*, 1999.
- [104] L. F. Bertuccelli, M. Alighanbari, and J. P. How, “Robust planning for coupled cooperative UAV missions,” *43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601)*, pp. 2917–2922 Vol.3, 2004.
- [105] J. G. de Wit and J. Zuidberg, “The growth limits of the low cost carrier model,” *Journal of Air Transport Management*, vol. 21, pp. 17–23, 2012.
- [106] G. Burghouwt, J. Hakfoort, and J. R. van Eck, “The spatial configuration of airline networks in Europe,” *Journal of Air Transport Management*, vol. 9, pp. 309–323, Sept. 2003.
- [107] J. Lin and Y. Ban, “The evolving network structure of US airline system during 1990-2010,” *Physica A: Statistical Mechanics and its Applications*, vol. 410, pp. 302–312, 2014.

- [108] “Open Street Map Wiki.”
- [109] “Routino documentation.”
- [110] J. Lerner, DorotheaWagner, and K. a. Zweig, *Algorithmics of Large and Complex Networks*. 2009.
- [111] E. W. Dijkstra, “A note on two problems in connexion with graphs,” *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.
- [112] “Overpass Api.”