

Imperial College London
Department of Civil & Environmental Engineering

Optimisation of Humanitarian Response using Unmanned Aerial Vehicles

Jose Javier Escribano Macias

Submitted in part fulfilment of the requirements
for the degree of Doctor of Philosophy at
Imperial College London, April 2021

I hereby declare that this thesis and the work reported herein was composed by and originated entirely from me. Information derived from the published and unpublished work of others has been acknowledged in the text and references are given in the list of sources.

Jose Javier Escribano Macias (2020)

The copyright of this thesis rests with the author. Unless otherwise indicated, its contents are licensed under a Creative Commons Attribution-Non Commercial 4.0 International Licence (CC BY-NC).

Under this licence, you may copy and redistribute the material in any medium or format. You may also create and distribute modified versions of the work. This is on the condition that: you credit the author and do not use it, or any derivative works, for a commercial purpose.

When reusing or sharing this work, ensure you make the licence terms clear to others by naming the licence and linking to the licence text. Where a work has been adapted, you should indicate that the work has been changed and describe those changes.

Please seek permission from the copyright holder for uses of this work that are not included in this licence or permitted under UK Copyright Law.

Abstract

Man-made and natural disasters have been affecting an increasing number of people in recent years. This trend is expected to continue as the number and size coastal settlements grow, forcing humanitarian organisations to seek cost-effective solutions to improve response operations.

In recent years, Unmanned Aerial Vehicles have been increasingly deployed in humanitarian response as they provide an economical solution to the numerous challenges of post-disaster land-based relief operations, including the presence of partial or complete damage to the transport infrastructure, or absence thereof. In addition to some regulatory constraints, vehicle range and integration with the existing response mechanisms remain as key limitations to their deployment in practice. Therefore, it is imperative to develop a quantitative framework capable of determining the optimal strategy for UAV-based humanitarian response. This thesis proposes novel methodologies to use UAVs in relief distribution operations.

A method for routing UAVs and locating supporting infrastructure necessary to deliver relief cargo is developed. A trajectory management process is proposed to represent UAV behaviour during cruise flight accurately. Building on this, we also study the relationship between payload and energy consumption. An endogenous stochastic model is developed to assess network damage and inform the routing of ground vehicles.

The algorithms are applied on a dataset drawn from the earthquake responses in Taiwan after the Chi-Chi earthquake of 1999 and Port-au-Prince after the Haiti earthquake in 2010. Results show that UAVs can be used as a means of transportation for relief during the first days of a disaster, with close to 20,000 kg of cargo transported in 35 hours. The approach developed provides up to 40% improvements in terms of mission duration compared to well-established methods. By assisting trucks through network assessment, UAVs provide vehicle travel times by up to 20% depending on the network damage levels. These results indicate that fleets of small UAVs can provide significant benefits to humanitarian response.

Acknowledgements

The completion of this thesis would not have been possible with the support from people I would like to express my gratitude to:

- To my supervisors Dr Panagiotis Angeloudis and Professor Washington Ochieng. Their guidance and support has allowed me to pursue and develop my ideas, for which I feel permanently indebted.
- To the Department of Civil and Environmental Engineering at Imperial College and the Engineering and Physical Sciences Research Council (EPSRC) that sponsored my PhD study.
- To my colleagues at the Centre for Transport Studies (CTS) and the Transport Systems and Logistics Laboratory (TSL).
- To my colleagues at WFP, in particular to Haidar Baqir, who offered me the opportunity to work at the organisation.
- A mi familia, en especial a mi hermano Fernando y a mis padres. Vuestro apoyo incondicional e irrevocable me ha transmitido la energía y fuerza necesaria para seguir adelante en los días mas grises, consiguiendo así que las dificultades sean mas livianas y las alegrías aún mayores. Todo los logros obtenidos antes, durante y tras este doctorado os los debo a vosotros.
- A Àngela, gràcies per tot el teu suport i consell. Aconsegueixes que cada dia siga millor que l'anterior. Aquesta tesi no estaria escrita sense tu al meu costat. Te vuic.

A mis abuelos. Aunque no estén con nosotros, siempre los llevo conmigo.

Contents

Abstract	3
Acknowledgements	4
Dedication	5
1 Introduction	14
1.1 Humanitarian Logistics	15
1.2 Overview of UAV Technology	19
1.3 Drone regulations	21
1.4 Statement of the Problem	23
1.5 Aims and Objectives	25
1.6 Outline	26
2 Current Approaches for Humanitarian Operations	29
2.1 Origins of Humanitarian Logistics	30
2.2 Humanitarian Strategy	31
2.3 Humanitarian and Commercial Logistics	33
2.4 Stakeholders in Humanitarian Logistics	34
2.5 Humanitarian Logistics Stages	36
2.5.1 Preparedness Phase	38
2.5.2 Response Phase	42
2.5.3 Rehabilitation Phase	52
2.6 Concluding Remarks	54
3 Drone Operations Regulations and Methods	56
3.1 Drone Regulations	57
3.1.1 Current Regulatory Framework	58
3.1.2 Traffic Management Services	60
3.1.3 Performance Based Navigation	62
3.2 Trajectory Planning	63
3.2.1 Mathematical Programming	64
3.2.2 Heuristic Algorithms	67
3.3 Concluding remarks	70
4 Fight Dynamics and Optimisation Methods	72
4.1 UAV Flight Dynamics	73
4.1.1 Vertical Take-off and Landing	73
4.1.2 Fixed-Wing Aircraft	74

4.2	Optimisation Theory	76
4.2.1	Exact Solution Methods	77
4.2.2	Heuristics	78
4.2.3	Metaheuristics	78
4.3	Thesis structure	80
4.4	Algorithm Selection Criterion	82
4.5	Concluding Remarks	84
5	Rapid Medical Deliveries Framework using Unmanned Aerial Vehicles	85
5.1	Framework	86
5.2	Trajectory Optimisation Model	87
5.2.1	Model Formulation	90
5.3	Depot location and routing for UAV operations	93
5.3.1	Mathematical Model	94
5.3.2	Sub-problem	98
5.3.3	Route Assignment	100
5.3.4	Metaheuristic solution algorithm	103
5.4	Case Study	107
5.4.1	Parameter Tuning	109
5.4.2	Results	112
5.5	Concluding Remarks	120
6	Sequential Real-time Path Planning for Unmanned Aerial Vehicle Fleet Management	122
6.1	Framework	123
6.2	Airspace Segmentation	124
6.2.1	Formulation	125
6.2.2	Solution Method	127
6.3	Real-time Path Planning Component	130
6.3.1	Sequential Implementation	133
6.3.2	Heuristic Approach	135
6.4	Results and Discussion	137
6.4.1	Path Planning Results	138
6.4.2	Airspace Segmentation Results	142
6.4.3	Sequential Trajectory Results	143
6.4.4	Discussion	145
6.5	Concluding Remarks	146
7	Endogenous stochastic optimisation for relief distribution assisted with Unmanned Aerial Vehicles	148
7.1	Current Methods for Road Damage Assessment	150
7.1.1	Post-disaster relief optimisation	150
7.1.2	Exogenous stochastic optimisation	152
7.1.3	Endogenous stochastic programming	153
7.2	Model Description	154
7.2.1	Mathematical formulation	156
7.2.2	Non-Anticipativity Constraints	160
7.3	Solution Method	162

7.3.1	Sequential algorithm	163
7.3.2	Genetic Algorithm	165
7.4	Case study	167
7.4.1	Sequential Algorithm Results	168
7.4.2	Genetic Algorithm results	171
7.4.3	Discussion	180
7.5	Concluding Remarks	181
8	Conclusions and future work	183
8.1	Research Objectives Analysis	183
8.1.1	Future work	189
A	Humanitarian Operations Literature Table	229
B	Drone Operations Literature Table	235
C	Optimal Relief Appendix	238
C.1	Flight Optimal Control Problem	238
C.2	Route Assignment Algorithm	239
C.3	Detailed Results	239
D	Trajectory Optimisation Appendix	250
E	Endogenous Stochastic Research Appendix	256
E.1	Formulation Linearisation	256
E.2	Genetic Algorithm Tuning	259

List of Tables

1.1	Immediacy of response and predictability depending on the disaster category. Adapted from Cozzolino (2012).	15
1.2	UAV projects for delivery and passenger transport.	20
1.3	UAV classification. Adapted from Ren, Castillo-Effen, Yu, Yoon, Nakamura, Johnson & Ippolito (2017). LTA - Lighter-than-air, HTA - Heavier-than-air . . .	21
2.1	Cluster Coordination Agencies UNOCHA (2019 <i>c</i>).	32
2.2	Humanitarian and Commercial Logistics Differences.	34
2.3	Humanitarian Stakeholder Levels.	36
2.4	Hub Location Problem Types.	40
2.5	Vehicle Routing Problem Characteristics.	43
2.6	Traffic Assignment Model Types.	50
3.1	EASA Drone Categories (EASA 2015).	58
3.2	Comparison between regulations in the European Union (EU) and the United States of America (US).	59
3.3	U-Space Service Levels (SESAR 2019).	60
3.4	UTM Technical Capacity Levels.	62
3.5	PBN performance measurements (ICAO 2008, 2009)	63
5.2	Flight stages during trajectory optimisation.	93
5.4	Model performance for CPLEX. Solutions were not obtained for missions with durations within 24-hour execution window	99
5.5	Heuristic runtime in sample problems. Fitness deviations compared with Table 5.4.	105
5.6	Nanto County post-earthquake data including coordinates of each warehouse location Sheu (2007).	108
5.7	Model Parameters.	110
5.8	UAV itinerary example.	110
5.9	Model outputs (without delivery balancing).	113
5.10	Model outputs using the Simulated Annealing (SA) and bi-level SA models. . .	114
5.11	Model outputs (with delivery balancing).	115
5.12	Model outputs (with conservative battery management).	116
5.13	Comparison of humanitarian response case studies involving the Chi-Chi Earthquake.	119
6.3	Segmentation Algorithm Results.	142
6.4	Humanitarian response with segmentation. A total of 19,800 kg were delivered. .	145
7.1	Variable, parameter and index definitions	156
7.2	Model parameters.	168

7.3	Route itineraries in worst case scenario. Explored links in every time steps are highlighted in bold. Node numbers refer to Fig. 7.6	171
A.1	Summary of literature reviewed in Section 2. Symbols used in this table are defined in Table A.2.	229
A.2	Symbol Glossary.	233
B.1	Summary of literature reviewed in Section 3. Symbols used in this table are defined in Table B.2.	235
B.2	Symbol Glossary.	237
C.1	Percentage battery depletion with no payload.	240
C.2	Percentage battery depletion with 1kg payload.	240
C.3	Percentage battery depletion with 2kg payload.	244
C.4	Percentage battery depletion with 3kg payload.	247
C.5	Parameter Tuning – fitness deviation examines the relative difference in optimal solution with respect to $c = 0.9999$ iteration.	247
C.6	Detailed relief distribution outputs per township (without balancing).	248
C.7	Detailed relief distribution outputs per township (with balancing).	249
D.1	Particle Swarm Optimisation tuning results - averaged over 20 iterations.	250
E.1	Manual Genetic Algorithm Tuning - each experiment is repeated 1000 times.	259

List of Figures

1.1	Global deaths per year since 1995. Adapted from EMDAT (2020).	15
1.2	Humanitarian logistics structure. Adapted from Balcik et al. (2008).	16
1.3	Disruption and recovery curve after a natural disaster. Adapted from Park (2015).	18
1.4	WFP logistics operation growth (WFP Aviation 2012, 2016, WFP 2017 <i>b</i> , 2018 <i>a</i>).	19
1.5	Demand evolution of UAVs to 2050. Adapted from SESAR (2016)	22
1.6	Thesis content overview.	28
2.1	Cluster approach schematic. Agencies presented in this schematic are described in Table 2.1.	32
2.2	Cyclical Humanitarian Logistics Nature UNOCHA (2019 <i>b</i>).	37
3.1	Trajectory Optimisation Schematic.	65
4.1	Point mass model of UAV during flight.	76
4.2	The coordinate system of the trajectory optimisation problem.	76
4.3	Methodological chapter relationship schematic.	82
5.1	Framework flowchart indicating the relationship between the two decision stages (yellow blocks) in this study. Operational aspects specifically addressed in this study are highlighted in green, with inputs highlighted in blue, and external factors that influence our outputs in orange.	88
5.2	Example trajectory between the Hsin Yi and Ren Ai regions of the Nanto County in Taiwan.	93
5.3	Route assignment flowchart.	101
5.4	Model Workflow	103
5.5	Nanto County Map, with the warehouse location indicated with a blue icon.	109
5.6	Trajectory connecting Chong Liao and Ren Ai with an empty UAV.	111
6.1	Model workflow.	124
6.2	Trajectory profile.	125
6.3	Airspace modification vertices. The green and blue nodes indicate fixed and free vertices respectively. The shaded region represents an immutable no-fly zone.	128
6.4	Sequential Trajectory Optimisation. Red line area indicates general planned route volume, with blue line representing the detailed trajectory considering airborne traffic. The shaded parameters indicate that their values have been determined by the planner.	133
6.5	Dubins trajectory types. The blue line indicates the centre-line such that the distance to each border is 2 RNP. In RF, $b = r + 2\text{RNP}$ and $a = r - 2\text{RNP}$	135
6.6	Static obstacle results.	140
6.7	Dynamic obstacle results.	141

6.8	Nanto County Map, with the relief locations indicated with a blue icon, and depots shown in red.	143
6.9	Segmentation solution for static demand case.	144
7.1	Truck and UAV routing example. A dashed line indicates an unexplored link, a gray line determines that the link has been explored when the vehicle is being routed, while the green and blue indicate the route followed by the drone and the truck respectively.	155
7.2	Scenario tree example where the normal distribution is modelled using the three-point estimation method.	162
7.3	Sequential algorithm workflow	164
7.4	Chromosome representation.	165
7.5	Case study road network in Haiti. Data from OpenStreetMap.	169
7.6	Truck and UAV routing per time step.	170
7.7	Deterministic and stochastic average vehicle routing travel times with network damage. The shaded region indicates the solution range.	171
7.8	Stochastic average vehicle routing travel times with network damage under UAV support. The shaded region indicates the solution variability.	172
7.9	Average routing costs with network damage for the sequential greedy and GA solution methods.	172
7.10	Average energy use by drones for the sequential greedy and GA solution methods.	173
7.11	Deviation between perceived and actual path cost with and without drone assisted reconnaissance.	174
7.12	Deviation between chosen and best possible path traversed with and without drone assisted reconnaissance.	175
7.13	Average mission time variation with penalty parameter value.	176
7.14	Deviation between perceived and actual path cost with and without drone assisted reconnaissance in 25 node case.	177
7.15	Deviation between perceived and best path cost with and without drone assisted reconnaissance in 25 node case.	177
7.16	Deviation between chosen and best possible path traversed with and without drone assisted reconnaissance in 25 node case.	178
7.17	Routing costs with and without UAVs in 25 node case.	178
7.18	Average energy use by drones with network damage for the GA in the 25 node case.	179
7.19	Routing costs with 1 and 2 UAVs in 25 node case.	179
7.20	Routing costs with 2 and 3 UAVs in 25 node case.	180
D.1	Dynamic obstacle rapid tree simulation results	254
D.2	Static and dynamic obstacle rapid tree simulation results	255

Publications related to this work

The work presented in this thesis produced the following publications:

Peer-reviewed journal articles

- Escribano J, Goldbeck N, Hsu PY, Angeloudis P, Ochieng W (2020). Endogenous stochastic optimisation for relief distribution assisted with Unmanned Aerial Vehicles. *OR Spectrum* 42, 1089–1125.
- Escribano J, Angeloudis P, Ochieng W (2020). Optimal Hub Selection for Rapid Medical Deliveries using Unmanned Aerial Vehicles. *Transportation Research Part C: Emerging Technologies* 110, 56-80.
- Otero A, Escribano J, Angeloudis P (2020). Design of hospital delivery networks using unmanned aerial vehicles. *Transportation Research Record* 2674(5), 405-418.

Conference proceedings

- Escribano J, Angeloudis P, Ochieng W (2019). Sequential Trajectory Optimisation Using Dynamic Airspace Sectorisation. *Transportation Research Board 98th Annual Meeting (TRBAM 2019)*.
- Escribano J, Angeloudis P, Ochieng W (2018). Integrated Trajectory-Location-Routing for Rapid Humanitarian Deliveries using Unmanned Aerial Vehicles. *2018 Aviation Technology, Integration, and Operations Conference*.
- Escribano J, Angeloudis P, Ochieng W (2018). Routing and Path Improvement Heuristics for Humanitarian Unmanned Aerial Vehicles. *2018 EURO Conference*.
- Escribano J, Angeloudis P, Ochieng W (2017). Integrated Trajectory Optimisation and Location-Routing Method for Rapid Medical Deliveries using Unmanned Aerial Vehicles (UAVs). *International Conference on Dynamics of Disasters*.

Chapter 1

Introduction

Disasters are unexpected events that cause injuries, loss of lives and widespread damage and disruption to the build environment and the functioning of a community (UNISDR 2009). Only in this last century, we witnessed two of the deadliest catastrophes ever recorded. The 2004 Indian Ocean earthquake, in which 227,000 people lost their lives, and the more recent 2010 Haiti earthquake, which claimed the lives of 220,000 people and destroyed over 100,000 homes in the process (Argollo et al. 2014, Disasters Emergency Committee 2015).

While the examples above may be considered outliers in terms of their magnitude, the Centre for Research on the Epidemiology of Disasters (CRED 2015, n.d.) documented the rising mortality of disasters over the last 20 years (presented in Figure 1.1), highlighting the increasing risks to which vulnerable communities are exposed. With the growing frequency of weather-related disasters as a result of climate change and the rising population levels located at coastal settlements, the trend is expected to continue over the next 50 years (Thomas & Kopczak 2005).

The type and cause of disasters directly affect their severity and predictability (Table 1.1). Also influencing disaster severity is the economical development level of the community affected, with lower income countries experiencing 44% of disasters since 1995, yet suffering 91% of deaths over the same period (CRED 2015, n.d.). As an indicative example, the 2016 Ecuador earthquake resulted in 600 fatalities and 4,000 injured approximately, despite being of greater magnitude than the 2010 Haiti earthquake (USAID 2016).

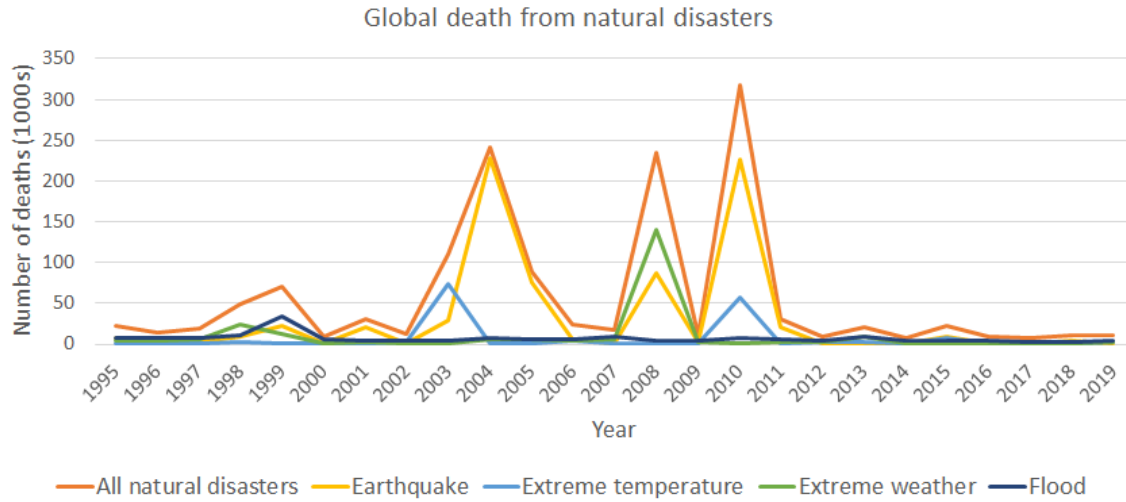


Figure 1.1: Global deaths per year since 1995. Adapted from EMDAT (2020).

Table 1.1: Immediacy of response and predictability depending on the disaster category. Adapted from Cozzolino (2012).

Disaster Attributes	Disaster Categories			
Cause of Disaster	Natural	Man-made	Natural	Man-made
Predictability	Sudden	Sudden	Slow	Slow
Relevance of Logistics	Very High	High	Medium	Low
Examples	Earthquakes, tsunami	Industrial accidents	Plagues, cyclones	Famine

Therefore, natural disasters not only pose an increasing threat to human lives, but hinder the economic development of the communities most viciously affected by them. Humanitarian response focuses on mitigating the impact of these events. While the specific operations vary based on the factors cited in Table 1.1, in general they consist of transporting essential material and personnel to the affected region, and are managed and coordinated by international actors. Hence, humanitarian response necessitates the development of large-scale logistics network to transport essential resources to affected communities.

1.1 Humanitarian Logistics

Humanitarian logistics is defined as the management of cargo flows within a humanitarian organisation, and underpins the operations aiming to alleviate the devastating effects of disasters. It is an essential process in disaster response, often determining the effectiveness of the operations,

and accounts for a significant proportion of the expenditure of aid organisations (Tatham 2011, Grafham & Lahn 2018).

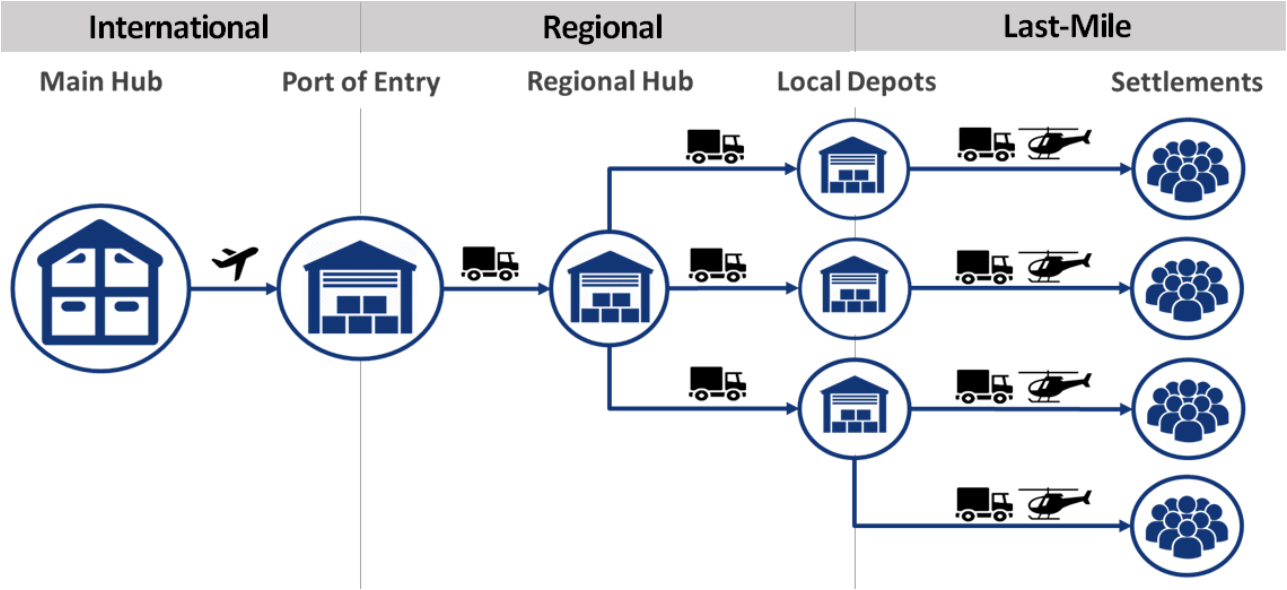


Figure 1.2: Humanitarian logistics structure. Adapted from Balcik et al. (2008).

The humanitarian logistics sector is generally classified into two categories: relief and rehabilitation, and recovery activities (Holguín-Veras et al. 2012). The former focuses on very short-term evacuation and rehabilitation activities by providing immediate essential resources (e.g. food, shelter, water) to the affected communities. The latter addresses the long-term reconstruction of lifeline systems such as water, gas, electricity, or transportation networks. While the architecture of the supply chains in both stages are similar (Figure 1.2), key differences can be found in their planning horizons (short-term or long-term), data quality, uncertainty and delay implications.

Despite the short-term nature of relief efforts (shown in Figure 1.3), the importance of this stage cannot be underestimated, as survival rates reduce exponentially during the first 72 hours after a disaster event (Huang & Lien 2012). To illustrate the importance of fast deployments in humanitarian operations, the World Food Programme¹ (WFP) can deploy response teams within 24 hours through the Immediate Response Account (WFP 2018b). Furthermore, the speed and effectiveness of these operations directly affect the long-term recovery and sustainability of the

¹WFP is the United Nations (UN) agency that acts as lead of the Logistics Cluster, which coordinates, manages and facilitates access to logistics services. WFP also provides logistics services to other humanitarian organisations as last resort (WFP 2019).

affected community, given that longer-term recovery operations build upon initial social and infrastructural improvements during relief efforts (CWGER 2008).

Relief operations are structured as push-based logistics strategies, where the design of the supply chain is informed by forecast demand. This process involves the overestimation of demand to compensate for the unreliability of data in the aftermath of a disaster. Such an approach requires high inventory levels and often results in high wastage and operational cost levels (Meier et al. 2016).

Ideally, information on demand quantity and location would be gathered before generating the order, reducing the uncertainty in the process and minimise wastage. These supply chains are said to be pull-based, and require agile processes that can adapt to changing demand patterns.

A significant proportion of the uncertainty associated with humanitarian logistics operations resides in the last-mile distribution process (refer to Figure 1.2). This is a result of damaged or inadequate transport infrastructure, which has the potential to isolate communities and pose a challenge to relief cargo deliveries. Notable examples can be found in the Peruvian Amazon, where communities rely solely on river transport that increases travel times to days (Meier & Bergelund 2017), and Papua New Guinea, where the rainy season cause dirt roads to become muddy delaying vehicles significantly (Meier & Soesilo 2014).

Armed conflicts present a further safety hazard to volunteers due to accidental or intentional damage. Recent examples include a major attack on a UN aid convoy in Syria on October 2016 (BBC News 2016).

Another key concern within humanitarian operations is the access to reliable and cost-effective energy sources. Diesel fuel still remains an essential resource in humanitarian operations, being required for electricity generation and for transportation fuel. According to Grafham & Lahn (2018), vehicle fuel consumption during humanitarian operations at Jordan, Kenya and Burkina Faso in 2015 and 2016 exceeded 600,000 yearly litres on aggregate. Motor inefficiency, poor maintenance, decentralised transportation, and low fuel quality are the leading causes for high diesel wastage. Sustainability targets outlined by numerous humanitarian organisations (UNICEF 2015, Grafham & Lahn 2018) provide further evidence for the need to reduce the

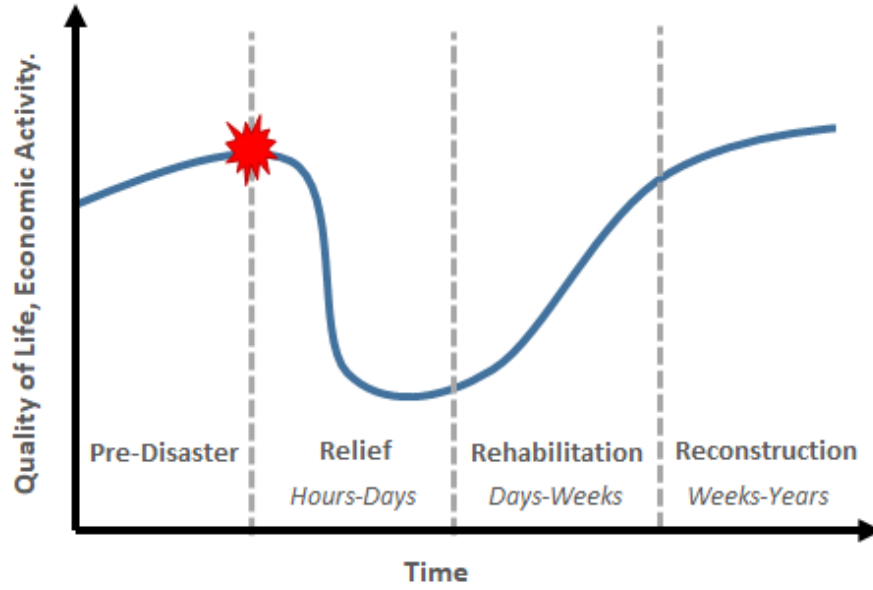
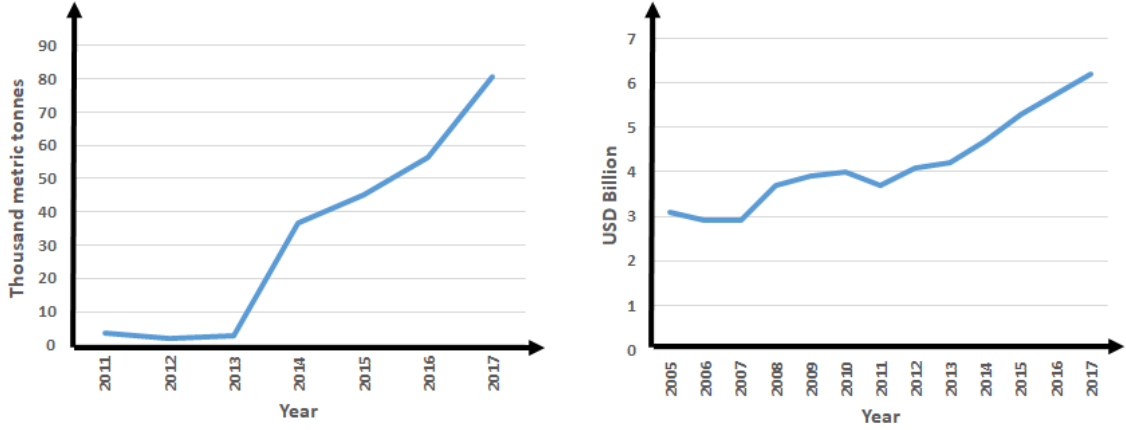


Figure 1.3: Disruption and recovery curve after a natural disaster. Adapted from Park (2015).

reliance on diesel fuel for transportation.

The factors outlined thus far contribute to the uncertainty, delays and danger that characterise relief operation which yield high wastage, time and economic losses. In response, humanitarian organisations increasingly rely on aerial delivery, which overcome road infrastructure limitations and reduce travel time uncertainties. As an indicative example, WFP’s Aviation division have increased total cargo delivery by 60% between 2011 and 2015, rising their total expenditure by 70% over the same period (WFP Aviation 2012, 2016). Moreover, WFP’s expenditure doubled over the 2005-2017 period to a record US\$ 6.2 billion (Figure 1.4).

The prohibitive operating costs hinder any increase in the scale of air transportation operations. In addition, a major concern within the humanitarian sector is the lack of human capital, with a limited pool of readily deployable skilled personnel (Tomasini & Van Wassenhove 2009). As a result, the humanitarian organisations are considering the use of Unmanned Aerial Vehicles (UAVs) to assist humanitarian operations (Meier et al. 2016).



(a) WFP Aviation cargo transportation.

(b) WFP costs fluctuation.

Figure 1.4: WFP logistics operation growth (WFP Aviation 2012, 2016, WFP 2017b, 2018a).

1.2 Overview of UAV Technology

UAVs are aircraft where the pilot is not onboard, thus the aircraft may be controlled remotely or fly autonomously. Given the absence of an onboard pilot, UAVs encompass a wide range of designs and weights, ranging from micro aircraft of few kilograms to passenger carrying platforms. While traditionally developed for military applications since the 1900s, UAVs are increasingly being integrated in a variety of industries, including delivery, photography, and security among others.

The recent improvements in UAV technology and reductions in manufacturing costs, combined with their ability to overcome most infrastructure and geographical challenges, make UAVs particularly suitable for cargo delivery. For these reasons, UAVs are being considered to facilitate last-mile distribution operations in both commercial and humanitarian sectors.

Efforts to deploy UAVs for delivery have been taking place since the foundation of Matternet² in 2011. Several organisations have announced or embarked on projects involving large-scale UAV deployments, with notable UAV initiatives presented in Table 1.2.

UAV capability and performance are influenced by their weight and body structure - a classification of different UAV types is presented in Table 1.3. In terms of body structure, three main categories exist: rotary-wing, fixed-wing, and hybrids. Rotary-wing systems operate

²Matternet is a leading company in the UAV urban delivery industry.

Table 1.2: UAV projects for delivery and passenger transport.

Industry	Drone	Organisation	Country	Status	Reference
Passenger	Rotors	Airbus	Brazil	Testing	Lewis & Magalhaes (2018)
	Rotors	Uber	US	Testing	Holden & Goel (2016)
	Hybrid	Kitty Hawk	NZ	Testing	Sorkin (2018)
	Rotors	Volocopter	Dubai	Testing	Moon (2018)
Delivery	Hybrid	Amazon	US/UK	Testing	Keach (2018)
	Hybrid	Google	US	Testing	X Development (2018)
	Hybrid	DHL	Germany	Ongoing	DHL (2015)
	Rotors	Toyota	Canada	Testing	Reagan (2018)
	Rotors	Flirtey	US	Testing	Hidalgo (2019)
	Rotors	Matternet	US	Testing	Bogaisky (2019)
Humanitarian	Fixed	WeRobotics	Peru	Completed	Meier & Bergelund (2017)
	Hybrid	WeRobotics	DR	Testing	Meier (2018)
	Hybrid	Zipline	Rwanda	Ongoing	Simmons (2016)
	Hybrid	RedLine	Rwanda	Planned	Afrotech EPFL (2016)
	Rotors	UNICEF	Malawi	Completed	UNICEF (2016)
	Rotors	MSF	PNG	Ongoing	Meier & Soesilo (2014)

Country codes: Dominican Republic (DR), New Zealand (NZ), Papua New Guinea (PNG), United Kingdom (UK) and United States (US).

similarly to helicopters and are capable of vertical take-off and landing (VTOL), and fixed-wing resemble conventional aircraft and are propelled by persistently exerting forward thrust. While rotary UAVs provide higher degree of control, fixed-wings possess greater endurance, speed and range. Hybrid systems combine the speed and range of fixed-wings with the greater control capabilities of rotary-wings, but are generally more expensive. Within the delivery, passenger transport, and humanitarian sectors, the most common UAV types are multi-rotors and hybrid systems due to the increased level of control provided and VTOL capabilities (see Table 1.2).

Since UAVs require little supporting infrastructure to operate, the initial capital investment required is lower compared to conventional transport modes. As an example, Matternet estimated that creating a drone network covering 150 square kilometres consisting of 50 stations and 150 vehicles would cost around 0.9 USD million (Raptopoulos 2013).

The advantages of UAVs are not only economical — UAV delivery networks can be established quickly, and require a small workforce to operate. Small UAVs, for example, require a single pilot for operation (in most cases only for monitoring purposes), allowing for fast deployments with minimal supporting infrastructure. Furthermore, since UAVs are remotely controlled or autonomous, they remove safety risks to humanitarian workers associated with aid convoy

delivery.

Table 1.3: UAV classification. Adapted from Ren, Castillo-Effen, Yu, Yoon, Nakamura, Johnson & Ippolito (2017). LTA - Lighter-than-air, HTA - Heavier-than-air

Classification			Description
LTA	Airship		Engine-driven lighter-than-air aircraft.
HTA	Fixed Wing	Glider	Not dependent on an engine for sustained flight.
		Airplane	Engine-driven propulsion, with wing generating lift.
	Rotor Craft	Helicopter	Engine-driven rotors with pitch and roll control based on cyclic pitch.
		Multi-rotor	Engine-driven rotors with pitch and roll control based on differential lift from multiple rotors.
	Hybrid		Engine-driven aircraft capable of vertical takeoff and landing, with flight dependent on forward engine thrust.
	Other		Any other aircraft configuration.

Within humanitarian applications, UAVs have been increasingly used since Hurricane Katrina in 2005 (American Red Cross 2015). While mostly used for reconnaissance and imagery, UAVs are being tested to deliver relief and assess road-damage (Meier & Bergelund 2017, DuPlessis 2016). WFP seeks to facilitate their deployment by developing a coordination system that aims to reduce redundancy, improve safety, and increase standardisation during UAV deployment (ETC 2017, Vornic 2017).

1.3 Drone regulations

Only in Europe, the UAV market is expected to increase its expected value to €10 billion annually by 2035 and €15 billion by 2050 (SESAR 2016). Commercially, over 415,000 UAVs are expected to be in operation 2050, with an additional 7 million estimated for leisure. The vast applicability of UAVs requires the development of robust regulatory frameworks that ensure safe and secure operations.

By 2017, 65 countries issued national UAV regulations that presented a set of similarities in terms of categorisation, scope, and limitations. The main categorisation system is based on the UAV's take-off weight and its purpose as a surrogate of impact risk (Stöcker et al. 2017). As a result, regulations impose greater limitations and requirements to heavier UAVs. Some of these

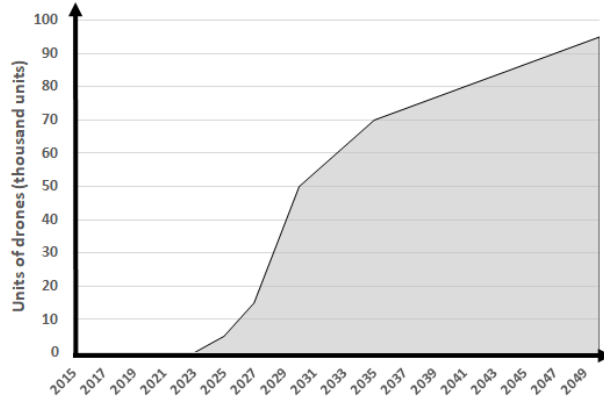


Figure 1.5: Demand evolution of UAVs to 2050. Adapted from SESAR (2016)

rules have a negligible impact to the adoption of UAV in certain applications, like registration requirements, pilot certifications, and occasionally altitude boundaries.

However, most regulations limit commercial UAV use to visual line of sight (VLOS) applications with few exceptions (Stöcker et al. 2017). In particular, the United States Federal Aviation Agency (FAA) issued these rules under FAR Part 107, with exemptions given on an individual basis under FAR Part 333 (FAA 2016, 2012). In Europe, all member states adhere to the principles set by the European Aviation Safety Agency (EASA), which classifies beyond visual line of sight (BVLOS) operations in the Specific category, requiring explicit authorisation from the relevant national airspace authority (EASA 2017*a*).

The regulatory structure remains a key challenge towards the development of the BVLOS UAV market, mainly consisting of on-demand automated operations, such as delivery. According to the Single European Sky Air Traffic Management Research (SESAR), BVLOS operations will constitute over 75% of all UAV operations by 2050 (SESAR 2016). In the humanitarian field, most desired applications require BVLOS approval (road assessment, monitoring, delivery).

Current regulations can enable safe operations in low spatial density and BVLOS may take place using segregated airspace and manual procedures. Such a system cannot be applied for large-scale on-demand services such as UAV delivery, which would operate in BVLOS conditions.

1.4 Statement of the Problem

The field of humanitarian logistics comprises an essential process in response operations taking place immediately after a disaster event, where the aim is to provide relief and alleviate suffering of the affected population. Thus, humanitarian logistics is fundamental to the rehabilitation of the community.

Given the time-related constraints and highly stressful environment in post-disaster contexts, errors in judgement become a significant concern that may preclude the success of the humanitarian operation (Campbell & Clarke 2018). Mathematical optimisation is therefore a fundamental tool to aid respondents during the decision-making process by providing them with a palette of solutions whose performance is measured using agreed metrics.

An extensive literature exists that focuses on humanitarian logistics and operations research, outlined in Chapter 2. However, most of this literature is applied to conventional methods of transportation, such as aid convoys, trucks or helicopters.

Although some of the modelling techniques are transferable to UAV delivery systems, they fail to capture the complexities of small battery-powered UAVs. For example, transporting 2 kg requires on average a 3 metre wingspan and a total empty mass of 5 kg (Vertical Technologies 2017). Due to their limited payload capacity, UAVs would transport high-value low-volume cargo, such as medicines and life-saving cargo.

The major advantage of using small-battery powered UAVs compared to ground vehicles lies in their deployment speed — 75 km range hubs are deployable in less than 24 hours (Meier et al. 2016). Moreover, the autonomous nature of the technology results in lower safety risks to humanitarian workers.

The systemic effect of UAV technologies to the nature of humanitarian logistics is yet unexplored, as new constraints are expected to emerge despite the elimination of some dependencies of conventional transport. As an indicative example, major simplifications to UAV range are commonly found in the literature, which may result in severe overestimation of UAV capabilities. Furthermore, various limitations of UAVs require the concept of operations to consider split delivery, recharge, energy management and allocation.

Equally important are the challenges associated with the introduction of UAVs in non-segregated airspace. As discussed in section 1.3, recent regulatory frameworks require airspace segregation for UAV operation considering their size and capabilities. While the approach is appropriate for VLOS operation, extensive BVLOS operations are not feasible under the current system (Stöcker et al. 2017). Exceptions are allowed through time-consuming ad-hoc procedures, precluding the use of UAVs for disaster relief operations as they require immediate deployments.

However, relaxations of these rules could result in significant safety risks due to the high level of airspace congestion. The main challenge relates to the large number of vehicles sharing the airspace simultaneously, which increases the problem size when developing non-conflicting trajectories. UAV operations must account for the trajectories of other vehicles and their capabilities. While obstacle avoiding trajectories have been developed, they are yet to be integrated with routing algorithms nor applied to humanitarian logistics.

Thus, the research gaps that were identified during the literature review survey and motivate this study are summarised as follows:

- Gap 1. Lack of understanding of the ways that complex energy consumption patterns are affected by flight stages, payloads, and aerodynamics in UAV routing literature. Given the short range of UAV platforms, inaccuracies in battery consumption estimation will have significant effects on UAV operational performance.
- Gap 2. Energy allocation, energy management, and recharge has yet to be explored in the context of supply chain design. The design of extensive and uninterrupted UAV-based delivery services require the provision of battery replacements. As such, the location of recharge stations and allocation of excess batteries must be incorporated into the mathematical formulation.
- Gap 3. A quantitative framework for UAV-based humanitarian relief performance has not been developed to date. Current models for UAV-based urban and humanitarian logistics focus on the routing problem as a single-depot VRP. This simplification ignore hub selection decisions in spite of the UAV range constraints, which limit the applicability of these approaches to small case studies. Additionally, as UAS will

act within managed airspace, Airspace Traffic Management should be considered.

Gap 4. Lack of UAV trajectory management systems for uncertain/dynamic environments.

UAV demand is expected to be highly variable and short-term, suggesting that the creation of fixed airspace corridors will be impractical. Therefore, the development of tactical trajectory planning becomes a necessity for any UAV-based delivery model, as it will affect delivery times, battery charge, and overall system efficiency.

Gap 5. Compliance with existing UAV regulatory frameworks in humanitarian operations.

SESAR envisions a system that limits access to airspace classes based on the existing Performance-Based Navigation (PBN) framework (SESAR 2019). This consists on the provision of Required UAV Navigation Performance (RUNP), which are similar to Required Navigation Performance (RNP) of conventional Air Traffic Management (ATM).

1.5 Aims and Objectives

The aim of this study is to develop a holistic quantitative framework for the design of UAV-based humanitarian response operations, including relief delivery and evacuation. In doing so, the thesis incorporates aspects of humanitarian preparedness (warehouse location planning and inventory management), UAV platform capability, and trajectory management. Figure 1.6 presents an overview of the topics researched in this thesis.

The proposed framework serves as a decision-support tool for humanitarian organisations that allows for the optimal planning of last-mile relief supply chain design assisted by UAVs. The study considers mission duration, energy provision, safety and total cargo output as key performance indicators. The following research objectives are defined:

Obj. 1. Develop a quantitative framework for measuring UAV-based relief distribution which considers ethical principles including equity and fairness (Bagshaw 2012). Further UAV-specific features are incorporated that relate to safety, energy use and access, which are implemented using complex objective functions.

- Obj. 2. Propose a last-mile relief distribution design algorithm for UAV-based humanitarian relief operation. The algorithm optimises key decisions of hub location and routing, and reflects the sequential decisions made during humanitarian operations, thus considering uncertainty and imperfect data (Boyles & Waller 2010, Özdamar et al. 2014).
- Obj. 3. Develop trajectory design algorithms with accurate energy expenditure estimations. While the linearisation of fuel consumption relationships is reasonable for road vehicles, this assumption misrepresents the effectiveness and operational range of UAVs, particularly when consideration variations due to weather and payload sizes. Hence, the nonlinear behaviour of battery use is no longer negligible, and have to be considered in planning.
- Obj. 4. Test and calibrate the proposed framework in large-scale scenarios. The case studies proposed are based on realistic disasters in scale and scope, which allows us to compare the effectiveness of UAV-based response with traditional vehicles.

1.6 Outline

This chapter has established the motivation to undertake the proposed study, and has provided the background information necessary to understand the implications, objectives and expected contributions. The remaining parts of this thesis are organised as follows:

The literature review is separated in two chapters. Chapter 2 provides a survey on the humanitarian operations research, with particular focus to post-disaster response literature. The chapter further reviews VRP-related literature, which is commonly used to define humanitarian logistics problems. Chapter 3 focuses on the UAV operations aspect, the regulatory framework, and requirements to integrate the UAV constraints into the humanitarian context.

Chapter 4 presents the vehicle dynamics of UAVs and the theory of optimisation that will be referred to throughout this thesis. A selection criterion is developed to determine the appropriate solution method to use depending on the problem instance, as well presenting the relationship between the remaining chapters in the context of humanitarian operations.

Chapter 5 develops the humanitarian operation model framework which includes a LRP that considers energy management, allocation, and warehouse selection to optimise UAV-based humanitarian response system. The algorithm is evaluated using on a case study based on the 1999 Chi-Chi earthquake, which affected the central region of Taiwan. Key performance indicators that measure the speed and equity of the response, quantity of material transported, and overall cost are explored.

Chapter 6 presents the drone operations model framework, composed of an airspace separation algorithm, and a sequential trajectory algorithm process. The chapter further describes how the proposed framework was developed and validated.

Chapter 7 describes a novel formulation for the distribution of relief assisted by UAV network damage assessment. The problem is formulated endogenous stochastic optimisation model, where the decisions made by the UAVs exploring the network and the ground vehicles delivering aid have affect the network uncertainty parameters during the operation time horizon.

Chapter 8 summarises the findings, provides a list of conclusions and future research that can build upon this work.

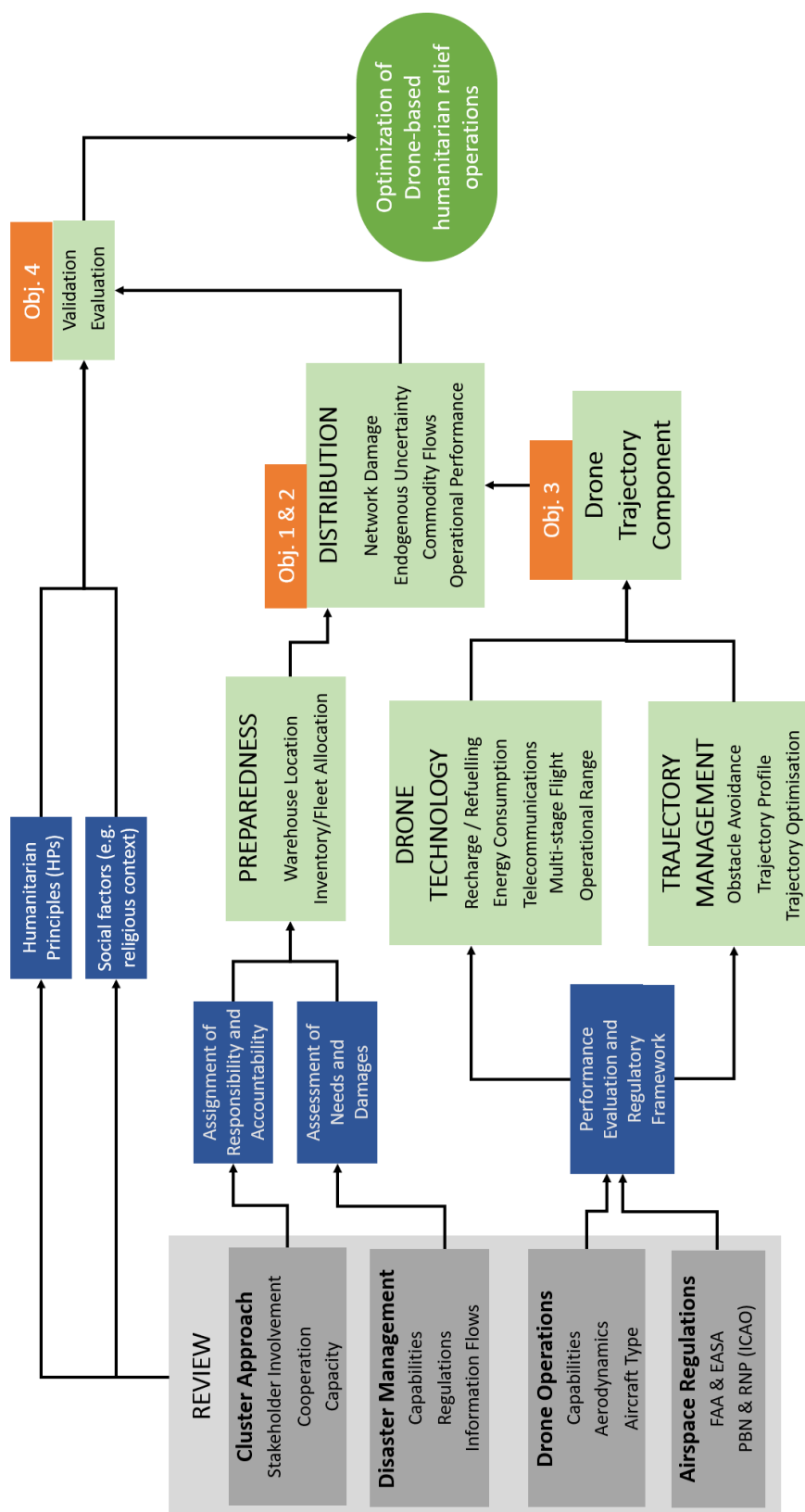


Figure 1.6: Thesis content overview.

Chapter 2

Current Approaches for Humanitarian Operations

Humanitarian logistics underpin the essential life-saving operations that take place during post-disaster missions. While the fundamental processes of humanitarian logistics coincide with those of commercial applications, significant differences that relate to the objectives, constraints and overall context preclude the adoption of commercial models for humanitarian operations.

This section reviews the state-of-the-art of post-disaster response operations. In particular, this survey aims to:

1. Identify the roles and responsibilities of humanitarian stakeholders. Determining which organisations are responsible for the planning and management of a humanitarian operation.
2. Analyse the common mathematical framework used to formulate post-disaster response optimisation problems, including relief distribution and evacuation. This includes reviewing commercial applications and novel technologies (UAVs) that can be integrated into humanitarian logistics research, considering the differences in objectives, contexts, and constraints.
3. Review the research on humanitarian operations that have a direct effect on post-disaster missions and can be integrated into the post-disaster formulation framework. This includes preparedness and rehabilitation operations that take place before and after response,

respectively.

This chapter is structured as follows: Sections 2.1-2.4 focus on describing the context of humanitarian operations, while Section 2.5 delves into the operational research concepts that are adopted in formulating preparedness, response and rehabilitation missions. In particular, the review analyses the Facility Location, Vehicle Routing Problems, and Traffic Assignment Models in detail.

The assessment suggests that the Location-Routing Problem is an adequate mathematical framework to model post-disaster response operations, and is capable of incorporating constraints relating to the humanitarian objectives, the uncertainties pertaining humanitarian contexts, and the design of supporting infrastructure. The latter is particularly important when considering novel UAV technology that possess limited operating range.

However, the review highlights that research focusing on UAV applications commonly ignore or overly simplify trajectory design. For this reason, an extensive literature review that aims to determine the key features to adopt is presented in Chapter 3.

2.1 Origins of Humanitarian Logistics

Initial forms of humanitarian response have existed since the nineteenth century in the context of colonial and religious practices, including famine relief, and health care service. Its institutionalisation can be traced back to the adoption of the first Geneva Convention in 1864 that considered the provision of relief to the wounded of war (Davey et al. 2013).

The adoption of the UN Charter (1945) after the conclusion of World War II preceded the creation UN Relief and Rehabilitation Administration, which later was distributed into specialised agencies: UNICEF, Food and Agriculture Organization (FAO), World Health Organisation (WHO) and the International Refugee Organisation - precursor of the United Nations High Commissioner for Refugees (UNHCR) (Paulmann 2013, Rysaback-Smith 2015). United States Agency for International Development (USAID), Oxfam, International Federation of Red Cross and Red Crescent Societies (IFRC) and Doctors Without Borders (MSF) comprise other notable Non-Governmental Organisations (NGOs) developed over the same period.

Institutionalisation brought further progress in humanitarian law: the Universal Declaration of Human Rights was adopted, and the Geneva Convention (1949) was expanded to protect non-combatant civilians (United Nations 1949). Further protections to civilians were established through the adoption of Protocol I and II in 1977, now supported by the Office of the High Commissioner for Human Rights (OHCHR) (United Nations 1977, OHCHR 2019).

The increasing capacity and number of NGOs active in humanitarian response lead the UN to develop coordination mechanisms for the effective and efficient deployment of humanitarian operations. This is the subject of the next section.

2.2 Humanitarian Strategy

Current humanitarian operations are governed by the cluster approach, which was established in 1991 through the adoption of the resolution 42/182 by the UN General Assembly (UNOCHA 2012). The document divides the humanitarian competencies into eleven different clusters (see Figure 2.1a), and designates a central coordinator: the ERC and OCHA (UNOCHA 2012). Table 2.1 presents the role of coordinating agencies, with Figure 2.1b showing the coordination architecture.

The main objective of the cluster approach is to improve predictability, timeliness, effectiveness, leadership and accountability within humanitarian response. To achieve this, lead organisations are designated for each cluster, with strictly defined responsibilities. In addition, global coordination forums are established to improve cooperation and coordination between clusters.

At the global level, the lead organisation assumes the role of coordinator, consolidating guidelines and providing operational support to other organisations. At a country level, the cluster mobilises resources, establishes links with donors, enables inter-cluster coordination and cooperation, and interacts with local authorities, among other responsibilities.

The implementation of the clusters at a country level may correspond to any of the eleven global clusters, or some may be combined to adjust to the specific needs of the disaster. A key feature of the approach is the assignment of accountability: global leads are accountable for

ensuring preparedness and technical capacity within their cluster, and at country level, leads act as providers of last resort - they must commit to ensure adequate response to the extent possible.

The evaluation of the cluster approach found evidence suggesting that it improved response operations in terms of leadership predictability (Steets et al. 2010). However, despite receiving increasing support from NGOs, improvements to the cluster approach are yet necessary in terms of effectiveness, efficiency, and support of global clusters to country level organisations, which should include training, capacity, and guidance. Finally, through the resolution 70/1 by the UN General Assembly, the UN adopts 17 Sustainable Development Goals to which all clusters must adhere (UN General Assembly 2015).

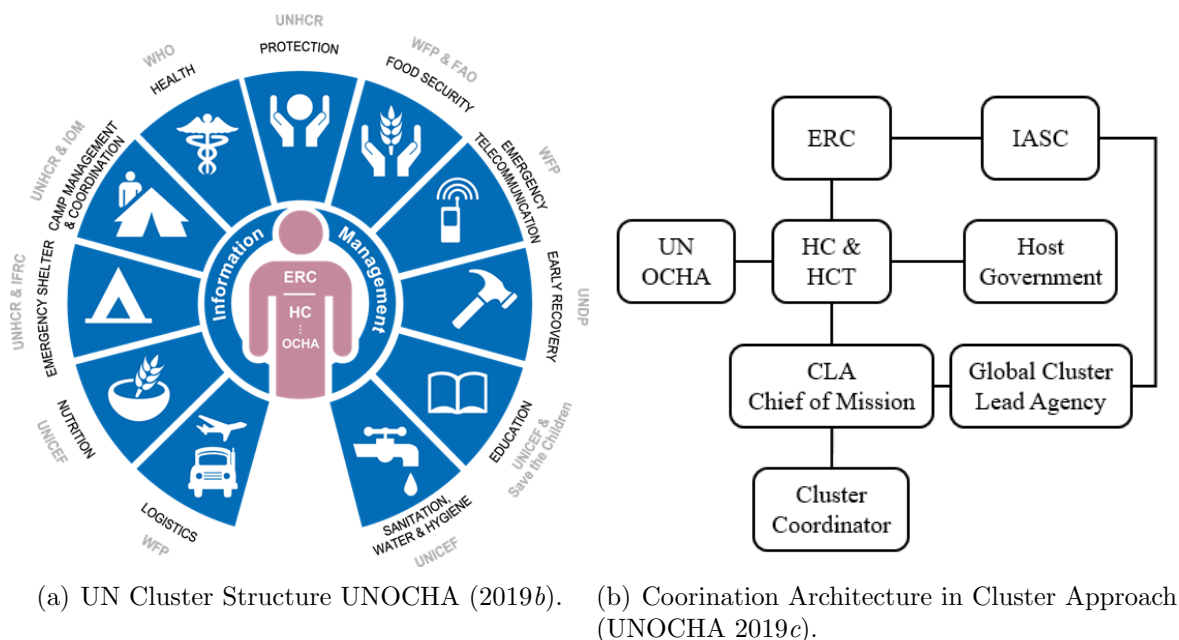


Figure 2.1: Cluster approach schematic. Agencies presented in this schematic are described in Table 2.1.

Table 2.1: Cluster Coordination Agencies UNOCHA (2019c).

Agency	Role Description
Emergency Relief Coordinator (ERC)	Oversees all emergencies with UN humanitarian operations and assigns a HC. Leads the IASC and ensures its approval of the HC selection.
Inter-Agency Standing Committee (IASC)	A forum formed by humanitarian partners that develops policy, assigns responsibilities, and addresses gaps in the humanitarian operation.

Office for the Co-ordination of Humanitarian Affairs (OCHA)	Facilitates coordination between clusters, as well as offers support and guidance to the HC and HCT.
Humanitarian Co-ordinator (HC)	Assesses the need for humanitarian assistance for a disaster, as well as organises response operations.
Humanitarian Country Team (HCT)	A forum composed by the UN, International Organization for Migration, international NGOs established by the HC that develops solutions to common strategic issues during humanitarian response.

2.3 Humanitarian and Commercial Logistics

While the key characteristics of humanitarian and commercial logistics are aligned, their context and objectives are distinctively different. Therefore, adopting commercial logistics frameworks to model humanitarian operations will fail to capture its complexities, despite the fundamental theorems and methodologies being transferable from one to the other (Thomas & Kopczak 2005, Van Wassenhove 2006, Beamon & Balcik 2008, Holguín-Veras et al. 2012).

The main distinction between both relates to their objectives. The aim of humanitarian missions is to alleviate suffering, which is quantified as a deprivation cost - a measure of the well-being of a community. The definition of deprivation costs is an active area of research, as the services provided during humanitarian logistics are often intangible (Holguín-Veras et al. 2012). In contrast, commercial logistics focuses solely on logistics costs and customer service.

A further differentiating factor is the origin of supply (Holguín-Veras et al. 2012). In commercial scenarios, commodity generation is controlled by the supplier and thus is regular. Instead, humanitarian organisations rely on en-mass donations by governments and organisations. A significant portion of this material is unsolicited and becomes counter-productive to the mission as NGOs must divert their scarce resources to manage and dispose unwanted products. (Van Wassenhove 2006, Ülkü et al. 2015).

Similarly, uncertainty in demand forecasting is greater in post-disaster contexts as they are derived from extreme events, so are unpredictable in terms of location, size and timing. The problem is further accentuated due to the lack of lead times between the occurrence of the

demand and the time supplies are needed, meaning that there is a very short time window to survey the affected community and understand their needs and the extent of damage (Van Wassenhove 2006).

The stakeholders have generally homogeneous interests in commercial contexts, and decision-making is relegated to small groups of people (Van Wassenhove 2006, Thomas & Kopczak 2005). Therefore, the decision-making structures and information systems are generally mature. Despite the existence of coordinator organisations (OCHA) and lead logistical support services (WFP), post-disaster logistics generally involve many different organisations, each with different supply chains, sizes, and aims. The result is a highly dynamic, heterogeneous environment that is difficult to coordinate and monitor. A more detailed description of the stakeholders is provided in Section 2.4.

Finally, post-disaster logistics operates on unstable networks, where multiple links may be severely damaged or blocked. The state of the network is generally unknown, which contributes to the high uncertainty in travel times, and risks posed to humanitarian workers (Holguín-Veras et al. 2012, Van Wassenhove 2006, Beamon & Balcik 2008).

Table 2.2: Humanitarian and Commercial Logistics Differences.

Category	Humanitarian	Commercial
Objective	Alleviate suffering.	Increase profits
Supply	Uncontrolled and uncertain.	Regular and controlled.
Demand	Unpredictable in size, location and timing.	Predictable and cyclical.
Stakeholders	Large in number, diverse in size, capability and objectives.	Small, with common objectives.
Network State	Severely damaged, the degree and location of which is unknown.	Robust and resilient.

2.4 Stakeholders in Humanitarian Logistics

The number and type of stakeholders involved in humanitarian relief differ significantly from those in commercial logistics. While in commercial logistics few stakeholders share a common goal under a regulated and controlled environment, humanitarian stakeholders have a diverse set

of objectives, capabilities, constraints and sizes, despite sharing the goal of alleviating suffering. The actors involved are summarised in Table 2.3.

International Relief Organisations will generally act under the United Nations cluster structure (described in section 2.2) which coordinates the response of other international and local organisations, and the military if deployed. However, the coordination of the many relief organisations is complex given the existence of communication barriers relating to culture diversity, the infrastructure deficiencies due to widespread damage, the politically volatile environment, and time pressure (Van Wassenhove 2006).

Local relief operators are fundamental in providing fast response given their geographical advantage. Their importance is being increasingly acknowledged by the humanitarian community, with the World Humanitarian Summit officially promoting localisation through the establishment of formal relationships and commitments with organisations and donors in 2016 (Ponda 2008, Sumaylo 2017, Barbelet 2018). Other advantages of local NGOs include lower structural and labour costs, as well as lower communication and cultural barriers.

The host government is ultimately responsible for disaster response, and must issue approval for international relief operators to intervene, which may incur in additional delays. Additionally, the government manages access to key infrastructure, such as airports, seaports, among others (Kovács & Spens 2007). Foreign governments may support recovery operations through financial or in-kind donations, or by providing consultant or expert staff for reconstruction projects (PAHO & WHO 2001).

The media provides an essential role to raise awareness and to inform the population of potential dangers, active evacuation procedures and best practices (Robinson 2000). Eftekhari et al. (2016) reported a positive correlation between the amount of media exposure NGOs receive and the quantity of donations obtained. However, the same study highlighted that excessive media attention may preclude coordination between organisations.

Donors provide the bulk of funding that humanitarian organisations require to finance their operations. Donations may comprise of cash or supply donations, any of which may be earmarked, restricting its spending to a specific activity or response operation. A major problem in humanitarian response is the presence of unsolicited donations, which frequently

cause bottlenecks in the supply chain and produce debris (Van Wassenhove 2006, Ülkü et al. 2015). In case of insufficient funding, NGOs may have access to common funds established by OCHA (UNOCHA 2015, 2019a).

The private sector is a major contributor to the humanitarian community by providing in-kind donation of products and services, or by partnering with specific humanitarian organisations. In particular, logistics service providers are important actors in relief supply chains, supporting NGOs and governments (Stirk 2014).

The military provides rapid deployment and additional capacity for humanitarian organisations (Wolfson & Wright 1995, Barry & Jefferys 2002, Tatham 2011). However, the humanitarian principles may be perceived endangered if organisations cooperate with international military. Peace-building initiatives may be viewed as coercive among beneficiaries, and may jeopardise humanitarian missions. Apart from common operations such as relief distribution or search and rescue, the military may be required for protection or to ensure freedom of movement in emergencies involving armed conflict.

Finally, beneficiaries are those recipients of humanitarian aid and have little control of the humanitarian supply chain.

Table 2.3: Humanitarian Stakeholder Levels.

Scale	Stakeholders
Local Level	Local NGOs
National Level	Host Government, Military
International Level	International NGOs, Foreign Governments
Other	Donors, Private Sector, Media, Beneficiaries

2.5 Humanitarian Logistics Stages

Despite the existence of formal international humanitarian organisations since the 1950s (refer to Section 2.1), the field of humanitarian logistics research is novel, with few papers being published before 2005 (Kovács & Spens 2009). However, contributions to the field have grown significantly ever since (Banomyong et al. 2017).

Humanitarian logistics literature includes all activities that focus on alleviating and reducing

the impact of disasters (Kovács & Spens 2007, Khan et al. 2008). The operations are structured as a cyclical process under the disaster risk management framework (see Figure 2.2), which highlights the inter-dependency between the stages. These are as follows:

- Disaster: An event that endangers human lives, and damages essential infrastructure.
- Response: Involves immediate procedures to alleviate suffering, which occur on the first few days after the disaster. Examples include search and rescue or disaster relief distribution.
- Rehabilitation: Focuses on restoring essential lifeline services such as transportation, energy or water. The process may take weeks to months.
- Reconstruction: Constitutes long-term processes that aim to return services to business-as-usual.
- Prevention: Concerns the development of risk-prevention plans to reduce the damage to a community in case of a disaster.
- Mitigation: Aims to minimise the impact of future disaster events. Examples include the development of vulnerability analysis or earthquake building codes.
- Preparedness: Involves activities that intend to reduce the time to respond to a disastrous event. Common examples include pre-stocking, warning systems and evacuation trials.

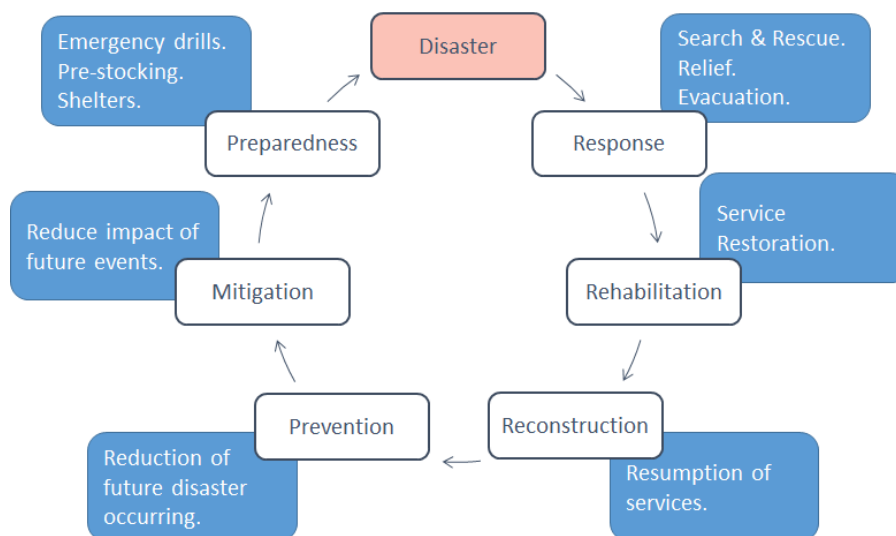


Figure 2.2: Cyclical Humanitarian Logistics Nature UNOCHA (2019b).

This review focuses on the response stage of humanitarian relief, which is characterised by its urgency and immediacy. In these situations, humanitarian respondents are heavily influenced by stress and emotional responses, which can lead to biased assessments of the environment and rushed decision-making (Campbell & Clarke 2018). Therefore, the development of algorithms that assist operators in the decision-making processes are essential to the success of the humanitarian operation.

As the operations during the preparedness phase have a direct impact on the relief distribution problem, these are also analysed. Furthermore, aspects of infrastructure repair are incorporated into this review, as these quantify network accessibility and damages caused by the disaster.

2.5.1 Preparedness Phase

The preparedness phase includes all activities related to the planning of humanitarian operations before a disaster. Common operations in this stage involve the estimation of damages and needs, which are used to determine the type and quantity of goods required and the potential distribution locations.

This process is essential to the survival of the injured population, as needs assessments generally require 72 hours to be produced (WFP 2018c), which represents the time period of highest survival rates in response operations (Huang & Lien 2012). In this stream of research, Vecere et al. (2017) developed a predictive model to estimate the number of displaced people likely to require sheltering and temporary housing. Likewise, Song & Yan (2016) proposed an evacuation demand estimator using demand curves.

An alternative approach is to estimate the needs requirements by assessing the disaster damages, which vary in terms of scale, time horizon depending on the type of disaster. As indicative examples, Battarra et al. (2018) developed demand estimation approach for earthquakes, while Kung et al. (2012) created a damage prediction model for debris-flow disasters. More recently, social media activity has been identified as a proxy for damage assessment by Kryvasheyev et al. (2016). Similarly, Nguyen et al. (2017) estimates damages using social media imagery data.

The results of the models described above are used as inputs to design the humanitarian supply chain based on demand scenarios probabilities, with the objective to maximise coverage of

service or ensure a quality threshold. The mathematical formulation of this problem corresponds to the Resource Allocation and Hub Location Problems.

Hub Location and Resource Allocation

The Hub Location Problem (HLP) seeks to determine feasible and optimal geographical location of distribution warehouses based on estimated operation costs. While the basic problem solely considers warehouse location, it is commonly combined with the Resource Allocation Problem to calculate the amount and type of commodity at each warehouse. The resultant problem is described as the Hub Location-Allocation Problem, which is the focus of this review.

The Location-Allocation problem has been explored extensively in the literature (Klose & Drexler 2005), but has generally focused on commercial applications. While the underlying concepts may be similar for humanitarian use, significant differences in objectives and constraints (detailed in Section 2.3) preclude the adoption of commercial models for humanitarian contexts.

Most papers in the humanitarian literature proposed the p -hub median and hub covering formulations (refer to Table 2.4). The former generally utilises objective functions that minimise total travel times as a proxy to reduce mission times, while the latter ensures a minimum quality of service is met, aligning with humanitarian concerns of equity and fairness.

The p -hub median problem was first presented by Hakimi (1964) and represents the fundamental HLP. Within this framework, extensive work has been produced to optimise humanitarian operations that seek to minimise operation costs and mission duration.

Rawls & Turnquist (2010) remains among the most cited work in this stream of literature. The study solved the pre-location and allocation of multiple commodities for relief distribution (water, food and medical kits) by using a heuristic algorithm that considers stochastic network conditions and demand. Link state is categorised by its usability, and determines whether a link is accessible or not, with the implicit limitation of not considering link capacity or partial damages in the network.

Extensions to this work have been proposed by Rawls & Turnquist (2011, 2012). The studies incorporate a service quality objective, which ensures that the maximum distance between hub and customers does not exceed a specified limit. Furthermore, a reliability parameter α is

defined to measure the probability of meeting all demand. Rawls & Turnquist (2012) introduced a dynamic component to the pre-position of cargo for short-term relief distribution - assuming a mission horizon of 72 hours.

Table 2.4: Hub Location Problem Types.

Problem Type	Description
p -hub median	Selects a p number of hubs that minimise transport flows between all origin-destination pairs k .
Fixed cost	A relaxation of the p -hub median approach where the number of hubs selected is a new decision variable. Each potential hub location is associated with a fixed cost c , so the algorithm minimises the compound cost of selecting hubs and transport flows.
p -hub center	Presented as a minimax problem and analogous to the p -hub median, the algorithm seeks to minimise one of the following: (1) the maximum cost of transporting goods for any origin-destination pair, (2) the maximum cost of transporting goods from any node to any origin-destination pair, or (3) the maximum cost of transporting goods through any link.
Hub covering	Aims to maintain the travel cost between hubs to any demand node below a maximum value.

Given the difficulty in accurately assessing population needs and damages, humanitarian HLP studies focus on incorporating stochastic factors to increase the robustness of their models. This is the line of research followed by Bozorgi-Amiri et al. (2012), who formulated a single-commodity p -median location problem under uncertain demand, procurement costs, and transportation costs. The study uses a Particle Swarm Optimisation (PSO) algorithm (the reader is referred to Section 6.2.2 for a detailed description of this algorithm), highlighting the need to rely on meta-heuristics to improve the applicability of proposed methodologies given the scale of humanitarian logistics, a sentiment that has been replicated by Özdamar & Ertem (2015).

An important limitation of the study is the focus on reducing operational costs, rather than evaluating other performance indicators that are relevant in humanitarian operations. Bozorgi-Amiri et al. (2013) addressed this by minimising maximum shortages, formulating a p -centre HLP. The study proposes a multi-commodity robust optimisation problem solved using Compromise Programming.

The limitations highlighted in Bozorgi-Amiri et al. (2012) are common in p -median algorithms, as their objectives generally consider cost minimisation. Hub covering problems closely align with humanitarian principles, as they focus on ensuring a specified minimum level of service.

Balcik et al. (2008) were among the first to explore the humanitarian hub covering problem. Their study presented a hub allocation-location problem to optimise humanitarian relief distribution operations. An exact solution method is used to develop a global relief supply chain of two types of commodity assuming fixed hub costs and capacity, but disregarding network reliability or network state.

More complex objective functions were explored by Timperio et al. (2017), who evaluated eight different parameters: population proximity to the selected hubs, distance to the demand points, exposure risk of area to a disaster, proximity or access to key infrastructure like airports, access to major transport corridors, congestion based of runway utilisation, costs, and proximity to an economic centre.

To this point, the reviewed research has only considered stochastic components that relate to demand and costs. Nasiri et al. (2018) proposed a hub covering problem to design humanitarian supply chains considering major disruptions in hubs as a result of the disaster. The algorithm additionally considers stochastic supply and demand, heterogeneous fleets, and budget limitations.

The HLP is also applied to optimise evacuation shelter location. Hu et al. (2012) proposed a modified PSO heuristic applied to a discretised geographic optimisation problem. The algorithm seeks to minimise the number of shelters that provide sufficient capacity to satisfy demand using a distance-based allocation of evacuees to each shelter. An extension was developed by Hu et al. (2017), where a Genetic Algorithm (GA) is implemented instead. However, an important limitation of the study is the use of shortest distance routes between residential and shelters, which may not be the fastest or safest routes when considering congestion.

While initially focused on their role for surveillance and data gathering (Tatham 2009), the integration of UAVs into humanitarian and commercial delivery has been recently studied in the literature. In this context, Golabi et al. (2017) presented a stochastic facility location problem to transport relief using UAVs that is solved using a flow capturing location problem. The study assumed that people will travel to their closest open facility, with the algorithm minimising aggregate travel time for UAVs and persons. Several scenarios are generated using various disaster magnitude and seismic models of the city.

Shavarani (2019) studied the design of relief distribution networks for UAVs by formulating a multi-level facility location that allocates demand to be transported, and then selects optimal recharge stations. A GA aims to minimise aggregate distance travelled, but implicitly assumes that recharge stations have unlimited reload capacity. A similar methodology is applied to solve a commercial drone delivery system in the city of San Francisco (Shavarani et al. 2018).

Scott & Scott (2017) proposed a healthcare network design model using UAV and terrestrial transportation minimising total weighted delivery time and maximum weighted delivery times. The lack of consideration of regulatory hurdles or energy usage precludes the applicability of this model in realistic scenarios.

In the commercial context, Hong et al. (2018) proposed a recharging station coverage model that ensures the delivery areas lie within the specified range. However, the algorithm ignores energy consumption variations due to weather and payload, but includes obstacles generated by airports and other areas that impede UAV routing. The model further ignores constraints relating to the inventory levels of the recharge stations.

The studies presented in this section have focused in determining the appropriate location of warehouses and allocation of demand in humanitarian contexts. In particular, studies have considered the stochastic nature of the demand and supply, with few including network damages and disruption of the operations as a result of the disaster. Incorporating these extensions into the original HLP have required the development of bespoke heuristics and meta-heuristics to improve the scalability and applicability of the model.

Improvements to the reviewed literature can be made by considering operational components of humanitarian response including relief transportation. This would provide more accurate transportation costs, driving times, and lead times. Aspects of network damages and accessibility can be included to further improve model robustness.

2.5.2 Response Phase

The response phase consists of the operations that occur immediately after a natural disaster and focuses on transporting the required materials or personnel to the correct location at the appropriate time, while ensuring the safety of those delivering the product and those receiving

it. This phase includes evacuation process, which instead aims to relocate the population to a safe location. Both operations are vital to the success of the humanitarian response mission and to the survival of the affected population.

Relief Distribution

The time-critical aspect of humanitarian logistics makes the optimal routing of vehicles paramount in ensuring a fast, fair and safe operation. For this purpose, the Vehicle Routing Problem (VRP) provides a means to develop delivery schedules considering cost, accessibility, and priority among other factors.

Fundamentally, the VRP is a generalisation of the Travelling Salesman Problem (TSP), which determines the optimal schedule for a vehicle to travel to all its customers and return to its initial position (the depot). Instead, the VRP considers multiple vehicles simultaneously, with some modifications. Those relevant to humanitarian relief are presented in Table 2.5.

Table 2.5: Vehicle Routing Problem Characteristics.

Feature	Description
Vehicle Capacity	Vehicles have limited/unlimited cargo carrying capacity.
Vehicle Fleet Size	The number of vehicles available is constrained by available budget, a pre-determined number, or may be unconstrained.
Uncertainty	Information is known (deterministic) or uncertain (stochastic). Stochastic properties can be considered in the following: delivery times, travel times, and demand quantities.
Load Splitting	Deliveries to a single customer may be carried out by more than one vehicle.
Time Windows	Deliveries must be carried out within a specified time-frame.
Time Horizon	Problem consider a single period of operation or multiple periods.
Vehicle homogeneity	Vehicle fleet may be composed of vehicles with the same, or different characteristics.
Refuelling or Recharging	Vehicles driving distance is limited and must recharge/refuel to extend operational range.

Haghani & Oh (1996) were among the first to address the disaster relief distribution problem, which is formulated as a multi-commodity network flow model that considers multiple transport modes. Improvements to this algorithm were proposed by Barbarosoğlu & Arda (2004) by including stochastic demand and supply based on disaster intensity.

Further stochastic components are evaluated by Sheu (2007), who applies fuzzy logic to

incorporate uncertainty in demand and location of victims. The model classifies damaged areas to generate clusters, then assigns rescue teams to each cluster, and solves the VRP to transport aid to the damaged areas. An expansion to this model is presented by Sheu (2010), which introduced demand forecasting and a multi-type relief distribution algorithm.

Yi & Kumar (2007) incorporate split-delivery constraints in their multi-commodity VRP. An Ant Colony Optimisation (ACO) algorithm governs bi-stage process that determines the vehicle routes and then assigns commodities to the routes. Vehicles are allowed to wait at any node, but the itinerary must end at a depot, which may differ from the starting depot.

A time-dependent VRP is proposed by Özdamar et al. (2004), which is solved continuously during the aid delivery operation, generating requests and updating available supplies. A distinctive feature of the model is that vehicles are allowed not to return to depots, and can instead wait in their last stop until the next order is given, allowing depot relocation during response. A similar problem is presented by Victoria et al. (2016), who consider time-dependent demand that varies based on the proximity to the source of the disaster.

More recently, Ozdamar et al. (2018) presented an arc VRP, similar to the Chinese Postman Problem, that incorporates the effects of the disaster response to traffic flows. In this problem, delivering goods at an arc blocks traffic, so the objective balances the speed of the response and any disruption to traffic.

In the context of aerial delivery, Barbarosoğlu et al. (2002) incorporates crew assignment to disaster logistics optimisation problems that is structured as a two-stage model. The first phase contains the crew assignment problem considering the helicopter type and pilot competence, while the second consists of a VRP. An iterative coordination heuristic and feedback mechanism is created to ensure both phases feed their results to the final solution.

De Angelis et al. (2007) presented a multi-depot multi-vehicle routing and scheduling problem to model the delivery of food using planes by WFP in Angola. The algorithm produces weekly schedules that maximise demand satisfaction, allowing a fixed length of operation for each day. A key component is that vehicles are allowed to return to any depot to park for the night.

Refuelling constraints are considered by Özdamar (2011), who designed a means to coordinate helicopter operations for aid distribution and evacuation assistance. The model follows a two-

step process, the first being a network flow model that minimises mission time, considering the commodities and injured persons transported by each helicopter. The second phase consists of a post-processing algorithm that transforms the flows into vehicle routes, ensuring fuel constraints are satisfied. A similar problem is evaluated by Al Theeb & Murray (2017), who incorporated a crew assignment problem to the simultaneous aid distribution and evacuation problem. An iterative heuristic is proposed to improve the scalability compared to exact methods.

Bruni et al. (2018) proposed a VRP that minimises vehicle arrival times while considering stochastic travel times. Given the complexity related to computing arrival times, an iterated greedy heuristic is developed which contains three stages: a construction procedure that generates initial solutions, an adaptive local search approach that improves solutions based on well known heuristics, and a destroy procedure that diversifies the solution pool.

Recent literature has studied the feasibility of humanitarian and commercial UAV delivery. One of the first studies on UAV deliveries was developed by D’Andrea (2014), which proposed a coarse method to estimate supply chain costs and energy requirements for multi-rotor UAV delivery. Haidari et al. (2016) developed a simulation model to analyse the impact of using UAV for vaccine distribution using a range of supply chain structures.

Nedjati et al. (2015) developed an UAV-based relief distribution model for densely populated cities in the aftermath of an earthquake. The location of victims and warehouses are assumed to be known, demands are estimated using population levels as a basis, and only one commodity is transported. The model specifies three mission types: 2.5, 3 and 24-hour missions, each with different first aid kit size associated.

Within commercial drone applications, Murray & Chu (2015) formulated the Flying Sidekick Travelling Salesman Problem (TSP), where a conventional ground vehicle acts as a mobile depot for UAV deliveries to customers along its route. Extensions to the Flying Sidekick TSP have been proposed by Ha et al. (2018), Es Yurek & Ozmutlu (2018) and Ham (2018), considering multiple vehicles and depots.

Aspects of refuelling and recharging are incorporated by Coelho et al. (2017). Their study modelled urban deliveries using the Green VRP that focuses optimising energy consumption. However, akin to the previous studies, it ignores the effects of payload on vehicle range and UAV

trajectory design, and warehouse locations. The latter is particularly significant in UAV-related applications given the limited range of UAVs, as it expands the scale of operations.

The key assumption of these studies relates to the UAV operating range, which ignores varying payloads and energy management. Song et al. (2018) proposed a simplistic linear relationship that ignores other sources of energy variability. A more accurate model was developed by Dorling et al. (2017) in their study of urban goods distribution model. However, they focus on multi-rotor vehicle operations (with simple flight dynamics), and a single depot without field recharging, limiting coverage and applicability.

The problem of aid distribution has been explored extensively in the recent years using the VRP framework. Studies have focused on incorporating different aspects relevant to humanitarian response, including crew assignment, integration with evacuation procedures, and uncertainty. These studies are solved using heuristics methods to overcome the complexities of VRPs.

However, key decisions regarding the location of the distribution centres are not covered in the research reviewed above. This is of particular interest for UAV-routing algorithms as determining the appropriate locations of recharge stations can extend their operational range substantially. For this reason, the next section explores the Location-Routing Problem.

Combined Relief and Facility Location

The Location-Routing Problem (LRP) integrates aspects of the HLP and the VRP to simultaneously optimise warehouse location and delivery itineraries that satisfy demand. Similarly to the VRP, the LRP is NP-hard, so exact solutions generally lack scalability (Perl & Daskin 1985, Laporte et al. 1986, Albareda-Sambola et al. 2005, Drexel & Schneider 2015).

Amongst the most cited publications in this stream of research is Yi & Özdamar (2007), who solved the capacitated LRP for evacuation and support in disaster response. A bi-stage model first determines the location and network flows, which are later converted into vehicle routes. The algorithm maximises the response level by locating medical facilities and distributing relief. Further objectives are evaluated by Abounacer et al. (2014) including total transportation times, total personnel required to operate hubs (hub number minimisation), and total demand not

satisfied in the area.

Bozorgi-Amiri & Khorsi (2016) developed the capacitated stochastic LRP using dynamic stochastic programming that integrates strategic, tactical, and operational decisions. The objectives aim to maximise fairness and efficiency by reducing costs, travel times, and undelivered demand. The proposed algorithm, while providing exact solutions, lacks scalability due to exponential computational run-time.

A two-stage, multi-objective, multi-period and multi-commodity LRP is presented by Vahdani et al. (2018a) for humanitarian logistics. The algorithm reduces total transport cost and response times while considering split delivery and uncertainty in problem properties. Two meta-heuristics are developed to solve the problem, including a GA and a PSO algorithm.

Ahmadi et al. (2015) incorporated network failure in their LRP formulation as a binary: complete failure (road closure) or business-as-usual. A two-stage stochastic optimisation procedure is proposed and a deterministic but scalable Variable Neighbourhood Search is proposed. The latter is applied to the large case study based in San Francisco.

Rennemo et al. (2014) proposed a stochastic LRP using a three-stage approach where the first determines optimal location of warehouses, and the last two determine the amount of aid to be distributed and the vehicle itineraries. The proposed algorithm assumes stochastic demand and supply, cost and time of travelling through the network, and budget.

The LRP framework has been commonly applied to UAV deployments as recharging station location is of particular importance given the short ranges of UAVs. In the context of military UAV deployments, Sarıçiçek & Akkus (2013) studied the use of UAVs to monitor the national borders of Turkey by modelling the LRP as a bi-stage algorithm. The first stage consists of a p -hub median problem that yields hub locations, while the second stage of the algorithm determines optimal UAV routes at each hub.

A similar problem is formulated by Yakici (2016b) for tactical reconnaissance missions. An ACO algorithm is used to solve multiple problem instances, providing solutions of similar quality to exact approaches. Further expansions to this study were proposed by Yakici (2016a) and Yılmaz et al. (2018), including time-windows and spatio-temporal constraints. The latter transforms the VRP section of the LRP to a Team Orienteering Problem, which considers the

selection of beneficiaries that will provide a greater contribution to the objective.

The LRP provides an accurate representation of the humanitarian operations given the reciprocal nature of location and routing decisions. The complexity of LRPs require the development of bespoke bi-stage algorithms, where the location and routing optimisations problems are solved in tandem. This framework has been applied to problems considering time-windows, multiple commodities, and stochastic components including uncertain demand, supply and network damages.

The capability of simultaneously solving warehouse location and vehicle routing is of particular importance to problems considering refuelling constraints, as the appropriate location of these refuelling stations have a significant impact on the vehicle route and overall operational performance. Despite the opportunity for rapid deployment that UAVs provide in the context of medicine and relief delivery, most studies have focused on the vehicle routing aspect and warehouse location problem separately. While this approach may be applicable for commercial settings, where networks are long-term and relatively static, humanitarian logistics networks are not as robust, especially during immediate response.

Furthermore, UAV-routing research has proposed simplistic a representation of energy consumption and trajectory design to date. Only Dorling et al. (2017) proposed payload-sensitive energy consumption patterns for multi-copter UAVs. Despite this, their study assumes a single depot, and does not consider energy management and inventory levels. The latter also applies to other HLPs and VRPs reviewed that consider UAV recharge.

Regulatory limitations are also commonly ignored in the reviewed literature, despite their notable impact on UAV applicability. Further discussion on this matter is presented in Chapter 3.

Evacuation

This subsection has been adapted from the following publication: Escribano J, Angeloudis P, Han K (2020). Optimal design of rapid evacuation strategies in constrained urban transport networks. *Transportmetrica A: Transport Science* 16 (3), 1079-1110.

Evacuation constitutes an essential and common disaster response mechanism aiming to temporarily relocate populations before, during and after large-scale disasters while seeking to minimise potential loss of life. Given the inherent managerial, societal, financial, behavioural, and operational aspects of evacuations, adequate planning is required to ensure their effectiveness and safety.

Key among the factors impeding evacuation processes is the unique traffic patterns that arise due to the sudden surge in travel demand, which contravene nominal traveller behaviours. However, even when extensive warning is given, the mass mobilisation imposes excessive strains on the transport network due to the small time window, the lack of resources and the number of evacuees (Thomas & Kopczak 2007). As a result, network congestion is among the most significant challenges encountered during emergency evacuations and is aggravated by spillbacks, shockwaves and gridlocks.

As is the case with ordinary traffic, many evacuation strategies rely on capacity expansion measures to preclude the occurrence of congestion. Among them, contraflow and crossing elimination have been found to result in capacity increases of up to 70% and 40%, respectively (Wolshon 2001, Cova & Johnson 2003). However, such techniques rely on prior familiarity with evacuation plans and are likely to increase safety risks in scenarios where evacuees are unaware or non-compliant (Xie et al. 2010).

Demand management has been explored as an alternative method to limit congestion effects by restructuring the distribution of evacuees across the network (Abdelgawad & Abdulhai 2009). This is achieved by providing alternate safe sites to evacuees or by staging departures at their source (Chen & Zhan 2008).

In this context, UAVs can be used as a guidance mechanisms to aid evacuation, or for traffic and crowd management (American Red Cross 2015, Hildmann & Kovacs 2019, Xiong et al. 2016, Seo et al. 2017, Puri 2005). In particular, UAVs may be utilised in case damages to conventional traffic management systems fails (traffic signalling) or to provide real-time changes to the network systems.

The development of demand management optimisation algorithms necessitate the accurate representation of congestion effects in the network. However, the estimation congestion levels

during an evacuation is not a trivial task. The occurrence and scale of gridlocks, spillbacks and shockwaves are dependent upon time and user behaviour, and as such cannot be integrated directly into linear optimisation algorithms without affecting model accuracy or computational performance (Lu et al. 2005). As a result, many previous studies on congestion prediction have adopted bi-level optimisation models that involve Stackelberg games.

In such a model, the upper-layer usually focuses on decisions taken by a “super-user”, who assumes the role of the decision-maker and possesses perfect knowledge of the evacuation process. In practice, this role is assumed by emergency management and law enforcement organisations, with support from transport agencies (Urbina & Wolshon 2003). Conversely, the lower-level focuses on modelling the effects of these decisions (Urbina & Wolshon 2003, Abdelgawad & Abdulhai 2009). Typical implementations use traffic assignment models, which may vary in scale and detail (Table 2.6).

Hobeika & Kim (1998) were among the first to use demand staging for evacuation modelling of user equilibrium (UE) and shortest path (SP) assignments in nuclear power plant evacuations. However, their study did not consider system optimum (SO) assignment, limiting the effectiveness of any evacuation plan.

Sbayti & Mahmassani (2006) developed a bi-level optimisation algorithm where the upper-level generates all evacuee origin-destination choices and initial time-dependent historic assignment paths to the lower level. The latter uses a mesoscopic cell transmission model (CTM) featuring dynamic SO traffic assignment.

Table 2.6: Traffic Assignment Model Types.

Model Type	Description	Examples
Microscopic	Models every individual’s behaviour, making it easier when compared to other models to track factors such as congestion. On the other hand, it is very computationally demanding.	OREMS (Li et al. 2006) and MATSIM (Nagel & Flötteröd 2012).
Mesoscopic	Models groups of individuals or vehicles. It reduces computational complexity in exchange to some detailed interactions.	DYNASMART-P (Sbayti & Mahmassani 2006), DTALite (Zhou & Taylor 2014), and models by Szeto & Lo (2004) and Ziliaskopoulos (2000).

Macroscopic	Models vehicle movement as fluid flows with no concern on individual interactions.	DYNEV (Goldblatt 2004), and models by Friesz et al. (1993) and Han et al. (2013).
-------------	--	---

Another SO CTM-based solution is presented by Chiu et al. (2007) for no-notice evacuations. Their formulation integrated departure schedule and traffic assignment decisions into the single-destination bi-level evacuation optimisation model.

An optimal-control formulation is proposed by Liu et al. (2006), discretising the evacuation area into staged departure zones, instead of the node-based departure staging strategies described above. A similar problem is presented by Bish et al. (2014), who formulated a CTM-based mixed integer linear problem for the reduction network of clearance times with multiple evacuee types.

Mitchell et al. (2007) use a microscopic traffic assignment method with pre-trip and en-route choice models to optimise network clearance time, highlighting the scalability limitations derivative of microscopic assignment methods. Although considering single-route evacuation, Chien & Korikanthimath (2007) presented a model that consider the total duration and any congestion delays of multistage evacuations.

Chen & Zhan (2008) used agent-based simulation to evaluate the merits of evacuation staging strategies under different network conditions, finding best results for geometric transportation networks in high-density areas. Abdelgawad & Abdulhai (2010) proposed a spatio-temporal multi-modal evacuation framework that minimises travel and wait times for evacuees.

More recently, Bretschneider & Kimms (2012) presented a pattern-based dynamic flow model solved using a bi-level optimisation heuristic to minimise evacuation clearance time. Li et al. (2012) formulated a shortest-path evacuation staging algorithm that minimises evacuation congestion. Extensions developed by Xie et al. (2014) and Li et al. (2018) considered multiple destinations and incorporated a time-expanded network flow model respectively.

Kimms & Maiwald (2017) developed a CTM-based evacuation planning model considering contraflow and staged evacuation. A further study by Kimms & Maiwald (2018) investigated route resilience by minimising potential capacity using a CTM evacuation model.

An alternative formulation developed by Even et al. (2015) used a tree-design framework

to minimise converging and diverging delays and improve evacuation performance. A similar framework is proposed by Pillac et al. (2016), where congestion and conflicts are minimised through demand staging over pre-generated routes.

Another evacuation path generation algorithm is developed by Seekircher (2017), who considers the selfish behaviour of the evacuees and manipulate evacuee behaviour through strategic link blockages. The solution procedure consists of a bi-level approach where the upper-level optimises the route choice and the lower-level manipulates the network structure.

The studies reviewed in this section have focused on the optimisation of evacuation processes. In particular, the review highlights the significance of reducing the effects of congestion that impede evacuation processes. While previous literature has formulated the problem as a VRP (Özdamar 2011, Al Theeb & Murray 2017), models structured as a Stackelberg game provide a adequate framework to integrate congestion modelling into the decision-making process. Using these frameworks, studies have included a variety of components in their models.

2.5.3 Rehabilitation Phase

The rehabilitation stage aims to restore key infrastructure that is essential to the normal functioning of the community. The objectives tend to consider the overall service recovery, restoration time and cost. Given our focus on disaster relief distribution, this section reviews transport network recovery literature.

In this context, Duque & Sörensen (2011) developed a greedy randomised adaptive search procedure based on the Variable Neighbourhood Search algorithm for allocating engineering resources to repair road networks. The procedure aims to optimise network accessibility constrained by the available budget and assuming a single time horizon.

A similar problem instance is solved by Aksu & Ozdamar (2014), who proposed an dynamic Integer Programming (IP) model by relaxing a VRP problem to a path-based algorithm. The algorithm maximises cumulative network accessibility during restoration considering resource limitations and scheduling. However, the approach does not provide detailed routing options for restoration vehicles. Zhang et al. (2017) presented a post-disaster network restoration algorithm assuming stochastic restoration times. A GA is used to minimise total restoration time and the

maximise the efficiency of the restoration strategy.

Further solutions to the "debris removal problem" are proposed by Sahin et al. (2016). A recursive Mixed Integer Linear Programming model (MILP) is developed to maximise the cumulative network accessibility whilst minimising clearing time. Similarly, Akbari & Salman (2017) conceived an exact MILP model to generate a synchronised road clearing schedule for multiple engineering teams. A relaxed heuristic based on large neighbourhood search is developed to solve large problem instances.

To this point, a recurring assumption relates to damages being static over the operation time horizon. Kim et al. (2018) remedied this simplification by adopting a MILP model to minimise the weighted sum of total damages and the completion time while considering time-dependent damage characteristics. An ACO algorithm extends its applicability to larger problem instances. However, the model was tested only in conditions where damage only varied once during the mission, nor considered that repaired nodes can be destroyed again.

The combined problem of relief distribution and network repair has been studied by Yan & Shih (2009). A multi-objective MILP model was proposed which is capable of planning emergency road repair and subsequent relief distribution for multiple commodities simultaneously. The objective reduces operational time, and considers equity constraints to ensure demand satisfaction. A three-step method prioritises restoration works, schedules workers, and optimises commodity flows.

Ransikarbum & Mason (2016) developed a Goal Programming-based (GP) multi-objective integrated response and recovery model that optimises network recovery and relief distribution simultaneously. Given limited operational budget, capacity and resource availability, the algorithm minimises operation cost considering fairness and equity. Li & Teo (2018) conceived a multi-period bi-level programming model to optimise road repair and relief optimisation. They employ a GA to maximise relief delivery quantity, network accessibility and reduce operation time.

Finally, Aslan & Çelik (2019) presented a pre-positioning model for relief items considering restoration efforts and relief transportation assuming stochastic road network damage. A two stage stochastic optimisation program is proposed, which is solved using a Sample Average

Approximation process. However, similarly to other approaches, the demand and damage of the network is assumed to be known before relief and recovery efforts take place.

Post-disaster network restoration literature generally focuses on the maximisation of accessibility. In this context, studies assume deterministic damage distribution on the network given that restoration operations occur weeks after the initial post-disaster operation, and sufficient time is assumed to assess network damages. Among the only problems that considers stochastic network damage is Aslan & Çelik (2019), but the uncertainty is only applied during the pre-positioning phase of their model.

The immediacy of response operations means network damages are not fully assessed when relief is distributed. Therefore, algorithms that integrate relief and repair operations could include stochastic network state in their approach. This is, however, not a trivial task, as repair operations would alter the stochastic parameters that model the network. The resulting problem is endogenous stochastic in nature, which is significantly more challenging to solve than exogenous stochastic problems.

2.6 Concluding Remarks

This section has analysed the current humanitarian logistics structure and state-of-the-art in post-disaster response research, including relief distribution and evacuation. The VRP has been extensively used to optimise relief distribution operations and have been extended to adopt stochastic optimisation frameworks that consider uncertain environmental elements, including demand and supply, vehicle fleet capacity or budget, and network state.

The review has included aspects of the preparedness phase of humanitarian response -hub location and resource allocation- as these are often integrated with relief distribution using the LRP. Similarly, operations belonging to the recovery phase -network repair- have also been reviewed.

The irruption of UAVs into the field of humanitarian response brings additional constraints and objectives that are yet to be explored. While a few studies evaluate UAV routing algorithms in the context of humanitarian relief, no holistic framework yet exists that includes key constraints

that relate to energy management, recharge station location, integrated vehicle routing, and accurate energy consumption representations. Chapter 5 develops such a framework.

Furthermore, UAVs provide an opportunity to develop novel mathematical problems inspired by their unique applications. For instance, UAVs can be deployed to carry out relief distribution and damage assessment simultaneously. However, no study yet has proposed such a problem, despite the existence of real-life examples (DuPlessis 2016). This problem has been formulated as an endogenous stochastic optimisation model and is presented in Chapter 7.

No study in the reviewed literature incorporates current regulatory frameworks to their UAV trajectory designs, as most assume straight paths between origin-destination pairs. Despite the application being appropriate in specific contexts like rural areas, the feasibility of these must be investigated in greater detail. For this reason, Chapter 3 reviews existing regulations and trajectory algorithms, which inform the development of a trajectory management algorithm presented in Chapter 6.

Chapter 3

Drone Operations Regulations and Methods

Chapter 2 reported a lack of integration of UAV-related constraints into humanitarian operations. In particular, the review highlighted the simplifications relating to energy consumption and the disregard of regulatory frameworks.

While the linearisation of fuel consumption relationships may be reasonable for some road vehicle applications, this assumption would misrepresent the effectiveness and operational range of UAVs, particularly when considering variations due to weather and payload sizes. Hence, the nonlinear behaviour of battery consumption is no longer insignificant, and have to be considered in planning. These factors are covered in the design and optimisation of UAV trajectories.

Furthermore, UAVs will operate in shared airspace in conjunction with other aircraft of different sizes and capabilities. As a result, any trajectory optimisation or planning algorithm developed must conform existing regulations.

This section reviews the current regulatory frameworks that outline the minimal technical and safety requirements for any UAV operation. Specifically, this chapter aims to:

- Review the international manned aircraft regulatory frameworks, given the early stage in development of UAV rules.
- Explore the field of trajectory planning, identifying gaps and common methodology including mathematical-based exact solution processes and heuristics.

The review identifies sampling-based trajectory planning heuristics as a suitable method to incorporate trajectory design aspects to the humanitarian relief model. To ensure the continuity of this thesis, the algorithms developed will conform with the Performance Based Navigation Framework that govern international manned aviation. A summary of the literature surveyed is presented in Appendix B.

3.1 Drone Regulations

Aircraft regulation arose in the aftermath of World War II with the adoption of the Chicago Convention in 1944 (ICAO 1944), which established the International Civil Aviation Organisation (ICAO) as the international regulatory body for manned and unmanned aviation and exists to this day. Its aim is to promote the planning and development of international air transport ensuring the safe growth of international civil aviation.

Until the development of the first UAV-specific regulations in 2002 by the UK and Australia, UAVs conformed with model aircraft regulations. However, the rapidly growing market and technology enabled their widespread use for commercial applications, which required the development of appropriate rules to ensure their safe and responsible use.

ICAO presents guidelines and common regulatory frameworks to aid national states in the development of Unmanned Aircraft System Traffic Management (UTM) systems and promote harmonisation (ICAO 2019*b*). In addition, the Task Force on UAS for Humanitarian Aid and Development (TF-UHAD) was formed to aid the development of harmonised regulations by national aviation authorities in the context of responding to emergency situations (ICAO 2019*a*).

Among the requirements for a harmonised regulatory framework is the enforcement of automatic geo-fencing and Detect-and-Avoid (DAA) mechanisms, which limit the use of UAVs to VLOS operations where the pilot holds responsibility of See-and-Avoid (SAA). Examples include Europe (EASA 2015), United States (FAA 2012) and China (Stöcker et al. 2017) among others.

Some important differences in the approach towards regulating UAV activity is found in leading legislation in Europe and the United States. These are reviewed in the following section.

3.1.1 Current Regulatory Framework

In Europe, the regulation of UAV activity is governed by the European Aviation Safety Agency (EASA), which was granted competence to legislate all drone activity by the revised European Commission Basic Regulation in 2018 (European Parliament and Council 2018). The EASA Technical Opinion NPA 10/2017 (EASA 2017c) identified three drone categories (defined in detail in Table 3.1): Open, Specific and Certified. The regulatory framework proposed is designed to be operation-centric and proportionate to risk and performance.

Table 3.1: EASA Drone Categories (EASA 2015).

Category	Description
Open	Constitutes very low risk operations, such that safety is ensured solely by minimum set of rules, requirements and operational limitations. No authorisation by the competent aviation authority nor declaration by the operator is necessary.
Specific	Consists of any operation in which drones pose a significant risk to persons or where a drone shares airspace with manned aviation. It includes any operation that exceeds the Open category requirements, and requires authorisation from the relevant national aviation authority, as well as a risk assessment by the operator.
Certified	Composed of high risk operations where the requirements are similar to those for manned aviation. Therefore, regulations are adapted from manned aviation.

Initial rules issued in 2017 (EASA 2017a) set the principles of registration, responsibilities of operators and pilots, product legislation, e-identification and geo-fencing. In addition, Member States are given responsibility to define airspace zones where the use of UAVs is prohibited, permitted or limited. The intention is to provide greater flexibility to the Member States.

Opinion 01/2018 (EASA 2018) extended regulation on the open and specific categories by defining specific measures to mitigate the risk of their operations. In particular, three risk areas were identified that preclude widespread UAV adoption: airborne conflicts, loss of control and system failure.

Requirements relating to the maximum flying height, VLOS requirements, remote pilot competency and determination of no-fly zones are identified as key towards the mitigation of risks of ‘open’ operations. In addition, the rules established subcategories A1 to A3 and C0 to C6, which differentiates operations based on its proximity to people and drone classes on the

UAV's take-off weight and its technical requirements, respectively.

Within the specific category, EASA proposed the creation of Light UAS Operator Certificates (LUC), providing the operator with privileges including authorising their own operations (EASA 2017b). Without an LUC, the operator requires clearance by the competent authority and must submit an operating risk assessment – comprising an airworthiness assessment, operating procedures and environment -, the competencies of involved personnel and organisations, and potential airspace issues.

The current road-map of EASA estimates that registration of drones in the open and specific categories will become mandatory by the end of 2020. Operations in both categories are expected to start in December 2021, depending on the competent National Aviation Authority.

In the United States, the Federal Aviation Administration (FAA) is responsible for developing new regulation. Drone rules were issued under FAR Part 107, with exemptions issued on an individual basis according to FAR Part 333 (FAA 2012, 2016). These rules are analogous to the open category proposed by EASA.

Any BVLOS operation, night-flight, flight of multiple UAVs simultaneously with one pilot require and any operation that exceed operational limitations defined in FAR Part 107 requires the waiver. The process to acquire the waiver is time consuming, with an estimated response time of 90 days.

A key difference between the EASA and FAA approach lie on the provision of BVLOS operations. While neither permit BVLOS flight yet, EASA provides a regulatory framework based on the specific category, and expects capabilities to be in place in the near future through U-Space (see Section 3.1.2). In the current legislation, FAA does not assess the potential regulations of BVLOS operations, and are only permitted in a case-by-case basis through the FAR 107 waiver programme.

Table 3.2: Comparison between regulations in the European Union (EU) and the United States of America (US).

Country	ID	Detect & Avoid	DfP	Max dis- tance	BVLOS	Max Height

US	Category dependent	Specific category	Category dependent	500m	Specific category	150m
EU	Category dependent	>25kg	150m	EVLOS possible	On approval	122m

3.1.2 Traffic Management Services

The introduction of UAS into the airspace requires the provision of traffic management services. The scale, altitude range and the extent of applications of UAS requires the development of bespoke systems and procedures to ensure their safe deployment. Two major projects: U-Space in Europe, and UTM in the United States, are explored in this section.

U-Space

U-Space consists of a set of federated services and procedures that provide simultaneous, safe and efficient access to the airspace for UAVs. The systems aim to integrate UAVs into non-segregated and controlled airspace, under all operating environments for all types of drones, mission, and user types.

Table 3.3: U-Space Service Levels (SESAR 2019).

Category	Description	Services
U1	Provides "foundation services", which would allow UAVs to operate in areas where manned aircraft density is low. Completed by 2019. The aim is to provide VLOS and extended VLOS capabilities.	Pre-tactical geo-fencing, e-registration and e-ID.
U2	Extends services from U1 and develops a first level interface between ATM and manned aviation. Drone flights will no longer be considered on a case-by-case basis, and some BVLOS operations will be deployed routinely. Completed by 2022.	Tracking, flight planning management, emergency management, and strategic deconfliction, among others.

U3	Focuses on expanding the developed technologies to high density and complex areas. The most significant increase in drone operations is expected in this category, as operations such as Urban Air Mobility will be possible. Completed by 2027.	Dynamic geo-fencing, collaborative interface with ATC and dynamic capacity management, among others.
U4	Increases automation and improves interfaces between ATM and manned aircraft with respect to U3. Full integration with manned aircraft is supported. Completed by 2035.	Yet undefined.

Four levels based on the technological requirements and deployment stage compose U-Space. Their deployment and design is incremental, with each step proposing new services and extending existing capabilities. The description and services provisioned at each service level are presented in Table 3.3.

U-Space will provide services to authorities by developing inter-operability capabilities between U-Space and ATM, data services through the provision of terrain, weather, and obstacle data, and preparation, optimisation and deconfliction services to drone operators before, during of after flight. Key components of U-Space include geo-fencing, flight path management, capacity management, weather monitoring and automation.

Equally important is the development of adequate standards that support the harmonised implementation of U-Space. In this effort, the European UAS Standards Coordination Group (EUSCG) has been developed, which monitors and maintains a standardisation development plan.

UTM

NASA's UAS Traffic Management (UTM) project aims to develop a traffic management system that enables small UAVs to access low-altitude airspace safely (below 400 feet above ground level). Similar to U-Space, the fundamental objective of UTM is the development of technologies, responsibilities and procedures to manage UAVs. The demonstration of its capabilities is categorised into four categories of increasing complexity named Technical Capabilities (TC) and shown in Table 3.4.

In this system, BVLOS operations require to obtain a Performance Authorisation from the FAA Regulators. The FAA then evaluates the request, ensuring that minimum performance levels for navigation, communications and avoidance capabilities are met. Initially, the process will be done on a case-by-case basis. However, it is expected that industry or government develop specific packages depending on geographical location and intent (FAA 2015).

Table 3.4: UTM Technical Capacity Levels.

Technical Capacity	Description
Level 1	Rural UAV operations like agriculture, infrastructure monitoring and firefighting. Completed in 2015 (Rios et al. 2017).
Level 2	BVLOS operations over sparsely populated areas. At this level, the UTM system checks for conflicts automatically, and rejects or accepts the plan based on these constraints. Completed in 2017 (Homola et al. 2017).
Level 3	BVLOS operations over suburban areas, including some manned/unmanned conflicts. The focus of TCL3 was to test the Concept of Operations, communication and navigation services, and particularly the sense and avoid and data exchange systems. Completed 2018 (Rios et al. 2019)
Level 4	BVLOS operations over densely populated areas and high air traffic density. The test is expected to occur in 2019 (FAA 2015).

The rapidly evolving regulations, and uncertain number and extent of services to be provided to UAV operators by Unmanned Traffic Management Software make the consideration of regulations in trajectory design challenging. For this reason, we review the current regulatory framework of manned aircraft.

3.1.3 Performance Based Navigation

While the services described above will allow widespread use of UAS within very-low airspace, any integration of UAS into manned airspace will require conformity with existing regulations. Currently, any manned aerial operation must adhere to the Performance-Based Navigation Framework (PBN).

These principles, outlined by ICAO, specify the Required Navigation Performance (RNP) and Area Navigation (RNAV) systems performance required to enter the airspace (ICAO 2008). The PBN requirements are defined in terms of accuracy, integrity, continuity and availability,

the meaning of each is presented in Table 3.5.

RNP determines the on-board performance specifications of its monitoring and alerting capabilities. As an example, RNP 5 specifies that the navigation system must calculate its position within a radius of 5 Nautical Miles (ICAO 2009).

Therefore, instead of determining flight conditions based on on-board navigation sensors, PBN ensures that any aircraft in the airspace will meet the minimum requirements to safely travel within this airspace. As the framework is technology-agnostic and operation-centric, any UAV that enters manned or shared airspace will need to conform with existing PBN and RNP requirements.

Table 3.5: PBN performance measurements (ICAO 2008, 2009)

Measurement	Description
Accuracy	Defines the error distribution position, speed and velocity measurements, which must remain within a specified limit 95% of flight time.
Integrity	Refers to the ability of the system to provide warnings if the aircraft transgresses a specified limit (referred to as containment limit) generally set at 2RNP.
Availability	Calculated based on the time that the navigator can use the services of the system.
Continuity	Refers to the probability of losing a navigation service during operation.

Currently, there are no RNP requirements specified for UAV applications. While some studies delved on the development of RNP requirements, extensive testing and numerical analysis are necessary for the incorporation of the RNP categorisation frameworks into UTM operations (Chiesa et al. 2014, Ren, Castillo-Effen, Yu, Yoon, Nakamura, Johnson & Ippolito 2017, Vascik et al. 2018). The problem is of particular complexity given the heterogeneity in pilot competency, automation degree, intent, and aircraft capabilities.

3.2 Trajectory Planning

In its most fundamental principles, the field of trajectory planning aims to determine the route from a given starting point to a specified destination that minimises a set of objectives. While commonly concerned with finding the shortest routes, further objectives may include maximising

safety or avoiding detection among others.

The solution approach of trajectory planning problems can be divided into four sectors: mathematical model-based optimisation, sampling-based heuristics, node-based heuristics, and meta-heuristics. The next section reviews the literature of mathematical model-based optimisation, while Section 3.2.2 presents recent studies on the rest. Note that a combination of multiple approaches is also common, for example the use of meta-heuristics to provide better solutions to mathematical modes, or the use of sampling and node-based heuristics.

3.2.1 Mathematical Programming

Trajectory planning problems can be formulated as an MILP, achieved by defining the problem as a series of discrete decisions considering linear constraints. Constraints resulting in non-linearity normally involve bank angle and maximum airspeed, which can be converted to forces acting on a point mass to create a linear problem. Richards & How (2002) used this method to solve the trajectory planning method of aerial vehicles with collision avoidance.

Another example is presented in Bellingham et al. (2002), who solved a receding horizon control problem to determine the trajectory of aircraft while avoiding rectangular obstacles. A more recent use of receding horizon MILP is provided by Kuwata & How (2011), who created a cooperative trajectory optimisation algorithm. The method is defined as a distributed robust safe but knowledgeable algorithm, where the vehicles define their trajectory that can be modified considering the movement of other vehicles.

Forsmo (2012) developed a MILP-based UAV path planner based on simplified kinematic constraints, with the resulting linear problem being similar to the TSP. Flight time minimisation is the objective of the MILP considering collision avoidance and static obstacles. Furini et al. (2016) proposed an MILP formulated as a Time Dependent TSP that considers airspace congestion. The algorithm generates conflict zones which enforces a wait procedure until sufficient clearance is given.

A recurring limitation of MILP formulation is the simplification of kinematic constraints and non-linear relationships that emerge in UAV flight. Optimal Control Problems (OCP) optimise trajectory designs by manipulating control variables like thrust and head angle, which affect the

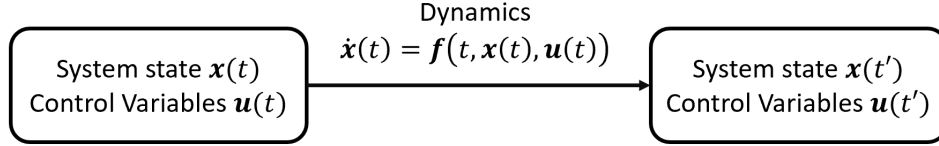


Figure 3.1: Trajectory Optimisation Schematic.

system state over time (see Figure 3.1). These problem instances are also known as Trajectory Optimisation Problems (TOP), and are solved using the indirect method or the more popular direct approach.

The indirect method is solved by constructing the optimal conditions analytically, which yields an accurate result but may not be applicable to complex problem instances. An example of this method's applicability is shown by Burrows (2012), who solved the energy-efficient trajectory optimisation problem for a Boeing aircraft by expressing the objective function as a Hamiltonian.

The direct method, on the other hand, transforms the OCP into a nonlinear problem (NLP) model via discretisation and parameterisation of control and state variables. Its popularity resides in its ability to find a solution without needing to develop a full analytic expression (Betts 1998).

The most commonly used solution methods are direct shooting and direct collocation (or direct transcription). Shooting estimates initial conditions and solves the NLP by integration, which is performed using piece-wise control while evaluating constraints. The process is repeated until convergence is reached.

Although single shooting is simple, it requires accurate guessing to yield a feasible solution, which is challenging considering the instability of some NLP. Multiple shooting allows the estimation of initial solutions in different sections of the trajectory, overcoming the instability of the NLP (Morrison et al. 1962).

On the other hand, collocation compares the first derivative of the interpolating polynomial to the derivatives of the equations of motion. If both derivatives are equal, then the interpolation technique used is assumed to be a good approximation. Numerous interpolation techniques can be used, ranging from very simple linear interpolation, to the more common Lagrange or

Hermite interpolation techniques (Delahaye et al. 2014).

The applicability of direct trajectory optimisation methods is widespread in the literature. In the context of military UAV routing, Geiger (2009) used NLP and neural networks to maximise surveillance time while accounting for wind speeds. The objective function consists of 4 sections: the first two penalise acceleration and bank angle, the third considers the distance to the target, and the last evaluates the position of the target in relation to the UAV camera.

Another major application of trajectory planning is to minimise the energy expenditure or to extend flight time, with a variation of techniques available in the literature. Langelaan (2007) presented a trajectory planner that uses ridge soaring to maximise flight range and speed. Zhou (2013) creates a dynamic model for trajectory optimisation of a fuel cell UAV. A model of the fuel cell is developed to estimate fuel consumption during flight, being validated through experimental data. Finally, a Genetic Algorithm (GA) is proposed to generate waypoint-based 3D paths, in which the GA alters the sequence of waypoints the UAV follows and the heading angle constraint at each waypoint.

Gao et al. (2014) developed an integrated methodology to optimise design parameters and flight trajectory of a high-altitude solar-powered UAV using the Gauss pseudo-spectral (GSM) in conjunction with a Particle Swarm Optimisation algorithm (PSO). The GSM solves trajectory optimisation, and the PSO determines the optimal design parameters. The trajectory is divided into five stages: ascent using maximum available power, high altitude flight with recharge, maximum power ascent and storage of solar energy through potential energy, gliding and low altitude flight with discharging battery. The GSM can solve all separate stages simultaneously, determining the flight status at the start and end of each stage.

Zhu et al. (2016) also used GSM for trajectory optimisation to extend flight time of a small-scale UAV through the use of dynamic soaring. Using a wind profile model and controlling flight parameters such as airspeed, heading, flight path and bank angles, a flight simulator was developed. Despite accurately calculating the distance travelled by the UAV, the simulation consistently overestimated the energy savings from soaring.

Trajectory optimisation is also applied by air traffic management bodies to analyse of conflict patterns, which informs the development and provision of traffic management services. For

example, the European Organisation for the Safety of Air Navigation (EUROCONTROL) developed the Free Route Airspace service which allows users to plan a route by knowing the entry and exit point.

In this context, Raghunathan et al. (2004) developed an algorithm for the detection and avoidance of collisions of aircraft in free space. The algorithm designs an obstacle-free trajectory that requires the minimum number of manoeuvres constrained by a three-dimensional vehicle dynamics model. A similar methodology was proposed by Ruiz & Soler (2015) to analyse conflict patterns in the European ATM by developing a model consisting of a microscale level in which optimal trajectories are found for each flight, and a mesoscale level that analyses conflict patterns in the airspace.

Further conflict and obstacle resolution has been explored by Arrieta-Camacho et al. (2007), who developed real-time coordination system for multiple aircraft based on a nonlinear predictive controller. After problem discretisation and formulation of the NLP problem, the Interior Point Optimisation (IPOPT) algorithm is used to determine trajectories between waypoints, avoiding static and dynamic obstacles in the process.

Finally, Pannequin et al. (2007) developed a collision avoidance trajectory planning for multiple aircraft. However, the method centred on severe weather avoidance, which was achieved by identifying and clustering these to create a boundary using elliptical functions to enclose them. Two objective functions are explored: a quadratic cost that penalises distance between the aircraft and their destination at the end of the time horizon, a Hamilton-Jacobi function that represents the cost for the vehicle to arrive at their destination.

The main limitation of the OCP problem formulation lie on their unreliability to find feasible solutions and their computational expense. The methods considered often require an initial solution to ensure solution feasibility, but this may not always be possible. In response, researchers use faster heuristics that provide sub-optimal yet feasible solutions.

3.2.2 Heuristic Algorithms

Trajectory planning heuristics can be divided into two sections: sampling-based and node-based methods. In the latter, the airspace is discretised into a node-based network, where the network-

construction step becomes a fundamental factor in the development of optimal paths. The network must consider the kinematic constraints of the aircraft.

Once the environment is discretised, a path-search algorithm is utilised to determine the optimal path. Dijkstra's, A* and Dynamic A* constitute some widely used examples which share common common limitations relating to its discretised nature and inability to consider complex objective functions (Yang et al. 2015, Ousingsawat & Campbell 2004, Sathyaraj et al. 2008, Yang et al. 2009, Li et al. 2009).

Sampling-based algorithms explore the environment by designing a path randomly or pseudo-randomly given specific constraints. This stream of literature is composed tree-based algorithms, and Probabilistic Road Maps (PRM).

PRM essentially constructs a network of nodes and directed edges. Using a pre-defined sampling method – outlined in Lavalley (2006) – the algorithm generates and groups nodes into clusters based on a specified cluster radius. Once the full graph is finished, a node-based search algorithm may be utilised to find the optimal path. The clustering method of the PRM seldom generates straight paths, and thus are likely sub-optimal. Modern variations of the PRM are produced to improve path optimality and performance.

In the context of UAS path planning, PRM has been utilised to cope with complex environments. Hrabar (2008) proposed a combined D*-Lite and PRM-based 3D path planner for rotor-craft considering stereo-based detect-and-avoid capabilities. Yan et al. (2013) developed a 3D PRM-A* algorithm in which the airspace is discretised to cuboids of different dimensions to determine the area occupied by obstacles. A similar approach is conceived by Li et al. (2014) which considers terrain and threat avoidance.

PRM algorithms are particularly useful in static environments, as they develop fully-connected networks that can be used to generate multiple shortest paths. However, given the highly dynamic environments in which UAVs will operate, tree-based algorithms are more suitable given their speed and versatility.

Instead of creating a network, tree-based algorithms develop a single path between an Origin-Destination pair. At each iteration, a new node is generated which is connected to the closest neighbouring node. This way, the tree is progressively extended until the destination

is reached. Their main limitation resides in their slow convergence rate given their stochastic nature.

Rapidly-Exploring Random Trees (RRTs) are the most common type of tree-based algorithm used in the path planning literature, and has been subject to numerous alterations and variations to improve its performance. Urmson & Simmons (2003) presented a heuristic sampling bias approach that alters the exploration algorithm based on predicted cost, improving runtime under variable cost planes. Kothari et al. (2009) explored the use of RRT for path planning of multiple UAVs in obstacle-rich 2D environments. Lin & Saripalli (2014) present RRT with Dubins curves for static and dynamic obstacles in 3D space. Lin & Saripalli (2017) implemented an RRT for collision avoidance, allowing the vehicle to edit intermediate waypoints in the event of incoming conflicts.

Gammell et al. (2014) improved convergence by developing an Informed RRT* that defines the search space as an ellipsoidal set, reducing its size as iterations increase. Pharpata et al. (2017) used the RRT* combined with a potential field that bias sampling to plan fixed-wing UAV flight paths.

Among the most recent applications, Chakrabarty et al. (2019) developed a UAV path planner for UTM-TCL4 (refer to Section 3.1.2). A Kinematic Tree Algorithm is proposed, which differs from an RRT by generating new points based on the non-extended tree, instead of selecting a random state.

Finally, the last methods used to solve the trajectory optimisation problem are evolutionary algorithms. Genetic algorithms (GA) have been used by researchers in combination with other solvers as they allow good initial guesses for the state and control variables. GAs have been previously seen in Zhou (2013).

Yokoyama & Suzuki (2005) developed a GA to solve the Brachistochrone and space re-entry problems after being converted to NLPs. The GA successfully identified the region around the global solution and yield results that are close to the analytical solution. However, the problems explored are deemed simple – only the space re-entry problem has some practical use.

Wuerl et al. (2003) used the GA with the Davidon-Fletcher-Powell penalty function and calculus of variations method (DFP-COV) to optimise a low-thrust space flight from Mars to

Earth. The GA learns using feedbacks from the DFP-COV, increasing the likelihood of the GA to find the optimal solution.

Lastly, Shippey & Subbarao (2008) developed a hybrid GA with a collocation method to determine the optimal trajectory optimisation of an aircraft during a turning manoeuvre. Again, the GA serves as a means to determine initial conditions that lead to optimality.

Among the presented methodologies, tree-based algorithms provide a fast and flexible framework which have been implemented in very recent applications (Chakrabarty et al. 2019). In comparison, PRM cannot account for highly dynamic elements, and other meta-heuristics have lower convergence rates. Node-based solutions provide simplistic frameworks that precludes the exploration of complex objective functions relating to safety and security.

3.3 Concluding remarks

This chapter presented an overview of the UAV regulations in Europe and the United States, which are developing bespoke UAV Traffic Management Systems, known as U-Space and UTM respectively. The deployment of fully functional UTM airspace is yet a long-term prospect, and any development of trajectory planning algorithms that considers current capabilities will shortly be outdated.

Given that current UAV regulatory frameworks are operation-centric, similar to international manned aviation rules, and that any integration of UAV into manned airspace will require conformity with the relevant regulation, any trajectory planning model developed must conform with PBN requirements. This allows processes to be generalised to other applications and ensures long-term continuity.

Within the trajectory planning literature, the survey identified four streams of study: mathematical-based solution approaches, sampling-based heuristics, node-based heuristics and meta-heuristics. Mathematical-based solutions require significant simplifications to transform the problem to MILPs, and NLPs are particularly unstable as they require the accurate determination of initial conditions to obtain feasible solutions.

Node-based heuristics require the discretisation of airspace into node networks, the accuracy

of which depends on the discretisation approach. In addition, these methods are unable to consider complex objective functions.

Sampling-based heuristics have been particularly useful in the field of robotic path planning problems. Within this stream of research, tree-based planning methods are particularly suitable for UAV path planning in dynamic environments. In addition to being able to evaluate complex objective functions, variations of the original RRT algorithm have improved convergence rate, performance, and ability to re-use sections of the generated tree, which are particularly useful when considering dynamic obstacles.

The review carried out in this section informs the development of drone operation algorithms presented in Chapter 6. The frameworks presented are to be used in combination with the humanitarian operations algorithms shown in Chapter 5.

Chapter 4

Fight Dynamics and Optimisation

Methods

The frameworks to describe humanitarian and UAV operations analysed in Chapter 2 have centred around mathematical optimisation. Most studies have structured their problems as a Vehicle Routing Problem, which is a combinatorial and integer programming problem that is NP-hard, meaning that finding an exact solution to large problem instances is infeasible. As a result, such problems require alternative techniques to find close-to-optimal solutions.

This section presents the field of mathematical optimisation and analyses existing methods determining their suitability for different problem instances. In addition, the dynamics of flight for UAVs are defined, as must be considered during modelling. In summary, this chapter aims to:

1. Derive and describe the flight dynamics of UAVs.
2. Review the field optimisation theory and classify optimisation algorithms.
3. Discuss the advantages and disadvantages of optimisation algorithms and develop a criterion for the selection of algorithms to solve the problems presented in this thesis.

This chapter is structured as follows: Section 1.2 derives the flight dynamics of hybrid UAV platforms. The following section evaluates optimisation theory, reviewing exact solution

methods, heuristics and metaheuristics. The relationship between the methodological structure is then analysed in Section 4.3, with the a solution method criterion developed in Section 4.4.

4.1 UAV Flight Dynamics

UAV flight dynamics are dependent on the type of aircraft to be modelled. As per Table 3.1, the most common UAV type used for logistics purposes are hybrid aircraft, that is fixed-wing aircraft with VTOL capabilities. In deriving the dynamics, we define two frames of movement, the inertial frame which describes the orientation of forces with reference to the ground (z being the altitude), and the body frame, which is defined by the orientation of the vehicle (see Figure 4.2).

4.1.1 Vertical Take-off and Landing

In this section, we consider only the aircraft movement on the vertical (z) axis. The thrust exerted by all motors are aligned in the same direction, so they can be aggregated into a single upward force. In turn, the only resistant forces acting on the vehicle are its weight and its drag. Thus, the force exerted by the rotors must exceed the vehicle's weight, its inertia and drag resistance in order to take-off as shown in Figure 4.1a.

$$ma = T - mg - D \quad (4.1)$$

In a state of vertical equilibrium where vertical take-off speed is constant to equation yields $T = mg + D$ where T denotes the thrust force, m represents the mass of the vehicle, a is the vehicle's acceleration rate, and g defines the acceleration due to gravity. The propulsive power P required to generate a given thrust T is determined from the following:

$$P = UT/\mu \quad (4.2)$$

where U determines the instantaneous speed of the vehicle, and μ is the efficiency parameter. The electric current required for P is determined by:

$$I = P/V \quad (4.3)$$

where V represents the battery voltage. The above relationships can be used to determine power consumption rates at any point in the flight. Over any discrete time-step δt , the resulting change in the state of charge (SOC) of the vehicle is:

$$\Delta SOC = -I\Delta t/B \quad (4.4)$$

where B is the capacity of the battery. Using these set of equations, we can estimate the energy cost of take-off based on a predefined take-off altitude and speed.

4.1.2 Fixed-Wing Aircraft

Given the short vehicle flight UAVs can take, we adopt a three dimensional point-mass aircraft model in a flat Earth-fixed reference frame shown in Figure 4.2 (Dai & Cochran 2009, Hosseini & Mesbahi 2013). In this frame, the acceleration $\dot{U}$ of the vehicle is described as the force imbalance of the vehicle, which exerts a forward thrust T and is subject to a resisting drag force D and a gravitational component $\sin \gamma$.

$$\dot{U} = ((T - D)/W - \sin \gamma)g \quad (4.5)$$

The vehicle position coordinates are governed by the flight speed V , flight path angle γ and heading angle ψ :

$$\dot{x} = U \cos \gamma \cos \psi \quad (4.6)$$

$$\dot{y} = U \cos \gamma \sin \psi \quad (4.7)$$

$$\dot{z} = U \sin \gamma \quad (4.8)$$

The bank angle is expressed based on the horizontal and vertical load factors $n_v = \frac{L \sin \phi}{mg}$ and $n_h = \frac{L \cos \phi}{mg}$ such that:

$$\phi = \arctan\left(\frac{n_h}{n_v}\right) \quad (4.9)$$

This expression can be used to

$$\dot{\psi} = \frac{gn_h}{U \cos \gamma} \quad (4.10)$$

$$\dot{\gamma} = \frac{g}{U}(n_v - \cos \gamma) \quad (4.11)$$

The aircraft dynamics are governed by the drag force D , which depends on the drag coefficient C_D , lift coefficient C_L , parasitic drag coefficient C_{D0} and aerodynamic coefficient K .

$$D = \frac{1}{2}\rho U^2 S C_D \quad (4.12)$$

$$C_D = C_{D0} + K C_L^2 \quad (4.13)$$

The final set of equations describe the flight pattern of a fixed-wing UAV in 3D:

$$\dot{x} = U \cos(\psi) \cos(\gamma) \quad (4.14)$$

$$\dot{y} = U \sin(\psi) \cos(\gamma) \quad (4.15)$$

$$\dot{z} = U \sin(\gamma) \quad (4.16)$$

$$\dot{U} = (T - D + L \sin(\gamma))/m \quad (4.17)$$

$$D = q S C_D(t) \quad (4.18)$$

$$C_D = C_{D0} + K C_L^2 \quad (4.19)$$

$$C_L = L_v(t) + K C_L^2 \quad (4.20)$$

$$L = \frac{mg}{\cos(\gamma)} \quad (4.21)$$

$$S\dot{O}C = -\frac{TU}{\mu BV} \quad (4.22)$$

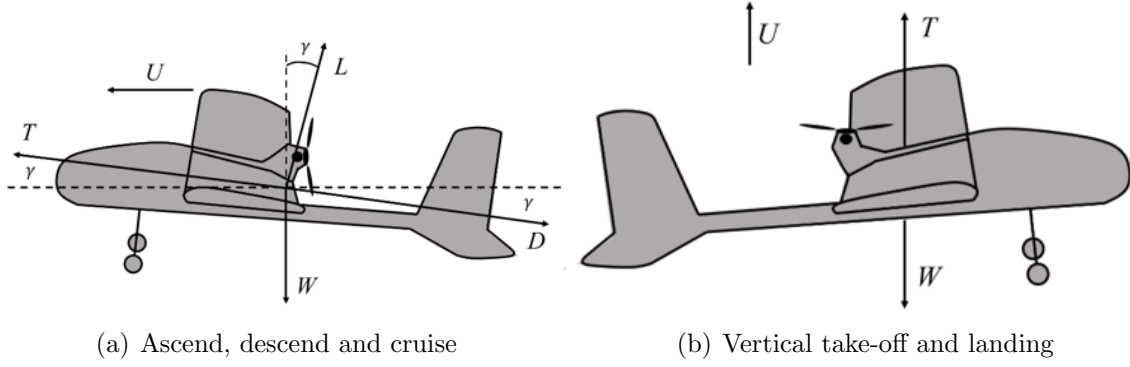


Figure 4.1: Point mass model of UAV during flight.

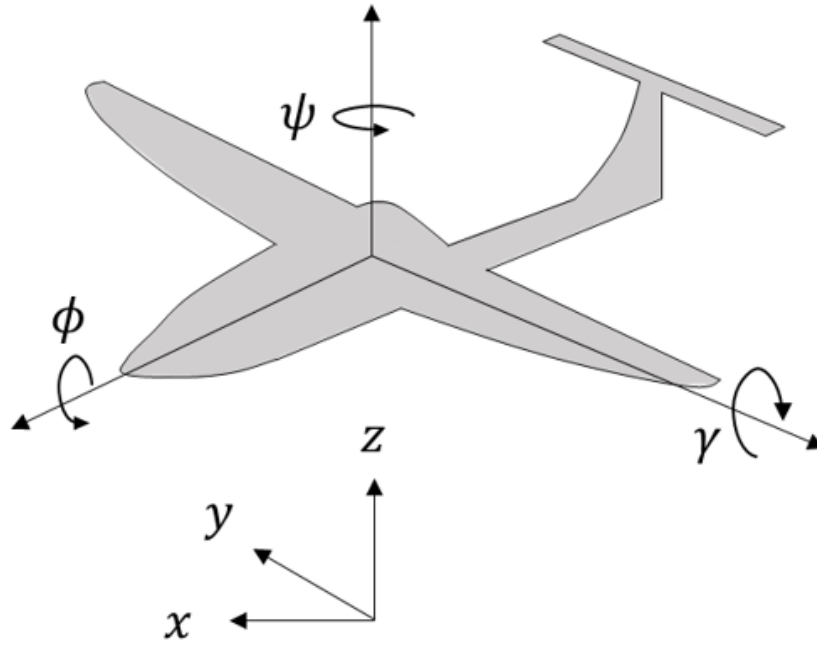


Figure 4.2: The coordinate system of the trajectory optimisation problem.

4.2 Optimisation Theory

Chapter 2 showed that humanitarian operations are generally structured as mathematical optimisation problems to determine the strategy that maximises or minimises a specific objective. In the context of humanitarian response, this objective is commonly related to the operation response time. The reviewed literature include Vehicle Routing Problems, Facility Location-allocation problems and other combinatorial mathematical formulations.

Given the size and complexity of these problems, researchers rely on mathematical programming to find optimal solutions. However, in many instances exhaustive search is intractable, so

alternative methods are carried out. These methods can be divided into exact solution methods, heuristics and metaheuristics.

4.2.1 Exact Solution Methods

These algorithms explore the search space until an optimal solution is found generally through a state space search, where different configurations of a problem are explored. Simplex and Branch and Bound methods are amongst the two most popular exact algorithms and will be the focus of this review.

Simplex Method

The simplex method is used to solve linear problem instances by converting the program into its canonical form (Griva et al. 2009). Once this is achieved, the algorithm explores the problem's feasible region by evaluating the extreme points. Geometrically, each constraint forms a boundary, such that a system of linear constraints creates a feasible region in the form of a polytope. Each vertex in this region is a basic feasible solution, and the algorithm navigates through the vertices until an optimal solution is found and, given that it is a linear program, the problem is convex meaning that it will eventually find the optimal solution.

The problem this solution method poses is that, while efficient, large problem instances become difficult to solve as the number of feasible solutions increases exponentially with the addition of more variables and constraints. Furthermore, the algorithm works for linear problems only, meaning that all non-linear conditions must be linearised for the Simplex method to function.

Branch and Bound

In contrast with the Simplex method, the branch and bound is used to solve integer and mixed integer problems (Brusco & Stahl 2006). This approach operates through the feasible region by splitting the space into smaller places (branching) and eliminating the areas of the search space where it can prove will not contain the optimal solution (bounding). A typical branch and bound approach will follow the next set of stages:

1. Initialisation: Find an initial solution to the optimisation problem and set it as the best solution thus far.
2. Populate the queue with all available nodes based on the starting solution.
3. Select a new node in the queue and delete it from the queue.
4. Determine the fitness of this node. If it is an improvement compared to the current best solution, set it as the current best. Otherwise, branch on the node and generate a new set of nodes to explore.
5. On the new nodes generated in the previous step, find the lower bound solution. If it is larger than the current best discard as they will never lead to the optimal. Otherwise store in the queue.
6. Steps 3-6 are repeated until all solutions are explored.

4.2.2 Heuristics

Heuristic algorithms are functions designed to solve a specific mathematical problem. In path searching problem, the A* algorithm is a heuristic which essentially works as a modified Dijkstra's (Dijkstra 1959, Hart et al. 1968), where an approximation function is used to estimate a node's distance to the objective. The TSP problem also has a number of heuristic methods (2-opt, 3-opt or nearest neighbour) that aims to find a close-to-optimal solution in a short time, and their performance is determined by a specified initial condition or parameter (Rosenkrantz et al. 2009). Due to their speed, these algorithms are normally paired with metaheuristic methods (see the next Section) in order to accelerate local search.

4.2.3 Metaheuristics

Metaheuristics are solution algorithms that can be utilised to solve multiple problem instances (Gendreau & Potvin 2010). They are purposely generic but can be modified to tackle specific problem instances. This review classifies the metaheuristic based on the structure of the solutions: single-solution, population-based, or swarm-based.

Single-solution methods

Single solution methods seek the improvement of a single solution at a time in each iteration. The algorithm holds the current best found solution, testing whether any modifications to it yield a better performing solution. In the case this happens, the original solution is replaced with the new best performing one. Depending on the parameters that control the process, these metaheuristics can be used for local or global search. Simulated annealing and neighbourhood search algorithms are two examples of single-solution methods.

Simulated annealing proposes a probabilistic process which determines whether the current best solution is substituted with a less optimal candidate solution. This aims to avoid local optima, and the probability of moving to a worse solution is commonly linked with the algorithm runtime. Another metaheuristic in this field is Tabu-search, where a list tabu list is maintained to prohibit the algorithm entering specific regions of the solution space to avoid being trapped in suboptimal regions. Neighbourhood search defines an area of the search space that can be obtained through the manipulation of a solution as a neighbourhood. Using this concept, the algorithm combines a neighbourhood change mechanism and a local search algorithm to find the final solution.

Several modifications to the neighbourhood search algorithms have been made to solve larger problem instances. Amongst the most popular is the large neighbourhood search created by (Shaw 1998) and increases the local improvement of the solutions using destroy and repair algorithms, meaning that the neighbourhoods explored are larger and can escape local optima more easily. As a result, it is an algorithm that has been used to solve multiple instances of the Vehicle Routing Problem and Location-Routing Problems.

Population-based methods

Population-based metaheuristics contain multiple solutions at each iteration, and their modification is generally affected by the rest of the population. These algorithms are used specifically in global search as they are able to explore different areas of the search space simultaneously through using the population. However, local search algorithms are generally able to find more optimal solutions if the search space is smaller as the disruption to the solution is smaller than

with population-based methods.

Genetic Algorithms provide an improvement method based by exploring different areas of the solution space and combining the accumulated knowledge of the population through crossover and mutation operators. An elitism mechanism is set to discard suboptimal regions of the space and focus on solutions that will likely lead to further improvements of the algorithm. Given its structure of carrying out evaluation, selection, crossover and mutation to each specific solution in the population, GA has the advantage over other algorithms of being easily parallelisable, accelerating its computational time. The GA is a very versatile algorithm and has been used to solve the vehicle routing problem, providing some of the best solutions to large instances of the problem.

While these can also be considered population-based methods, swarm-based methods employ self-organised agents instead of solely using population characteristics to drive the search. Particle Swarm Optimisation, for example, uses a combination of "personal" information or data exclusive to the agent, "social" information given by the population, and inertia or previous data to determine the modification to the solution. Ant-Colony optimisation creates "ant" agents that explore the problem instance. While navigating the problem, each agent keeps a memory of the path followed until a solution is found. Once all agents have found a solution, their performance is evaluated and the "pheromone" values of the problem are updated. This "pheromone" component effectively increases or decreases the chances of an agent to explore a certain region of the solution, and is based on the accumulated knowledge of all agents. This procedure is repeated until a termination criterion is satisfied.

4.3 Thesis structure

The aim of this thesis is to propose a quantitative framework for UAV-based humanitarian operations. Chapter 5 reviewed the field of humanitarian logistics and identified the preparedness and response stages as the key areas of study for the integration of UAV in humanitarian operations due to their rapid deployment capabilities and limited payload, making them most effective when used to transport low-volume high-value cargo.

Chapter 5 proposes a network design formulation coupled with a trajectory optimisation method to develop a framework for the use of UAVs to transport relief materials after a natural disaster. The problem is formulated as a location-routing algorithm as the deployment of UAV is highly dependent on the location of the depots and/or recharge stations due to their limited flight range.

While the network design component serves to plan the UAV operations, the framework did not consider the operational management of UAVs in the airspace. To overcome this limitation, Chapter 5 proposes a trajectory management framework based on the generation of airspace cluster to provide safe and real-time path planning capabilities. This approach is based on the ICAO PBN framework reviewed in Chapter 3 and is applicable to any UAV operation as the airspace clustering method is capable of isolating weather events, no-fly zones, and areas of excessive traffic.

Finally, while Chapter 5 proposed a relief transportation framework with UAVs, given their limited payload capacity humanitarian relief will be carried out using ground vehicles in the future. The main issue with this is their susceptibility to road transportation infrastructure damages, which not only lags the response operation but poses a significant safety concern to humanitarians. Satellite imagery not only takes significant time to be taken and transferred to the organisations, but it is expensive, has low quality and can be obscured by clouds. UAVs can be used to obtain high-quality data of the transportation system to detect available paths and inform vehicle routing decisions. Chapter 7 proposes an endogenous stochastic formulation to model the operation of UAVs for road damage assessment and simultaneous relief distribution using ground vehicles. The methods developed in this chapter can also be used in conjunction with the trajectory framework in Chapter 6 to provide operational paths and ensure the safety of the flights. The chapters together provide a holistic framework to operate UAV within the response phase of humanitarian logistics.

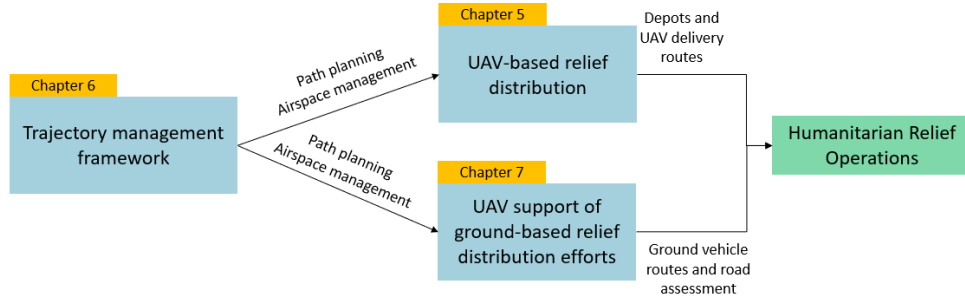


Figure 4.3: Methodological chapter relationship schematic.

4.4 Algorithm Selection Criterion

This section has reviewed different types of optimisation algorithms and discussed their applications. The exact solution solvers presented can determine the optimal solution to linear and integer problems but may result in large computational requirements due to the problem scale. Heuristics are reviewed and can solve problem instances in short computational time but are problem-specific. Metaheuristics instead provide generic solution methods, although their inherent structure and properties can make them more suitable to solve specific problem instances. This section documents the selection of solution method for the problems in this thesis.

The methodological chapters of this thesis will propose three different mathematical problems. Their inherent complexity (non-linear constraints and endogenous stochasticity) will require the use of metaheuristics to be solved. Chapter 2 introduces a Location-Routing algorithm, Chapter 6 proposes a clustering algorithm with a complex non-linear objective function, and Chapter 7 deals with a time-dependent endogenous stochastic Vehicle Routing Problem.

Location-Routing Algorithm

The location-routing algorithm, as described in Chapter 5, is the combination of two the Hub Location Problem (HLP) and Vehicle Routing Problem (VRP). The VRP is an NP-complete problem which requires the use of non-exact solution methods to solve any instance sizes of realistic proportions. Adding the complexity of the HLP only exacerbates the need for these algorithms. The HLP is commonly solved using a bi-stage process to reduce the problem size,

as the hub location decisions influence the possible solutions of the subsequent VRP.

While population-based approaches have been used to solve the LRP, neighbourhood search methods have solved several variants of the LRP (Şatir Akpunar & Akpınar 2021). This may be due to the bi-stage nature of the problem, where developing a population-based heuristic that solves the full problem instances becomes computationally inefficient. In contrast, most bi-stage approaches utilise some local search approach or heuristic combined with a neighbourhood method or metaheuristic. The use of local-search and heuristic-based search allows for rapid exploration of the search space and quick estimation of the neighbourhood, ensuring that the algorithm focuses on the areas of the search space where there is a higher probability for the final solution to be found.

For this reason, the approach proposed to solve the LRP in Chapter 5 involves a large neighbourhood search algorithm coupled with a heuristic algorithm to solve the bi-stage location routing problem.

Clustering

Clustering algorithms commonly utilise solely geographical metrics to evaluate their solutions. In such cases, the use of heuristics such as the k-means provide close-to-optimal solutions within a short computational time. However, the main limitation of most clustering heuristics relate to their inability to evaluate complex objective functions. In the cases where clustering must consider factors unrelated to the geographical position of data, these heuristics are insufficient.

To overcome this limitation, researchers introduce metaheuristics to improve the original solution given by a heuristic. In this context, several studies have been conducted to study the performance of metaheuristics for clustering, showing that the PSO outperforms most other tested algorithms (Cura 2012).

The performance of the PSO algorithm is related to the structure of the algorithm, in which each agent navigate the space directed by their current inertia, personal knowledge of the best solution, and swarm knowledge of the best solution. This momentum-based navigation of the search space maps seamlessly with the clustering problem, as the problem aims to locate the centroids of different collections of data.

Stochastic Vehicle Routing Problem

The vehicle routing problem has been extensively studied since its inception (Braekers et al. 2016). As discussed in Section 4.2, the GA has shown to be one of the best performing algorithms to solve the VRP and its iterations (Berger & Barkaoui 2003, Guezouli & Abdelhamid 2017). Furthermore, as the problem instance analysed in Chapter 7 is stochastic in nature, a Monte Carlo approach will be used to generate a large sample of problem instances to be evaluated. Thus, the easy parallelisation of the GA provides an additional advantage as it significantly reduces computational time.

4.5 Concluding Remarks

This chapter presented an overview of UAV flight dynamics for both hybrid UAVs by considering rotorcraft VTOL flight and fixed-wing cruise flight. The dynamics developed in this section are utilised in the coming chapters where the operational frameworks for the deployment of UAVs in humanitarian response is presented.

This chapter also reviews the optimisation theory literature, including exact algorithms and metaheuristics. A categorisation of metaheuristics based on their solution structure is given, and the advantages and disadvantages of each are presented. Furthermore, a selection criterion is presented based on the problems studied in Chapters 5, 6 and 7.

Chapter 5

Rapid Medical Deliveries Framework using Unmanned Aerial Vehicles

This chapter has been adapted from the following publication: Escribano J, Angeloudis P., Ochieng W. (2020) Optimal Hub Selection for Rapid Medical Deliveries using Unmanned Aerial Vehicles. *Transportation Research Part C: Emerging Technologies* 110, 56-80.

The lightweight nature, limited range and small payloads of current UAVs designs distinguish them from other modes of cargo delivery, and thus require the development of bespoke supply chain design models and novel operational frameworks. Chapter 2 revealed that a holistic approach to UAV-based humanitarian mission design, considering recharge station location to extend UAV mission range, is still to be developed.

To address this gap, this chapter develops a UAV-based humanitarian logistics mission design framework that can determine energy-efficient trajectories for hub-supported operations accounting for aerodynamic effects and payload variability. The resulting framework incorporates tactical hub-location planning, trajectory optimisation, and operational routing for UAVs considering payload capacity and split-delivery. Further aspects of energy storage and battery management are considered with the objective of minimising battery stocks at distribution depots while accounting for recharging and UAV waiting time.

The problem is defined as a dynamic MILP LRP and is solved using Large Neighbourhood Search algorithm. The latter is coupled with a static sub-problem and a route assignment

heuristic. Thus, the contributions of this chapter are as follows:

- It formalises the UAV-based humanitarian response planning process, which is evaluated using a composite objective function of cost, response time, energy use, and distribution equity as per the humanitarian principles of impartiality, neutrality, independence, and humanity followed by the UN agencies (WFP 2002, Bagshaw 2012).
- It is the first study to simultaneously consider aspects of hub location, battery allocation/recharge, split delivery and trajectory optimisation. Complex energy consumption patterns are incorporated that relate to payload variations, multiple flight stages and variable land topography. Such features were not previously considered in research literature, with associated models therefore resulting to overly conservative or optimistic flight plans.
- It presents a computationally efficient solution heuristic that can accommodate large UAV fleets. This is demonstrated with the included case study that focuses on a hypothetical UAV-based response to the 1999 Chi-Chi earthquake in Taiwan. Our analysis demonstrates that such a response mechanism could meaningfully support humanitarian response operations.

We introduce the modelling framework in Section 5.1. Section 5.2 describes the trajectory optimisation formulation, followed by the hub location-routing model in Section 5.3. A numerical case study focusing on a hypothetical UAV-based response to the 1999 Chi-Chi earthquake in Taiwan is presented in Section 5.4, with our results compared to those from previous studies. Conclusions and recommendations for future work are provided in Section 5.5.

5.1 Framework

The framework presented in this section aims to optimise humanitarian relief operations using UAVs following a natural disaster. Given a set of relief distribution centres, the algorithm determines vehicle trajectories, the number and location of charging stations, as well as vehicle itineraries that satisfy relief demand, while minimising mission completion time, cost and energy expenditure.

While time and cost reductions comprise conventional humanitarian objectives, energy expenditure is incorporated as a unique factor given the scarce availability of energy and limited access to UAV charging infrastructure. Additionally, a reduction of energy expenditure results in shortened charging times requirements. Alternative formulations are provided that consider distribution equity, a common concern in humanitarian relief literature (Holguin-Veras et al. 2013). We structure the framework into two independent decision stages that underpin mission design (Fig. 5.1): a flight and a humanitarian operations model.

The flight operations model (Section 5.2) focuses on the selection of aircraft trajectories, aerodynamic effects, varying payloads and energy requirements. We develop a direct trajectory optimisation model that estimates UAV endurance for a range of payloads, determines obstruction-free flight paths and considers the variable energy demands associated with changes in flight altitude. The algorithm minimises battery consumption using aerodynamic parameters, battery properties, and terrain topography as inputs. Unique to our approach is the consideration of multiple flight stages, exclusive to hybrid UAV deployments.

The humanitarian operations stage (Section 5.3) designs the relief cargo supply chain and selects optimal warehouse depots using aspects of hub selection and Green VRP, considering split delivery, battery replacement and battery allocation. Inputs to the model include the trajectories developed in the UAV operations stage, as well as the location and demand levels of nodes. Depot locations and delivery strategies are selected to minimise supply times and the overall duration of the humanitarian mission. The main contributions of our approach include the consideration of battery allocation and recharge, and simultaneous relocation of recharge stations. Additional performance indicators are used to evaluate the number of batteries used, number of recharges, average flight time, total energy expenditure, and inter-arrival times.

5.2 Trajectory Optimisation Model

The model presented in this section focuses on key decisions at the tactical stage of humanitarian UAV deployment. At its core lies a point-mass and force balance model, expressed as an optimal control problem (OCP), which aims to minimise battery consumption under geographical

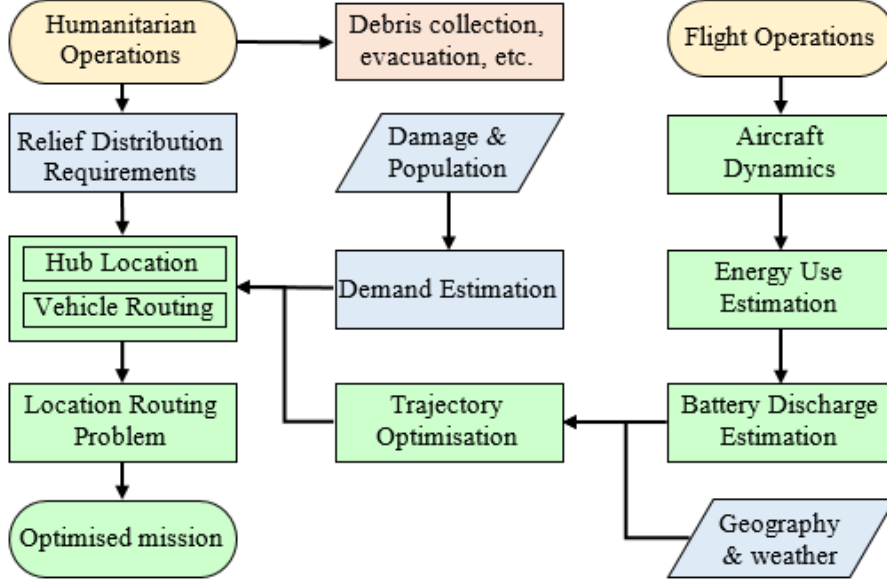


Figure 5.1: Framework flowchart indicating the relationship between the two decision stages (yellow blocks) in this study. Operational aspects specifically addressed in this study are highlighted in green, with inputs highlighted in blue, and external factors that influence our outputs in orange.

constraints of the terrain, considering both stages of hybrid UAV flight.

We use this model to determine energy usage, expressed as an alteration in the state of charge *SOC* of a battery with capacity B , and voltage V , for a UAV of mass m and with parasitic and polar drag coefficients of C_{d0} and K , respectively. The UAV engines have a collective efficiency parameter μ , and produce a thrust force T , to vary the instantaneous vehicle speed U , given a heading angle Ψ , and the pitch angle γ . We adopt the following assumptions and modelling simplifications:

Assumption 5.1: VTOL-capable hybrid UAVs are used for relief distribution, as per recent industry trends (Table 1.2). Such vehicles do not require runways for take-off and landing, therefore reducing reliance to supporting infrastructure. Furthermore, they can deliver cargo with a precision level that is not easily achievable with other designs (or air drops).

Assumption 5.2: The UAVs are powered by electric batteries. This assumption is based on our review of recent UAV logistics initiatives (Table 1.2) and limits vehicle weight fluctuations to only payload variations. Furthermore, battery-powered systems amount to 96% of the UAV market (Drone Industry Insights 2017). Nevertheless, we intend this formulation to be as generic as possible and capable of incorporating future UAV operational aspects (such as

fuel cells or fossil fuels and further aerodynamic and kinematic constraints).

Assumption 5.3: VTOL operations are carried out at pre-defined acceleration rates and low velocities. Following from Stoney (1993), we assume that the desired acceleration during VTOL is equal to $1.2mg$ (where m is vehicle mass and g the gravity coefficient), with deceleration fixed at $0.8mg$. Given the low velocities during VTOL, we consider the effects of air resistance to be insignificant during this stage and, therefore, are not accounted for in this paper. These assumptions have negligible impact on mission duration and overall energy consumption given the long distances to be covered by UAVs.

Assumption 5.4: Cruising is carried out between 50 and 70 metres above ground level, which provides sufficient clearance from most obstacles and ensures compliance with the majority of national drone regulations on maximum flight altitudes (100-150 metres) for low weight UAVs (Stöcker et al. 2017). Furthermore, a low flight altitude minimises the chances of interaction and interference with other aircraft, including military flights used for search and rescue that take place at 76 metres above ground (UK Government 2014).

Assumption 5.5: UAVs considered have an aggregate efficiency constant $\mu = 0.5$. Propeller efficiency varies continuously during flight and is dependent upon aircraft velocity and the propeller diameter, rotation speed (RPM) and design (Brandt & Selig 2011). Motor efficiency can reach 80% and depends on torque, rotation speed and electric current (Gottlieb 1997). Finally, battery performance will depend upon a broad range of factors, including cell chemistry, temperature, discharge rates and cycle numbers (Linden 1984). As the definition of accurate mathematical models for the above is beyond the scope of this chapter, we adopt an aggregate efficiency parameter of $\mu = 0.5$ that follows from previous work by D’Andrea (2014) and that we consider appropriate for UAVs deployed in humanitarian operations.

Assumption 5.6: The parasitic and polar drag coefficients are $C_{d0} = 0.015$ and $K = 0.03$ respectively. These values are influenced by vehicle and wing shape, air viscosity and compressibility effects (the latter negligible below sonic speed). Their numerical evaluation can be challenging (Hall 2016), so we adopt the values determined by Ostler et al. (2009) for similar UAVs.

Assumption 5.7: UAVs can operate Beyond Visual Line of Sight (BVLOS). At

the time of writing, EASA rules do not allow UAVs to operate BVOS without prior approval (EASA 2017a). The FAA authorises BVOS operations through an application process that requires 90 days to respond, which is excessive for humanitarian applications (FAA 2018). However, there is a record of such deployments in Rwanda (Simmons 2016), with WFP-led efforts to develop regulations for UAV humanitarian response still in progress at the time of writing (Vornic 2017).

Assumption 5.8: The difference between true and indicated airspeeds is negligible. We considered true airspeed (actual movement of the aircraft relative to the air mass) to be equal to indicated airspeed (measured speed of the vehicle). The impact of this assumption is minimal, as vehicles will travel at low speeds (below sonic speed) and air density variations with sea level can be ignored (Gracey 1980). This assumption excludes the impact of weather effects and wind speeds, which cause a reduction in UAV performance (Brandt & Selig 2011). However, we assume that UAVs will be deployed under favourable climate conditions, so the potential effects of wind are minimal.

5.2.1 Model Formulation

We develop a trajectory OCP model from first principles. The derivation of underlying aerodynamic, kinematic and battery consumption relationships is provided in Chapter 4. Our model uses a set of state and control variables that alter the system state. Only minimal changes to heading angles (Ψ) are performed given our objective to minimise battery consumption. The following notation conventions are used for the formulation of the OCP model.

Indices		Sets	
t	= Time step	$\mathcal{T}$	= Time step set
v	= Vehicle (UAVs)	$\mathcal{V}$	= Vehicle set
State Variables		Control Variables	
U	= Airspeed [metres/second]	T	= Thrust force [Newtons]
SOC	= State of charge [-]	Ψ	= Heading angle [radians]
x, y, z	= Cartesian coordinates [km]	γ	= Pitch angle [rads]
C_L	= Lift coefficient [-]		
C_D	= Drag coefficient [-]		
s	= Path travelled [km]		
Force Parameters		Battery Parameters	

D	=	Drag force [metres/second]	P	=	Power [Watts]
L	=	Lift [N]	I	=	Current [Amps]
W	=	Weight [N]	V	=	Voltage [Volts]
m	=	Drone Mass (with payload) [kg]	B	=	Battery capacity [Amps-Hour]
g	=	Acceleration due to gravity [m/s ²]	Δt	=	Time step [s]
a	=	Acceleration [m/s ²]	μ	=	Aggregate efficiency [-]

Aerodynamic Parameters

K	=	Drag polar [-]
ρ	=	Air density [kg/m ³]
S	=	Wing area [m ²]
q	=	Dynamic pressure: $\frac{1}{2}\rho U^2$ [kg/ms ²]
C_{D0}	=	Parasitic drag coefficient [-]

The objective of the model is aims to maximise the state of charge by the end of the flight, with SOC values of 1 and 0 representing a fully charged and depleted battery, respectively.

Objective:

$$\text{Minimise } 1 - SOC_v(t_f) \quad \forall v \in \mathcal{V} \quad (5.1)$$

Subject to:

$$ds_v/dt = U_v(t) \cos(\gamma_v(t)) \quad \forall v \in \mathcal{V} \quad (5.2)$$

$$dz_v/dt = U_v \sin(\gamma_v(t)) \quad \forall v \in \mathcal{V} \quad (5.3)$$

$$dU_v/dt = (T_v(t) - D_v(t) + L_v(t) \sin(\gamma_v(t)))/m_v \quad \forall v \in \mathcal{V} \quad (5.4)$$

$$D_v(t) = qSC_{D_v}(t) \quad \forall v \in \mathcal{V} \quad (5.5)$$

$$C_{D_v} = C_{D0} + KC_{L_v}^2 \quad \forall v \in \mathcal{V} \quad (5.6)$$

$$C_{L_v} = L_v(t) + KC_{L_v}^2 \quad \forall v \in \mathcal{V} \quad (5.7)$$

$$L_v = m_v g / \cos(\gamma_v(t)) \quad \forall v \in \mathcal{V} \quad (5.8)$$

$$dSOC_v/dt = -T_v(t)U_v(t)/\mu BV \quad \forall v \in \mathcal{V} \quad (5.9)$$

$$|\gamma_v| < \frac{\pi}{2}, \quad 0 \leq T_v \leq T_{max}, \quad U_v = U_{cr}, \quad SOC_v \leq 1 \quad \forall v \in \mathcal{V} \quad (5.10)$$

$$f(x, y) + z_{min} \leq z_v \leq f(x, y) + z_{max}$$

The following boundary conditions are assumed:

$$\text{Initial: } s_v(t_0) = s_{0_v}, \quad z_v(t_0) = z_{0_v}, \quad SOC_v(t_0) = 1, \quad \gamma_v(t_0) = \gamma_{0_v} \quad \forall v \in \mathcal{V} \quad (5.11)$$

$$\text{Final: } s_v(t_f) = s_{f_v}, \quad z_v(t_f) = z_{f_v}, \quad SOC_v(t_f) \text{ free}, \quad \gamma_v(t_f) \text{ free} \quad \forall v \in \mathcal{V} \quad (5.12)$$

Equations (5.2-5.3) represent the movement of the UAV along the path s at altitude z , governed by the pitch angle γ . Equation (5.4) is derived from the balance of horizontal forces acting upon the UAV:

$$ma = T - D + L \sin(\gamma) = T - qSC_D + mg \tan(\gamma) \quad (5.13)$$

Equation (5.5) determines the drag force exerted on the UAV as a function of the drag coefficient C_D , which is described in constraint (5.6) as a function of the drag polar K , parasite drag coefficient C_{D0} and lift coefficient C_L . C_L is proportional to the UAV mass m as indicated by equations (5.7) and (5.8), where the vertical component of the lift force L can be assumed to equate to the weight force W . Finally, constraint (5.9) describes battery consumption, with an extended derivation provided in Appendix C.

The equations in (5.10) define the rest of variable constraints. We limit the pitch angle γ to 90 degrees and assume that at any time the thrust T is positive and lower than the maximum level of thrust provided by the motors. The target speed U_{cr} is predetermined and fixed throughout the cruise stage, but allowed to vary during ascend and descend given a stall speed U_{st} such that $U_{st} \leq U_v \leq U_{cr}$. We also consider a variable terrain geography, described by a function $f(x, y)$ and assuming a fixed ground clearance z_{min} . Additional boundary conditions enforce pitch angle γ values on the initial and final points of the trajectory. UAVs are assumed to loiter and decelerate once they reach their destinations.

The finite difference method (Grossmann & Roos 2007) is used to discretise the problem into elements, therefore transforming the OCP into the following NLP:

Objective:

$$\text{Minimise } 1 - SOC_{vt_f} \quad \forall v \in \mathcal{V} \quad (5.14)$$

Subject to:

$$s_{v,t+1} = s_{v,t} + \Delta t U_{v,t} \cos(\gamma_{v,t}) \quad \forall v \in \mathcal{V}, \quad \forall t \in \mathcal{T} \quad (5.15)$$

$$z_{v,t+1} = z_{v,t} + \Delta t U_v \sin(\gamma_{v,t}) \quad \forall v \in \mathcal{V}, \quad \forall t \in \mathcal{T} \quad (5.16)$$

$$U_{v,t+1} = U_{v,t} + (T_{v,t} - D_{v,t} + L_{v,t} \sin(\gamma_{v,t}))/m_v \quad \forall v \in \mathcal{V}, \quad \forall t \in \mathcal{T} \quad (5.17)$$

$$D_{v,t} = qSC_{D_{v,t}} \quad \forall v \in \mathcal{V} \quad \forall t \in \mathcal{T} \quad (5.18)$$

$$C_{D_{v,t}} = C_{D0} + KC_{L_{v,t}}^2 \quad \forall v \in \mathcal{V}, \quad \forall t \in \mathcal{T} \quad (5.19)$$

$$C_{L_{v,t}} = L_{v,t} + KC_{L_{v,t}}^2 \quad \forall v \in \mathcal{V}, \quad \forall t \in \mathcal{T} \quad (5.20)$$

$$L_{v,t} = m_{v,t}g / \cos(\gamma_{v,t}) \quad \forall v \in \mathcal{V}, \quad \forall t \in \mathcal{T} \quad (5.21)$$

$$SOC_{v,t+1} = SOC_{v,t} - \Delta t T_{v,t} U_{v,t} / \mu BV \quad \forall v \in \mathcal{V}, \quad \forall t \in \mathcal{T} \quad (5.22)$$

$$|\gamma_{v,t}| < \frac{\pi}{2}, \quad 0 \leq T_{v,t} \leq T_{max}, \quad U_{v,t} = U_{cr}, \quad SOC_{v,t} \leq 1 \quad \forall v \in \mathcal{V}, \quad \forall t \in \mathcal{T} \quad (5.23)$$

$$f(x, y) + z_{min} \leq z_{v,t} \leq f(x, y) + z_{max}$$

Constraints (5.15-5.23) follow from (5.2-5.10), with the number of discrete steps t in T being

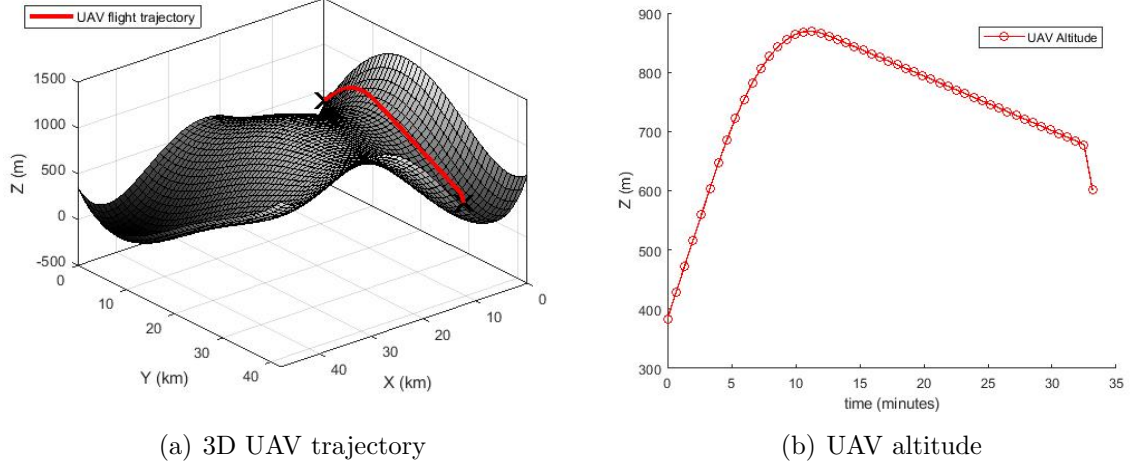


Figure 5.2: Example trajectory between the Hsin Yi and Ren Ai regions of the Nanto County in Taiwan.

Table 5.2: Flight stages during trajectory optimisation.

Flight Stage	Controls	State Variables	Thrust Use	Condition
Take-Off	T	U, μ, g, V, B	$T = (m + p)(a - g)$	$a > mg$
Climb	T, γ	$s, z, C_{D0}, U, \mu, g, V, B, q, S$	$T = D + (m + p)\dot{U} - L \sin \gamma$	$\gamma > 0$
Cruise	T, γ	$s, z, C_{D0}, U, \mu, g, V, B, q, S$	$T = D + (m + p)\dot{U} - L \sin \gamma$	$\gamma = 0$
Descend	T, γ	$s, z, C_{D0}, U, \mu, g, V, B, q, S$	$T = D + (m + p)\dot{U} - L \sin \gamma$	$\gamma < 0$
Landing	T	U, μ, g, V, B	$T = (m + p)(a - g)$	$a < mg$

a compromise between computational time and trajectory detail. We implemented this model using the Pyomo.dae toolkit and the IPOPT optimisation engine (Hart et al. 2011, 2012). An illustrative trajectory output from our implementation is presented in Fig. 5.2.

Using the above NLP model, we can determine optimal UAV routes with respect to battery consumption, costs and flight durations while acknowledging terrain morphology and payload variations. These are provided as inputs to the humanitarian operations stage of the model, described in the following section.

5.3 Depot location and routing for UAV operations

The humanitarian operations stage focuses on the determination of optimal warehouse locations and UAV itineraries, modelled using a hub selection-routing problem specific for UAV operations. Our proposed model combines aspects of discrete un-capacitated hub location selection and time-dependent Green VRP design with multiple hubs, split deliveries, and battery replacements

during depot visits. Our initial objective is to minimise the number of hubs and required mission time for the distribution of a predefined amount of relief cargo. We adopt the following assumptions and modelling conventions:

Assumption 5.9: A unitised commodity type is provided throughout the operation. Relief operations rely on standardised supply kits that are developed for specific incident type (earthquake, flooding, displacement), demography (diet, religion) and climate combinations. This practice is used to simplify the logistics of disaster response, as only a single commodity needs to be stocked, transported and distributed (WFP 2017a).

Assumption 5.10: Battery replacement times are considered negligible. We consider battery replacement times to be negligible compared to overall mission durations. This follows from previous work on the design of autonomous systems with instantaneous replacements (Michini et al. 2011, Suzuki et al. 2012, Fujii et al. 2013) and the availability of UAV models with easily accessible battery compartments, such as eBee (Sensefly 2014) and DeltaQuad (Vertical Technologies 2017).

Assumption 5.11: Hubs are un-capacitated. This assumption was made given the small payloads of UAVs in comparison to the scale of humanitarian response operations, typically involving several tonnes of relief cargo (Sheu 2007, Chang et al. 2014). Therefore, it is safe to assume that the hubs to be used provide sufficient storage capacity to undertake the full UAV response. The following notation is used:

5.3.1 Mathematical Model

Indices	Sets
i, j = nodes	$\mathcal{N}$ = node set
t = mission time steps	$\mathcal{D}$ = demand node set $\mathcal{D} \subset \mathcal{N}$
v = vehicles (UAVs)	$\mathcal{H}$ = hub set $\mathcal{H} \subset \mathcal{N}$
b = battery	$\mathcal{T}$ = mission time set
	V = vehicle set
	B = battery set
Parameters	Variables
F_i = Hub setup costs (fixed) [\$]	r^{ijvt} = Boolean: v travels from i to j at time t
Pen = Delivery penalty per demand unit [\$/kg]	c^{ijvt} = Integer: cargo units carried from i to j by vehicle v at time t

Q	= Vehicle payload capacity [kg]	w^{ijbt}	= Boolean: b carried from i to j at time t
C_{ij}	= Travel time costs from i to j [\$/hr]	h_i	= Boolean: node i hub status
A_i	= Aid demand at node i [kg]	u_i	= Integer: unfulfilled demand at node i
S_i	= Aid supply at node i [kg]	e_{bt}	= Fraction: battery charge at time t for b
E_{ij}	= Percentage battery consumption for travel between i and j [-]	a_b	= Boolean: battery b availability status
E_r	= Percentage battery recharge rate per unit time [-]		
G	= Cost of battery [\$/]		
Δt_{ij}	= Travel time from i to j [\$/]		

Given the multi-objective nature of this problem, we initially adopt a cost-based objective function 5.24 that minimises the total cost Z of relief distribution (5.25-5.28).

$$Z = HC + RC + BC + PC \quad (5.24)$$

$$HC = \sum_{i \in \mathcal{H}} h_i F \quad (5.25)$$

$$RC = \max_{v \in \mathcal{V}} \left(\sum_{t \in \mathcal{T}, i, j \in \mathcal{N}} r_{ijvt} C_{ij} \right) \quad (5.26)$$

$$BC = \sum_{b \in \mathcal{B}} a_b G \quad (5.27)$$

$$PC = \sum_{i \in \mathcal{N}} |u_i| Pen \quad (5.28)$$

Equation (5.25) considers the costs of setting up a depot in node i . We use a per-depot cost F across all nodes, which incorporates material and construction costs. Equation (5.26) captures the total mission duration, which is defined as the period from UAV deployment to mission completion (all cargo is delivered, and UAVs return to base). Given that all UAVs are deployed simultaneously, mission duration is calculated as the maximum flight and wait time among the UAV fleet. The trajectory model in Section 5.2 is used to calculate the flight costs C_{ij} between every node pair i, j .

Equation (5.27) considers the cost G of using battery b . An increased number of batteries ensures that the response time is minimised, as it allows the immediate replacement of batteries when a UAV returns to the hub. Finally, we introduce the Pen parameter, which acts as a numerical penalty (equation 5.28) attributed to a missed, or excess delivery of planned relief cargo units, which forces the model to prioritise mission completion. Therefore, this penalty

value can be seen to reflect the material waste and potential loss of life that can occur as a result of excess and missed deliveries, respectively. The formulation of the model is as follows:

$$\text{Minimise } Z \quad (5.29)$$

Subject to:

$$\sum_{j \in \mathcal{N}: t - \Delta t_{ji} \in \mathcal{T}} r_{jiv(t - \Delta t_{ji})} - \sum_{j \in \mathcal{N}: t - \Delta t_{ji} \in \mathcal{T}} r_{ijvt} = 0 \quad \forall i \in \mathcal{N}, \forall v \in \mathcal{V}, \forall t \in \mathcal{T} \quad (5.30)$$

$$r_{iivt} = 0 \quad \forall i \in \mathcal{D}, \forall v \in \mathcal{V}, \forall t \in \mathcal{T} \quad (5.31)$$

$$\sum_{t \in \mathcal{T}, v \in \mathcal{V}, j \in \mathcal{N}} r_{jivt} - \sum_{t \in \mathcal{T}, v \in \mathcal{V}, j \in \mathcal{N}} r_{ijvt} = 0 \quad \forall i \in \mathcal{N} \quad (5.32)$$

$$\sum_{t - \Delta t_{ij} \in \mathcal{T}, i, j \in \mathcal{N}} r_{ijvt} = 1 \quad \forall v \in \mathcal{V} \quad (5.33)$$

$$\sum_{i \in \mathcal{H}, j \in \mathcal{N}} r_{ijvt_0} = 1 \quad \forall v \in \mathcal{V} \quad (5.34)$$

$$\sum_{i \in \mathcal{N}, j \in \mathcal{H}} r_{ijvt_f} = 1 \quad \forall v \in \mathcal{V} \quad (5.35)$$

$$\sum_{t \in \mathcal{T}, v \in \mathcal{V}, j \in \mathcal{N}} c_{jivt} - \sum_{t \in \mathcal{T}, v \in \mathcal{V}, j \in \mathcal{N}} c_{ijvt} + u_i = (1 - h_i)A_i - h_i S_i \quad \forall i \in \mathcal{N} \quad (5.36)$$

$$r_{ijvt} Q \geq c_{ijvt} \quad \forall i, j \in \mathcal{N}, \forall v \in \mathcal{V}, \forall t \in \mathcal{T} \quad (5.37)$$

$$\sum_{j \in \mathcal{N}: t + \Delta t_{ij} \in \mathcal{T}} c_{jivt} - \sum_{j \in \mathcal{N}: t + \Delta t_{ij} \in \mathcal{T}} c_{ijv(t + \Delta t_{ij})} \geq 0 \quad \forall i \in \mathcal{D}, \forall v \in \mathcal{V}, \forall t \in \mathcal{T} \quad (5.38)$$

$$\sum_{b \in \mathcal{B}} a_b w_{ijbt} = r_{ijvt} \quad \forall i, j \in \mathcal{N} : i \neq j, \quad t \in \mathcal{T} \quad (5.39)$$

$$\sum_{j \in \mathcal{N}: t + \Delta t_{ij} \in \mathcal{T}} w_{jibt} - \sum_{j \in \mathcal{N}: t + \Delta t_{ij} \in \mathcal{T}} w_{ijb(t + \Delta t_{ij})} = 0 \quad \forall i \in \mathcal{N}, \forall b \in \mathcal{B}, \forall t \in \mathcal{T} \quad (5.40)$$

$$w_{iibt} = 0 \quad \forall i \in \mathcal{D}, \forall b \in \mathcal{B}, \forall t \in \mathcal{T} \quad (5.41)$$

$$\sum_{i \in \mathcal{H}, j \in \mathcal{N}} w_{ijvt_0} = 1 \quad \forall b \in \mathcal{B} \quad (5.42)$$

$$\sum_{i \in \mathcal{N}, j \in \mathcal{H}} w_{ijvt_f} = 1 \quad \forall b \in \mathcal{B} \quad (5.43)$$

$$e_{bt_0} = a_b \quad \forall b \in \mathcal{B} \quad (5.44)$$

$$e_{b(t+1)} = e_{bt} + a_b \sum_{i \in \mathcal{H}} E_r w_{ijbt} - a_b \sum_{i, j \in \mathcal{N}} E_{ijt} w_{ijbt} \quad \forall b \in \mathcal{B}, \forall t \in \mathcal{T} \quad (5.45)$$

$$\sum_{i \in \mathcal{N}} h_i \geq 1 \quad (5.46)$$

$$0 \leq e_{bt} \leq 1 \quad \forall b \in \mathcal{B}, \forall t \in \mathcal{T} \quad (5.47)$$

$$r_{ijvt} \in \{0, 1\}, \quad c_{ijvt} \in \mathbb{Z}^+ \quad \forall i, j \in \mathcal{N}, \quad v \in \mathcal{V}, \quad \forall t \in \mathcal{T} \quad (5.48)$$

$$h_i \in \{0, 1\}, \quad u_i \in [-S_i, A_i] \quad \forall i \in \mathcal{N} \quad (5.49)$$

$$w_{ijbt} \in \{0, 1\} \quad \forall b \in \mathcal{B} \quad \forall t \in \mathcal{T} \quad (5.50)$$

$$a_b \in \{0, 1\} \quad \forall b \in \mathcal{B} \quad (5.51)$$

Constraint (5.30) ensures that vehicles entering a node leave the next time step after arrival (estimated using Δt_{ij}). Vehicles can wait at a hub if necessary, a condition defined as $r_{iivt} = 1 \quad \forall i \in \mathcal{H}$, but cannot do so at a demand node as expressed in (5.31). Equation (5.32) enforces the conservation of vehicle flows over the mission, with the number of vehicles entering and exiting every node being equal. Constraint (5.33) ensures that vehicle flows remain unassigned until a trip between i and j is completed. Equations (5.34-5.35) indicate that vehicles must start and end the mission at a hub.

Constraint (5.36) enforces a mass balance of cargos at each node and ensures that cargo is transferred from a hub (cargo from node i is negative, $-c_{ijvt}$) to a demand node (cargo arriving at i is positive c_{jivt}). Any excess or non-delivery is captured by variable u_i . Constraint (5.37) limits cargo assignment to vehicle flows ($r_{ijvt} = 1$) with each carrying a maximum capacity Q . (5.38) prevents cargo collection from demand nodes. This constraint, in conjunction with equation (5.37), limits vehicles to start and end tours at hubs, firstly to carry material to the desired demand node, and secondly to collect the payload for the next tour.

Equations (5.39-5.45) explore the behaviour and management of batteries. Constraint (5.39) limits battery assignment to a vehicle route. Constraints (5.40-5.41, 5.42-5.43) are analogous to (5.30-5.31, 5.34-5.35) and ensure that batteries start and end the mission at a hub and are assigned a new destination after arrival at a node. As with vehicular flows, batteries can remain at a hub ($w_{iibt}=1 \quad \forall i \in \mathcal{H}$). Equation (5.42) initiates battery charge to 1 (fully charged) if the battery is activated ($a_b = 1$), and 0 (empty) otherwise. The following constraint (5.45) captures battery charge and discharge respectively, where $E_{i,j}$ is derived from the trajectory model presented in Section 5.2. The battery charges only if it is not used at a time step and remains at a hub, otherwise it discharges.

Equation (5.46) ensures that at least one hub is selected. Constraint (5.47) bounds battery charge between 1 (full) and 0 (empty). Equations (5.48-5.49) limit the decision variables c_{ijvt} , u_i and slack variables s_i , t_i to integer values, while r_{ijvt} and h_i are limited to Boolean values. Finally, (5.50-5.51) define w_{ijbt} and a_b as Boolean values.

The main limitation of the above formulation is the considerable number of decision variables

needed, resulting in long computational time requirements for any instance of practical size. The model was implemented using the Pyomo toolkit and solved using a Branch and Bound algorithm provided by the CPLEX solver, on a workstation with an Intel Xeon E5-1650 CPU and 32GB RAM. Only a few instances that did not account for battery management (equations 5.39-5.45) were solvable using this implementation. Therefore, the above formulation is impractical for humanitarian operations planning, where time and resources are limited.

5.3.2 Sub-problem

To overcome the computational limitations of the above model, we develop an approximate multi-stage heuristic. In the first instance, a static, integer-based hub-flow assignment model is incorporated, which is a relaxation of the VRP-specific aspects of the original formulation:

$$\text{Minimise } Z = HC + RC + PC \quad (5.52)$$

$$HC = \sum_{i \in \mathcal{N}} h_i F \quad (5.53)$$

$$RC = \sum_{i, j \in \mathcal{N}} r_{ij} C_{ij} \quad (5.54)$$

$$PC = \sum_{i \in \mathcal{N}} |u_i| Pen \quad (5.55)$$

Subject to:

$$\sum_{j \in \mathcal{N}} r_{ji} - \sum_{j \in \mathcal{N}} r_{ij} = 0 \quad \forall i \in \mathcal{N} \quad (5.56)$$

$$\sum_{j \in \mathcal{N}} c_{ji} - \sum_{j \in \mathcal{N}} c_{ij} + u_i = (1 - h_i)A_i - h_i S_i \quad \forall i \in \mathcal{N} \quad (5.57)$$

$$r_{ij} Q \geq c_{ij} \quad \forall i, j \in \mathcal{N} \quad (5.58)$$

$$\sum_{i \in \mathcal{N}} h_i \geq 1 \quad (5.59)$$

$$r_{ij} \in \mathbb{Z}^+, \quad c_{ijvt} \in \mathbb{Z}^+ \quad \forall i, j \in \mathcal{N} \quad (5.60)$$

$$h_i \in \{0, 1\}, \quad u_i \in [-S_i, A_i] \quad \forall i \in \mathcal{N} \quad (5.61)$$

The revised objective function continues to use costs to describe the various operational aspects of the problem. Equation (5.53) determines the total upfront costs for the establishment of a hub at a node i , following from (5.25). Equation (5.54) captures aggregate travel times across the entire fleet. Given that individual UAV trips are no longer trackable, total response

Table 5.4: Model performance for CPLEX. Solutions were not obtained for missions with durations within 24-hour execution window

Instance	Demand (kg)	Drones	Payload (kg)	Mission Duration (mins)	Cargo (kg)	Hub	Runtime (s)
4 nodes	525	15	2	50	180	1	0.23
				100	360	1	0.40
				150	503	1	2.75
			3	50	270	1	0.57
				100	-	-	-
				150	525	1	1.75
	900	25	2	50	300	1	0.23
				100	360	1	0.51
				150	838	1	1.04
			3	50	450	1	0.67
				100	-	-	-
				150	900	1	0.71
	1200	35	2	50	420	1	0.25
				100	-	-	-
				150	900	1	0.71
			3	50	630	1	0.59
				100	-	-	-
				150	1200	1	0.98
7 nodes	800	15	2	50, 100, 150	-	-	-
			3	50, 100, 150	-	-	-
	1100	25	2	50, 100, 150	-	-	-
			3	50, 100, 150	-	-	-
	1200	35	2	50, 100	-	-	-
			3	150	1200	1	2.20
10 nodes	1450	15	2	50, 100, 150	-	-	-
				50	300	2	1.39
				100	600	2	2.70
			3	150	862	2	18.23
				50	450	2	1.32
				100, 150	-	-	-
	1450	25	2	50	500	2	1.39
				100	950	2	2.65
				150	1314	2	5.42
			3	50	450	1	1.32
				100, 150	-	-	-
				50, 100, 150	-	-	-
	1650	35	2	50, 100, 150	-	-	-
			3	50, 100, 150	-	-	-

time is not calculated. Equation (5.55) tracks any penalties associated with undelivered and excess cargo, following from (5.28). Battery management is not considered in this formulation.

The removal of time and vehicle dimensions allows us to perform several simplifications to the constraints of the model, with battery management and energy monitoring being considered in a separate sub-problem. Constraints (5.56) and (5.57) ensure vehicle and cargo balance respectively. The remaining constraints are carried over from the previous formulation: (5.58) limits the transport of cargo to a maximum allowed payload, and (5.59) enforces the creation of a hub. Finally, the decision variable bounds in (5.60-5.61) are analogous to equations (5.48-5.49). However, in this problem instance r_{ij} is an integer that indicates the number of vehicles travelling from i to j .

The resulting aggregate flow outputs r_{ij} and c_{ij} represent a lower-bound solution to the original problem and act as proxies for fleet size and cargo volumes to be transported across the network. The results are used as inputs to a route assignment sub-problem presented in the following section.

The main advantage of the proposed algorithm is that computational time no longer depends on the number of vehicles, but only on the number of nodes and demand satisfaction requirements. Furthermore, the simplified model described in equation set (5.52-5.61) is solved using exact methods. This allows us to apply the algorithm to large UAV fleet sizes envisioned that otherwise would limit applicability to small problem instances.

5.3.3 Route Assignment

The approximate algorithm presented in this section builds upon earlier work by Özdamar (2011) and Dorling et al. (2017) and transforms downscaled aggregate flows into individual UAV itineraries. The algorithm comprises five distinct decision stages, route continuity, cargo assignment, energy monitoring, vehicle assignment, and battery distribution, as structured in Fig. 5.3.

Our main contribution with respect to the previous algorithms is the inclusion of energy monitoring and battery assignment steps that monitor energy consumption of each vehicle considering variable payload weight and allocate battery resources given existing recharge station

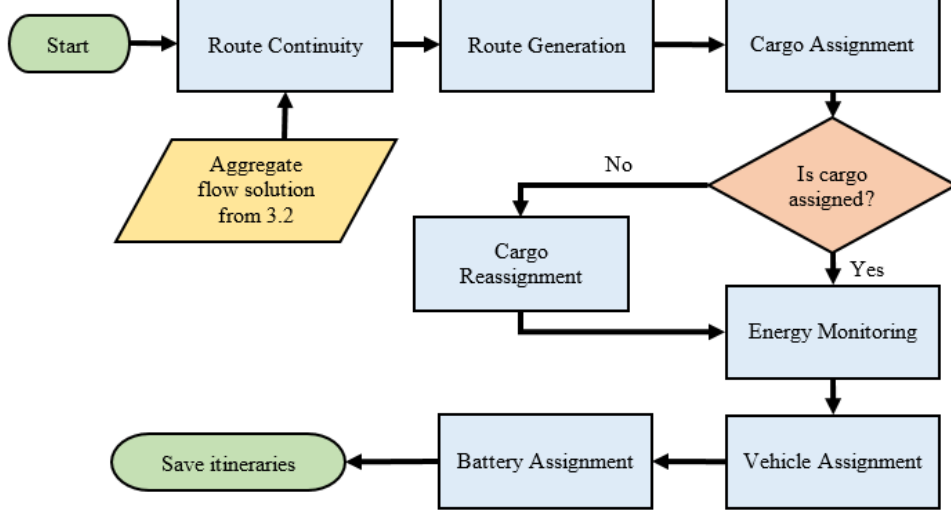


Figure 5.3: Route assignment flowchart.

location. Given the inclusion of these aspects, we consider our approach to be considerably more practice-ready compared to previous studies. We describe the approach adopted at each stage below and provide the detailed algorithms in Appendix C.

Route creation and continuity: This stage generates vehicle tours that comply with constraints (5.30-5.35). Given that the initial aggregate flows satisfy constraint (5.56), they comply with vehicle balance in accordance to (5.60). Once the total number of routes operating from each hub is determined, the algorithm identifies origin-destination (O-D) pairs with unequal bi-directional flows. From these, excess flows are extracted to derive the “imbalanced” and “balanced” flow matrices. For each matrix, the algorithm iterates through outgoing vehicle flows in hubs until it encounters a node with no onward flows, prioritising nodes that connect to a hub ($r_{ij} > 0 \ \forall j \in \mathcal{H}$). The “balanced” matrix is then used to direct UAVs towards the tour origin, while the “imbalanced” matrix directs vehicles randomly to a hub, given the remaining flows $r_{i,j}$.

Cargo assignment: In this stage cargo flows are assigned to vehicle tours, considering indicative flow optimality c_{ij} , vehicle capacity Q , and total unmet demand of any nodes visited, ensuring compliance with constraints (5.36-5.38). Once all routes are assigned cargo, the algorithm checks whether any material flow c_{ij} remains unassigned and calculates whether demands for relief aid have been satisfied. If this is not the case, the algorithm activates a reassignment process that identifies positive or negative deviations from a set of demand targets

and adjusts cargo flow assignments.

Energy monitoring: Once route continuity and cargo assignments are addressed, the energy monitoring stage which ensures compliance with battery capacity constraints (5.39-5.45). Battery consumption is calculated at every tour node, diminishing for subsequent stops due to payload variations. Should energy consumption rates in a tour exceed battery capacities, a battery replacement tour is introduced ahead of the tour leg that violates capacity. In the first instance, the algorithm opts for the nearest hub to the last feasible node. If this is too far to ensure compliance, the original route is used with the stop at an earlier step. This step is repeated for any future tour legs that violate battery capacities.

Vehicle assignment: This stage evaluates tour durations, which are then assigned to vehicles while minimising overall mission duration. Initial UAV allocations are proportional to tour durations from each hub. Trip assignments are then determined with the objective of minimising flight and relocation times (where UAVs travel between two hubs to collect cargo). As part of this process, total flight times for each tour are estimated, and the shortest routes are assigned while accounting for any required relocation times. This procedure is repeated until all tours have been assigned.

Battery assignment: Following tour allocations, batteries are distributed across the network using a heuristic that iterates across every time step of the time horizon, while keeping track of vehicle locations. Batteries are assigned to departing vehicles, their original hubs, and any intermediate battery swap points along their routes. Batteries assigned to UAVs are assumed to be fully charged, with the remaining being charged at a rate of 1C. A coefficient is defined for each battery that estimates any incurred delay created when the battery is not used, due to the need for UAVs to wait until other batteries are charged. The algorithm then proceeds to discard the battery with the smallest coefficient and estimates the overall mission durations and final battery costs as defined by equations (5.26) and (5.27). The process continues until total mission costs begin to increase (due to route infeasibility and incurred non-delivery penalties).

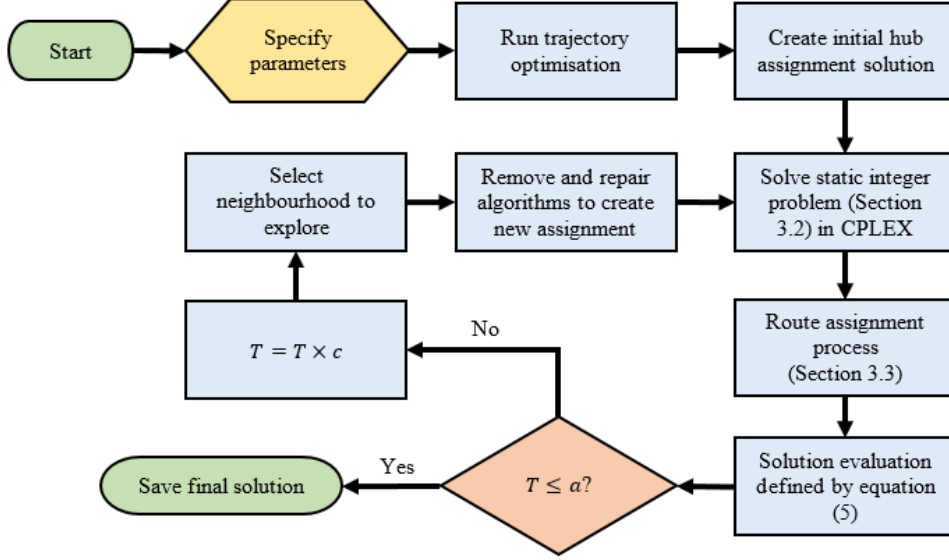


Figure 5.4: Model Workflow

5.3.4 Metaheuristic solution algorithm

The algorithms outlined in sections 5.3.1 and 5.3.2 were implemented using the C# language and embedded within a custom Large Neighbourhood Search (LNS) metaheuristic that was developed as part of this study. The latter navigates through a constrained search space using probabilistic search (Pisinger & Ropke 2010) and, unlike Genetic Algorithms, is capable of simultaneously exploring multiple neighbourhoods that reside in different branches of the search space. This ability to avoid local optima is particularly useful for greedy stochastic algorithms (such as those presented in section 5.3.3). The overall workflow of the algorithm is presented in Fig. 5.3.4 and Algorithm listing 1.

The search commences with a randomly generated solution-neighbourhood N_i . A roulette selection stage then determines which solution to manipulate for every subsequent iteration, ensuring that all solutions can be explored. Selected solutions are manipulated using custom destroy and repair algorithms (chosen with probability rates α and β , respectively).

The overall progress of the search is monitored using a “temperature” metric T . This represents the search space that can be explored from the best-known solution and decreases with every iteration of the LNS. The search process and its behaviour is controlled by two parameters, “cooling rate” c (between 0 and 1) that defines the temperature T change rate at each step, and “absolute temperature” a that defines the minimum allowed temperature and

terminates the search process once this is reached.

In every iteration, a parent solution s_1 is evaluated against a newly generated candidate solution s_2 using a scalar fitness value, f_{s_1} and f_{s_2} , for each solution. If f_{s_2} is higher than f_{s_1} , s_2 replaces s_1 in neighbourhood N . Otherwise, either a) s_1 is replaced with s_2 , or b) s_2 is discarded, and a new solution is generated. Replacing the solution with an inferior one may appear counter-intuitive but it ensures that the algorithm is not limited by a local optimum. The probability P of a worse solution s_2 replacing an original solution s_1 is defined using the difference in fitness of both solutions F and the expression:

$$P = \exp(-F/T) \quad (5.62)$$

The search produces a set of optimal hub locations, evaluated using the trajectory optimisation and routing assignment objectives in Sections 5.3.1 and 5.3.2, respectively. The “Destroy” function (Algorithm 2) eliminates hub assignments using one of three processes: random hub elimination (using a predefined random distribution), cost-based elimination (of hubs with the highest setup costs) and utilisation-based elimination (of hubs that deliver the least cargo). A similar approach is used in the “Repair” function (Algorithm 3) albeit to the inverse effect (random, cost-based, and utilisation-based addition). The number of hubs to be added and removed at every iteration is uniformly randomised.

The LNS algorithm was able to solve the full set of problem instances. The results are presented in Table 5.5 and accompanied by the fitness deviations against any exact solutions, measured in cargo delivered at discrete mission times (Table 5.4). As expected, the heuristic provides a suboptimal response compared to the exact method. However, the latter can only tackle smaller missions that cannot satisfy all cargo distribution objectives. As the starting locations for UAVs are determined in accordance with overall demand requirements, heuristic solutions for such instances have larger fitness deviations due to relocation requirements (especially in the case of 10-node instances).

Algorithm 1 Large Neighbourhood Search

```

1: function LNS
2:   Input: temperature  $T$ , absolute temperature  $a$ , cooling rate  $c$ , neighbourhood size  $n$ 
3:   Output: optimal solution  $s^*$ 
4:   neighbourhood  $N \leftarrow \text{LNS}(T, a, c, n)$ 
5:   for solution  $s$  in  $N$  do
6:     fitness  $f_s \leftarrow \text{EvaluateSolution}(s)$ 
7:   end for
8:   while  $T > a$  do
9:      $s_1 \leftarrow \text{RouletteSelectionMethod}(N, f_N)$ 
10:     $s_2 \leftarrow \text{Destroy}(s_1)$ 
11:     $s_2 \leftarrow \text{Repair}(s_2)$ 
12:     $f_{s_2} = \text{EvaluateSolution}(s_2)$ 
13:     $z = \text{Random}(0, 1)$ 
14:    if  $(f_{s_2} < f_{s_1})$  or  $(z < e^{\frac{|f_{s_1} - f_{s_2}|}{T}})$  then
15:       $N_{s_1} \leftarrow s_2$ 
16:    end if
17:     $T = cT$ 
18:    for solution  $s$  in  $N$  do
19:      integer flows  $s_x \leftarrow \text{IntegerFlowsSolution}(s)$ 
20:      routes  $s_r \leftarrow \text{RouteAssignmentAlgorithm}(s)$ 
21:      fitness  $s_f \leftarrow \text{EvaluateSolution}(s)$ 
22:       $s^* \leftarrow \text{FindMinimum}(s_f)$ 
23:    end for
24:  end while
25:  return  $s^*$ 
26: end function

```

Algorithm 2 Destroy Function

```

1: function DESTROY
2:   Input: solution  $s_1$ 
3:   Output: new solution  $s_2$ 
4:    $z = \text{Random}(0, 1)$ 
5:   if  $\alpha_d > z$  then
6:      $s_2 \leftarrow \text{RandomElimination}(s_1)$ 
7:   else if  $\alpha_d \leq z < \beta_d$  then
8:      $s_2 \leftarrow \text{UtilisationElimination}(s_1)$ 
9:   else
10:     $s_2 \leftarrow \text{CostElimination}(s_1)$ 
11:   end if
12:   return  $s_2$ 
13: end function

```

Algorithm 3 Repair Function

```

function REPAIR
  Input: solution  $s_1$ 
  Output: new solution  $s_2$ 
   $z = \text{Random}(0, 1)$ 
  if  $\alpha_r > z$  then
     $s_2 \leftarrow \text{RandomAddition}(s_1)$ 
  else if  $\alpha_r \leq z < \beta_r$  then
     $s_2 \leftarrow \text{UtilisationAddition}(s_1)$ 
  else
     $s_2 \leftarrow \text{CostAddition}(s_1)$ 
  end if
  return  $s_2$ 
end function

```

Table 5.5: Heuristic runtime in sample problems. Fitness deviations compared with Table 5.4.

Instance	Demand	Drones	Payload	Mission Duration	Cargo	Deviation	Runtime
----------	--------	--------	---------	------------------	-------	-----------	---------

	(kg)		(kg)	(mins)	(kg)	(%)	(s)
4 nodes	525	15	2	50	179	0.6	83
				100	359	0.3	
				150	498	1.0	
		25	3	50	270	0	86
				100	495	-	
				150	525	0	
	900	25	2	50	300	0	83
				100	600	0	
				150	820	2.1	
		35	3	50	448	0.4	88
				100	823	-	
				150	900	0	
7 nodes	1200	35	2	50	420	0	88
				100	840	-	
				150	1140	2.6	
		25	3	50	630	0	92
				100	1140	-	
				150	1200	0	
	800	15	2	50	230	-	515
				100	460	-	
				150	610	-	
		25	3	50	348	-	521
				100	616	-	
				150	800	-	
10 nodes	1100	25	2	50	450	-	512
				100	750	-	
				150	1050	-	
		35	3	50	646	-	508
				100	1019	-	
				150	1100	-	
	1200	35	2	50	600	-	530
				100	980	-	
				150	1200	0	
		25	3	50	821	-	532
				100	1200	-	
				150	1200	-	
10 nodes	1450	15	2	50	287	4.3	184
				100	525	12.5	
				150	828	3.9	
		25	3	50	433	3.7	323
				100	841	-	
				150	1189	-	
	1450	25	2	50	449	10.2	187
				100	898	5.5	
				150	1246	5.2	
		35	3	50	694	-	355
				100	1244	-	

1650	35	2	150	1450	-	191
			50	489	1	
			100	917	-	
		3	150	1450	-	369
			50	735	-	
			100	1239	-	
			150	1618	-	

5.4 Case Study

The 1999 Chi-Chi earthquake in Taiwan had a magnitude of 7.3 and was one of the largest catastrophes to occur in East Asia over the past 20 years. Over 2,000 casualties were recorded, alongside 10,000 injuries, and damage/collapse of 14,000 buildings (Weimin 2000). As earthquake effects were more pronounced in the central Nanto county, it was used for a numerical case study by Sheu (2010) that focused on the logistics of disaster response. The demand forecasting model used includes a detailed breakdown of earthquake effects across the provinces and is used to develop a relief cargo demand dataset, which has been used by later studies as a reference. Given the variable terrain and damages to road infrastructure, this was identified as a suitable case-study for a UAV-based response strategy.

The proposed location-routing algorithm is set at the strategic and tactical planning stages (battery-optimal trajectories, strategic planning for hub selection and operational planning for optimal delivery tours). In this paper, the focus is on the distribution of relief resources to a frail population composed of elders and children, characterised by increased mortality rates (Liang et al. 2001). We assume that a care package (providing basic sustenance and medicine) weighing 1 kg must be delivered to each person.

The physical and technical specifications of the UAVs used in this analysis (Table 5.7) are informed by the commercially available QuestUAV Q-Pod and X8 cargo drones (QuestUAV 2017, UAV Systems International 2017). Although these are not capable of VTOL operation, they are similar in size, range (60 minutes when empty) and payload capacity (3kg) to hybrid models that are currently in development. Finally, it is assumed that a single delivery point is established per township (Fig. 5.5), capable of satisfying the entirety of relief cargo demands.

Table 5.6: Nanto County post-earthquake data including coordinates of each warehouse location Sheu (2007).

Node	Name	Code	Warehouse Location		Population	Frail population	Fatalities
			Latitude	Longitude			
1	Nanto City	NC	23.92176	120.6808	104,777	215	119
2	Tsao Tun	TT	23.99312	120.7258	96,833	2,373	45
3	Ji Jii	JJ	23.82843	120.7865	12,250	8,656	151
4	Pu Li	PL	23.99367	120.9647	88,271	161	114
5	Chu Shan	CS	23.70884	120.6932	62,269	342	206
6	Guo Hsin	GH	24.01024	120.8684	24,643	567	62
7	Lu Gu	LG	23.73367	120.7771	21,279	925	58
8	Ming Jian	MJ	23.8458	120.6764	42,754	1,126	123
9	Sui Li	SL	23.8092	120.8736	23,425	287	14
10	Yu Chi	YC	23.87155	120.9303	17,894	327	267
11	Chong Liao	CL	23.90838	120.781	18,252	3,892	24
12	Ren Ai	RA	24.02771	121.1435	15,358	298	4
13	Hsin Yi	HY	23.65008	121.0226	17,869	314	3

Hub establishment costs are assumed to be proportional to the local demand, with each item contributing US\$20 to the original US\$1,000 hub costs. These costs are derived from Balcik et al. (2008), where hub fixed costs were estimated to be US \$100,000 for 100,000 m³ capacity. Their fixed costs estimation are reduced, given the need for smaller hub capacity and constraints in UAV payload size (usually in the orders of 0.01 m³). A unit travel cost of US\$55/hour is considered as provided by Holguin-Veras et al. (2013).

The trajectory optimisation model is applied to obtain trajectories taking the shortest and most direct route to their destination, following terrain elevation, and overcoming geographical obstacles. An example trajectory for the journey between Chong Liao and Ren Ai is presented in Fig. 5.6. As expected, battery consumption rates vary significantly with different payloads. An empty trip for this OD pair consumes 47% of the battery charge, while a fully loaded trip consumes 68%, resulting in a 20% reduction in performance during loaded trips. A detailed analysis of battery consumption rates across payloads is included in Appendix C. Over the duration of a journey, our model provides mostly constant battery consumption rates, which are attributed to the small pitch angles used in the flight path.

Using the derived trajectories for all OD-pairs, the algorithms in Section 5 are used to determine an optimal set of UAV itineraries. Each itinerary is composed of a route r specifying

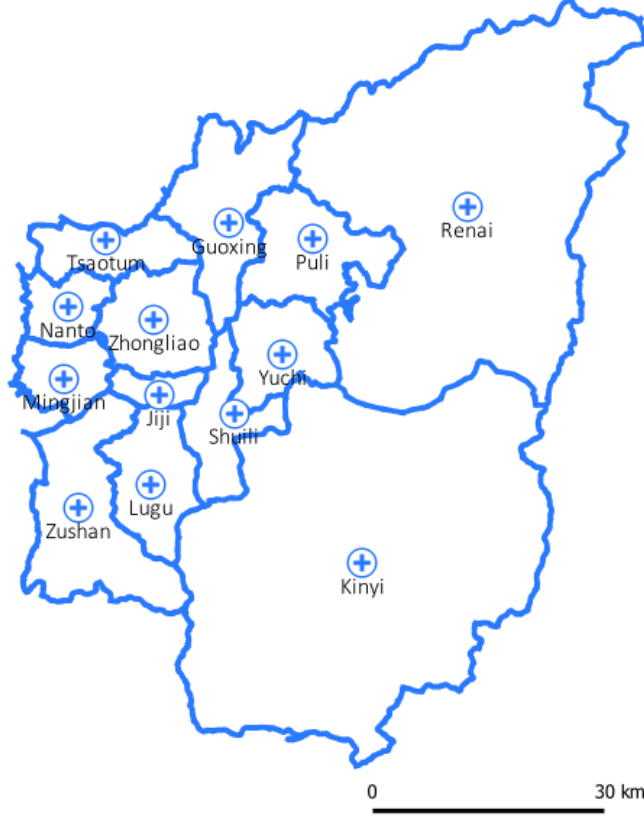


Figure 5.5: Nanto County Map, with the warehouse location indicated with a blue icon.

the order of deliveries for each drone (including any loading and battery swap operations) and an array of cargo assignments c . Nodes in r are identified using a numerical index provided in Table 5.8. Given that r contains n nodes, c will be composed of $n - 1$ members, where each element in the cargo itinerary c_i specifies the total cargo transported from r_i to r_{i+1} in the destination itinerary. Therefore, the total cargo delivered to node r_i equates to $c_{i-1} - c_i$ when $c_{i-1} - c_i \geq 0$, and to 0 kg otherwise. An example is provided in Table 8.

5.4.1 Parameter Tuning

Before applying the algorithm, manual tuning is used to derive a suitable set of LNS control parameters, while tracking fitness deviations and runtime requirements for each combination of values. As part of the tuning process, suitable values are determined for neighbourhood sizes N , the cooling rate c and the repair/destroy probabilities α_r , α_d , β_r and β_d .

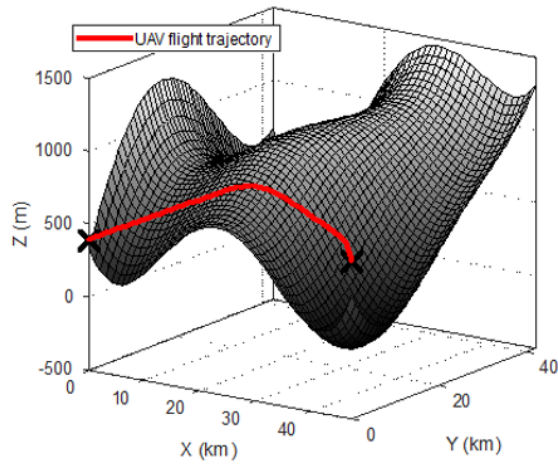
As recommended by Park & Kim (1998), the initial temperature value T should allow new solutions to be explored, so solution replacement probability P defined in equation (5.62) should

Table 5.7: Model Parameters.

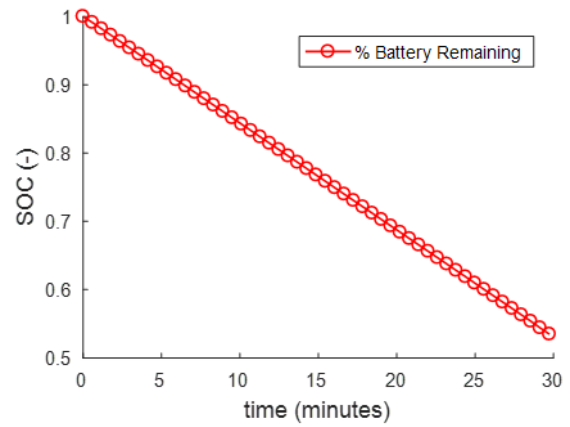
Parameter	Symbol	Value
Vehicle Parameters		
Parasite Drag Coefficient	C_{D0} [-]	0.015
Drag Polar	K [-]	0.13
Payload Capacity	Q [kg]	3
Empty Vehicular Mass	m [kg]	5
Wing Area	S [m ²]	0.75
Vehicle cruise speed	U_{cruise} [m/s]	22
Battery Parameters		
Battery Voltage	V [V]	11.1
Battery Capacity	B [Ah]	20
Cost Parameters		
Fixed Hub Cost	F [\$]	1,000
Time Cost	C [\$/hr]	55
Battery Cost	G [\$]	100

Table 5.8: UAV itinerary example.

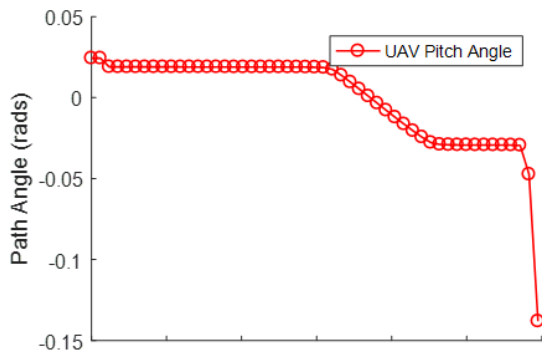
UAV	Type	Itinerary - using node indices from Table 5.6.
1	Destination Itinerary	10, 3, 11, 10, 3, 11, 10, 3, 11, 10...
	Cargo Itinerary	3, 1, 0, 3, 2, 0, 3, 1, 0...
2	Destination Itinerary	10, 12, 9, 10, 8, 10, 8, 10...
	Cargo Itinerary	3, 1, 0, 3, 0, 3, 0...



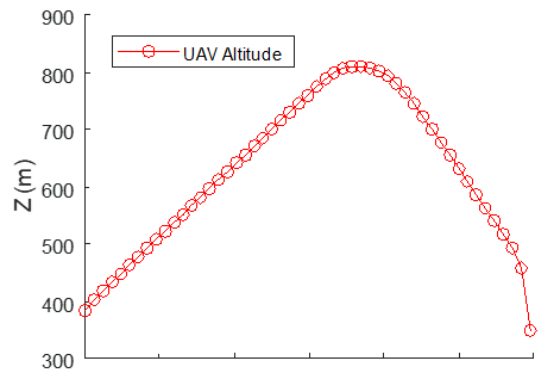
(a) 3D UAV trajectory



(b) Battery consumption



(c) Path Angle



(d) Trajectory Altitude

Figure 5.6: Trajectory connecting Chong Liao and Ren Ai with an empty UAV.

yield a unity value. Given that the initial fitness deviation ranges 100 without accounting for penalties, we specify a temperature value of $T = 10,000$. Lower absolute temperatures yield near-zero P values, ensuring an elitist selection process, which is desirable for the latter stages of a search. We, therefore, adopt value for a ranging 0.01 (Table C.5).

The algorithm provides low fitness deviations on multiple occasions. The best solution is obtained using control variables values of $a = 0.001$ and $c = 0.9999$ that provide a larger number of iterations. However, algorithm runtimes under this configuration exceed 76 hours, which is impractical for humanitarian logistics planning. As a compromise, the values $a = 0.01$, $c = 0.9995$ are used accompanied by a neighbourhood size $N = 100$, which yield 0% deviation from the best-found solution under reasonable runtimes.

With respect to operator probability, greater values of α_r and α_d increase randomness and reduce the effectiveness of the algorithm. Conversely, higher values β_r and β_d increase the frequency of utilisation addition and elimination, which yield low fitness deviations. As a result, the remaining results are gathered using $\alpha_r = 0.1$, $\alpha_d = 0.1$, $\beta_r = 0.55$ and $\beta_d = 0.55$.

5.4.2 Results

Table 9 presents the results obtained for the case study. In addition to cumulative deliveries and mission completion time, we measure the inter-arrival time that indicates the average time between deliveries, the average flight times indicating average travelling times for vehicles, and the number of batteries and average energy required per battery. Our algorithm consistently selects Yu Chi (located centrally) as the preferred location of the logistics hub, bringing remaining townships within the range of UAV operations. A comprehensive list of preliminary model outputs with optimal relief distribution rates for each township under a range of fleet sizes is provided in Appendix C (Table C.6).

For scenarios involving less than 60 UAVs, the algorithm selects an additional hub at Nanto City. Despite being a spatial outliers, Nanto City is located near Ji Jii, Tsao Tun, Ming Jian, Lu Gu, and Chong Liao, which are the townships with the highest demand rates (collectively accounting for over 80% of the total).

Table 5.9: Model outputs (without delivery balancing).

UAVs	Hubs	Mission Time (hrs)	Inter- arrival (mins)	Flight Times (mins)	Wait Times (hrs)	Batteries (per hub)	Average Recharge (kWh)	Cumulative Deliveries (kg) Day 1 Day 2
35	NC, YC	71	5.70	10.56	0	54, 80	4.50	8,820 15,024
40	NC, YC	62	4.21	10.59	0	52, 83	4.39	9,600 16,734
45	NC, YC	66	5.26	13.73	0	54, 107	4.37	8,065 15,385
50	NC, YC	62	3.79	13.75	1.00	75, 98	4.21	8,545 16,195
55	NC, YC	46	3.76	10.66	0.15	107, 104	2.83	12,057 18,942
60	YC	55	4.47	12.68	3.40	120	6.54	10,509 17,889
65	YC	51	2.35	12.68	0.46	132	5.71	11,541 19,298
70	YC	47	4.95	12.68	1.23	140	5.44	12,258 19,499
75	YC	46	2.40	12.68	0.47	154	5.01	13,332 19,499
80	YC	39	2.48	12.68	0	172	4.49	14,148 19,499
85	YC	37	5.46	12.68	0	190	4.06	14,790 19,499
90	YC	35	4.82	12.68	0	204	3.78	15,575 19,499

Algorithm Comparison

In order to assess the suitability of our LNS implementation, we compare our solution method with a Simulated Annealing (SA) algorithm, and a bi-level SA solution method proposed by Wu et al. (2002). The former is structured similar to our implementation, but uses a SA instead of a LNS, while the latter solves the location and routing problem in each level, using a feedback loop to transfer information between the two modules. Both algorithms employ the same absolute temperature a and initial temperature T . Only the cooling rate c is modified in the bi-level SA implementation to reduce algorithm runtime.

The results of the SA implementation are presented in Table 5.10. Under all fleet sizes, the algorithm selects Yu Chi as the preferred location. In addition, the battery storage requirements are lower, with a reduction in total stock of 40%, 2% and 14% for 40, 60 and 90 UAVs. However, the mission times are substantially longer in all cases, with minor improvement in inter-arrival times. Two factors contribute to this output: longer average flight time and greater wait times, which increase from 0 to up to 7 hours for 40 UAVs. The latter is a consequence of smaller battery stocks provided in the hubs, requiring larger average recharge (137%, 0%, and 22%

increases for 40, 60 and 90 UAVs). The results presented show that the LNS provides a better solution given its ability to traverse the solution space using multiple neighbourhoods.

The bi-level SA algorithm proposed by Wu et al. (2002) solves the multi-depot location-routing problem. The algorithm is adapted and run to solve the depot selection-routing problem in this paper, with results summarised in Table 5.10. Our LNS approach provides considerable improvements on mission completion times using any number of UAVs (48%, 16%, 34% for 40, 60 and 90 UAVs), in addition to increasing the computational costs to over 48 hours. Further improvements are seen in the reduction of battery stocks by 20% and 8% for 60 and 90 UAVs (with an increase in battery stocks for 40 UAVs), as well as reductions in wait times in all tests.

The bi-level SA employs stochastic exploration to traverse through the search space governed by the cooling rate c . Its main drawback relates to the large computational runtime required to explore through both levels of the SA simultaneously. The excessive computational runtime reduces its effectiveness for humanitarian logistics planning, which requires rapid decision-making under limited resources. While a reduction in value of the cooling rate c accelerates the search process, it results in a coarser exploration of the solution space, leading to larger optimality gaps. Therefore, the LNS and heuristic approach provided provides lower response mission times, as well as better resource allocation in shorter runtimes, despite being limited to a lower-bound solution.

Table 5.10: Model outputs using the Simulated Annealing (SA) and bi-level SA models.

Model	UAVs	Hubs	Mission Time (hrs)	Inter- arrival (mins)	Flight Times (mins)	Wait Times (hrs)	Batteries (per hub)	Average Recharge (kWh)	Cumulative Deliveries (kg)	
									Day 1	Day 2
SA	40	YC	114	1.61	13.36	7.57	76	10.68	6,129	11,640
	60	YC	64	3.07	12.68	3.40	118	6.54	9,341	15,736
	90	YC	53	1.32	13.36	2.20	175	4.64	15,939	18,693
B-SA	40	NC, CS, YC	92	3.56	12.77	8.60	13,10,68	9.49	5,381	10,430
	60	NC, CS, YC	63	3.44	12.82	2.37	20,24,105	5.89	7,883	15,155
	90	JJ, SL, YC	47	2.84	11.78	0.44	51,42,128	4.01	12,587	19,499

Response Fairness

At this point, it is noted that these solutions provide an uneven coverage of distribution demands. Hence, an alternative objective is proposed that seeks a proportional distribution of relief in addition to cost minimisation, ensuring that all townships are served fairly and in accordance to demand. This conforms to the humanitarian principles and WFP policy as well as to the Humanitarian UAV Code of Conduct, a document endorsed by over 60 organisations including WFP and other United Nations agencies (WFP 2002, Bagshaw 2012, UAViators 2015). Fair distribution is accounted for by including the following balancing operator to the initial objective function:

$$FP = \sigma \sum_{t \in T} \max_{i \in N} \left| \frac{del_{i,t}}{A_i} - \sum_{j \in N} \frac{\frac{del_{j,t}}{A_j}}{N} \right| \quad (5.63)$$

where $del_{i,t}$ is the total cargo delivered to node i at time step t , N indicates the total number of nodes and σ represents a cost factor that weights the impact of the component with respect to the rest of the objective function. As a balancing operator, it calculates the maximum difference in demand satisfaction at each time step.

The results for balanced deliveries with fleet sizes of 40, 60 and 90 UAVs are summarised in Table 5.11, while detailed results are provided in Appendix C (Table C.7). The revised objective succeeds in providing an even distribution of relief cargo with minimal deviation (under 2%) among nodes and reduced inter-arrival times between successive UAV deliveries. This comes at the expense of considerably longer mission durations, at rates of 44%, 7% and 23% (for 40, 60, and 90 UAVs) when compared to the results obtained in Table 5.9.

Conversely, when the original objective is used, the algorithm initially focuses response on a cluster of counties (Tsao Tun, Chong Liao and Pu Li), while Ren Ai and Hsin Yi (the two remotest counties) being last to serve. Additionally, accounting for balanced delivery results in substantial reductions in inter-arrival times. Another unexpected consequence is the 25% and 10% reduction in battery stocks for 40 and 60 UAVs. However, the wait times increase as a result, as well as the average recharge requirements, contributing to the longer mission duration.

Table 5.11: Model outputs (with delivery balancing).

UAVs	Hubs	Mission Time (hrs)	Inter-arrival (mins)	Flight Times (mins)	Wait Times (hrs)	Batteries (per hub)	Average Recharge (kWh)	Cumulative Deliveries (kg)	
								Day 1	Day 2
40	NC, YC	89	1.44	11.98	3.80	55, 46	7.96	6,129	11,640
60	YC	59	1.14	12.68	5.25	124	6.89	8,894	17,741
90	YC	43	1.17	12.68	1.82	182	4.38	13,079	19,499

Conservative Energy Estimation

A final run of the model was carried out without the payload-aware energy consumption model. This would represent a more conservative approach to battery management and assumes that UAVs always consume energy at the highest possible rate (as if fully loaded), imitating the static approach viewed in the literature, where UAV range is determined by a single parameter. Due to the resulting decrease in flight range, the model establishes additional hubs, resulting in higher capital investment and operating costs.

This method (Table 5.12) also produces significantly longer mission durations (12%, 4% and 4% respectively) compared to the initial approach. However, due to the use of more hubs, distances and flight durations among demand points are reduced, leading to lower average inter-arrival times. Furthermore, the selection of additional hubs require the provision of excess battery stock ready for replacement to reduce wait times. As a result, the battery stock increases in all scenarios (30%, 198% and 78% for 40, 60 and 90 UAVs).

These results suggest that our approach substantially improves the time and cost efficiency of UAV delivery by considering variable energy consumption, battery management and recharge location aspects that are neglected in the reviewed literature.

Table 5.12: Model outputs (with conservative battery management).

UAVs	Hubs	Mission Time (hrs)	Inter-arrival (mins)	Flight Times (mins)	Wait Times (hrs)	Batteries (per hub)	Average Recharge (kWh)	Cumulative Deliveries (kg)	
								Day 1	Day 2
40	PL, CS, SL	71	2.07	12.37	0	84, 43, 49	3.82	9,337	15,168

60	NC, PL,CS, SL, HY	57	3.45	12.26	0.80	90, 51, 82, 68, 66	1.92	12,360	18,044
90	PL, CS, SL	40	3.46	12.49	0.47	163, 110, 91	1.90	15,939	18,693

Discussion

Regarding the utility of UAV compared to conventional trucks, their main advantage relates to their independence to transport infrastructure, which was significantly damaged during the Chi-Chi earthquake (Weimin 2000). Beyond infrastructure concerns, the reduced delivery inter-arrival times are another key advantage of UAVs. Our results indicate that a fleet of UAVs would be capable of performing deliveries within 15 minutes from depot establishment. As survival rates decrease significantly over time (Huang & Lien 2012), such an approach would be readily deployable and capable of delivering life-saving resources within a time window that would not be achievable by other transport methods.

As acquisition costs and payload capacities preclude the exclusive adoption of UAVs for missions seeking to serve the entire population, our approach seeks to provide aid to frail individuals in the first instance. Using a fleet of 90 drones, the implied fleet acquisition costs would be approximately US\$ 450,000 (assuming a market-consistent unit cost of US\$ 5,000). Such a fleet would have the added benefit of being easily deployable across a broader region and allowing respondents to target areas that would be harder to reach using conventional means. It is safe to assume that as costs reduce and vehicle performance/capabilities improve, it would be possible to use UAV fleets for more ambitious mission designs.

A potential limitation to UAV operations relates to climate and weather conditions, with precipitation and wind gusts being a key concern. The performance of battery-powered UAVs can be readily affected by temperature variations – an increase in background temperature leads to increased power consumption rates, while lower temperatures reduce battery efficiency (Baronti et al. 2011).

A further concern, specific to the mission structure adopted in this study relates to the

availability of an energy source for battery charging, which could be addressed with the use of generators, renewables, or liquid fuels. We compare the findings of our study to mission outputs obtained by earlier studies that also focused on the response to the Chi-Chi earthquake, with the results summarised in Table 5.13.

Sheu (2007) proposes a relief routing approach using demand clustering and a dynamic programming model. A case study is provided for Taichung County, aiming to deliver 180,000 gallons of water, 25,000 meal boxes, 13,000 sleeping bags and 2,200 camps over three days using a two-stage distribution strategy (6 supply nodes, one distribution node per township, and a total of 24 demand points). The algorithm meets 74% of the delivery target within 3 days, an average inter-arrival rate of 4.6 hours, and an overall cost of US\$15.8 million (accounting for both transportation and inventory).

Sheu (2010) also focuses on Nanto County, estimating fatalities over time and using a clustering approach (relief urgency, damages) to coordinate response. Even though a different mission objective and methodology are used, the resulting mission also prioritises response to Chong Liao, Ji Jii and Yu Chi. This is comparable to the results from our study, where Chong Liao and Ji Jii are prioritised due to the large numbers of frail population.

Chang et al. (2014) used a GA-based approach to identify an effective response strategy using a homogeneous fleet of trucks (each with a payload of 4000 kg). Their analysis is focused on Nanto County, using a total of 30 demand points (instead of 13) and a predetermined set of 4 distribution hubs. Their model delivers enough cargo to support 159,373 people, each provided with a care-package of 3 kg. A fleet of 97 trucks completed the mission in under 24 hours, at an assumed cost of US\$ 1,000 per vehicle. The mean travel time of 30 minutes between each arrival is estimated even with higher number of distribution depots and shorter distance between delivery nodes when compared to our study.

The discussion above intends to provide an overview of findings from mission design studies that used the same event as a case study premise. It should be noted that these studies focused on land-based methods of relief distribution. In doing so, the studies do not account for debris and infrastructure damage that affected field operations in the aftermath of the earthquake – a key motivating factor for the delivery concept discussed in this study.

Despite the improvements provided in comparison with conservative battery management and alternative solution methods, there are a number of challenges that should be resolved before implementing our algorithm in real-world applications. Firstly, the algorithm does not consider the activity of other humanitarian organisations in the area. However, the algorithm is applicable in operations where control is centralised or coordinated by a singular body.

The case study presented also ignores the potential interactions with other aircraft types, such as helicopters or other UAVs. Despite Assumption 5.4 limiting potential conflicts, the existence of aircraft operations under different performance requirements necessitates the integration of performance based navigation frameworks.

Finally, while the algorithm provides nearly 20,000 kg of cargo in 35 hours, UAVs will most likely be used in tandem with ground based vehicles that are able to carry larger payloads, with UAVs servicing inaccessible areas. The proposed algorithm, however, would be useful in the design of relief distribution systems for inaccessible areas due to damaged or lack of road infrastructure, where aerial transportation is necessary.

Table 5.13: Comparison of humanitarian response case studies involving the Chi-Chi Earthquake.

Author	Objective	Methodology	Indicators	Scenario	Results
Sheu (2010)	Demand forecasting; group prioritisation	Data fusion, clustering, group prioritisation	Fatalities, relief demand, urgency	Nanto County	5-day relief-demand priority to Chong Liao, Ji Jii and Yu-Chi
Chang et al. (2014)	Relief distribution using trucks	Greedy-search, genetic algorithm	Demand satisfaction, travel time, costs, interarrivals	Nanto County	Completed in 32 hours, serving 159,373 people with 97 vehicles. *
Sheu (2007)	Demand grouping; relief distribution.	Clustering; dynamic programming.	Interarrivals, demand satisfaction, costs.	Tai Chung County.	Total cost US\$15.8 million, 4.6 hour inter-arrival time.
This paper	Relief distribution using UAVs.	Trajectories; Bi-stage LRP.	Total mission duration, demand satisfaction, inter-arrivals.	Nanto County.	Completed in 35 hours serving 19,499 people and 4.8-minute inter-arrival time.

* A mean arrival time of 30 minutes is estimated between each delivery given vehicle velocity and the average distance between destination nodes and supply nodes.

5.5 Concluding Remarks

Humanitarian response organisations are increasingly relying on UAVs for field operations, with most current deployments focusing on photography, mapping, and mission planning. Current trends indicate that UAVs have the potential to become a vital asset in the last-mile domain, especially for low volume high-value cargo. To this effect, we present a quantitative framework for the design of humanitarian response missions considering operational aspects of UAV deployment.

Our analysis demonstrates that a failure to consider payload variations and their relationship with vehicle range leads to a significant underestimation of operational capability, especially in the case of larger fleets. We also account for relief fairness as defined by WFP and other UN agencies, which leads to increased mission times but yields more evenly distributed delivery plans. The algorithm provides between 20-40% reductions in mission completion times against alternative algorithms, as well as significant improvements in computational runtime.

To the best of our knowledge, the framework presented in this Chapter is the first to simultaneously consider aspects of trajectory optimisation, warehouse selection, inventory, and battery management. Additionally, it has shown to be applicable for large vehicle fleets and long mission duration, a major limiting factor in multi-vehicle green routing problems.

Some of the most pressing operational limitations that currently affect UAV deployments relate to weather-induced performance constraints. Winds and precipitation represent critical challenges to current UAV models, as they may lead to loss of control and damages to the electronic equipment. At the same time, given the increasing investment and research in autonomous systems, communications, propulsion technologies and airframe designs, it is fair to expect that significant improvements will be made to UAV performance in the years to come.

Efforts are currently underway by the International Civil Aviation Organisation (ICAO), the European Aviation Safety Agency (EASA) and the Federal Aviation Agency (FAA) to create the necessary regulatory framework and airspace management techniques that would allow for better integration of UAVs to managed airspace activities. Furthermore, WFP has imminent plans for their deployment in last-mile operations and is currently developing coordination

mechanisms to maximise their added value and reduce any potential safety risks. The project consists of building local capacity for UAV deployments in disaster-prone countries, reducing uncertainties in availability and response times.

The approach discussed in this paper is used to develop low-altitude trajectories that would minimise interaction with other air traffic. Further research is required to develop techniques that cater for changing demand patterns, and to consider further air traffic control aspects. These topics are explored in Chapter 6.

Furthermore, while the approach considered in this Chapter involves the use of UAVs for last-mile delivery, their limited payload and range compared to ground vehicles suggest that deploying them for inaccessible and specialised cargo (such as vaccine) will yield the greatest benefits. Nevertheless, research that explore their role to support last-mile delivery by estimating demand or assessing infrastructure damage using aerial imagery. This is the focus of Chapter 7.

Chapter 6

Sequential Real-time Path Planning for Unmanned Aerial Vehicle Fleet Management

This chapter has been adapted from the following publication: Escibano J, Angeloudis P, Ochieng W (2019). Sequential Trajectory Optimisation Using Dynamic Airspace Sectorisation. Transportation Research Board 98th Annual Meeting (TRBAM 2019).

The previous chapter designed the placement of supporting infrastructure considering UAV flight trajectories. While the approach is sufficient for planning purposes, drone operations must consider the changes in demand for airspace due to the volume of UAVs expected to operate simultaneously.

Within the context of Urban Air Mobility, which manages the metropolitan air traffic operations, Thipphavong et al. (2018) highlighted that congestion management and scheduling still require further development, including on-demand route planning and system-wide optimal vehicle scheduling. The main challenge towards implementing UAM relates to scalability as a result of the number of UAVs sharing the airspace. This problem can be overcome by creating a static network with pre-defined planned flight paths.

However, this approach is only applicable when origin-destination pairs remain static, as is

the case with Rothfeld (2018). In cases where demand patterns change, such as humanitarian operations studied in this thesis where new data is gathered throughout the mission, flight trajectories should be revised without compromising safety. This calls for the development of alternative methodologies that can provide feasible solutions under reasonable runtime, even at the expense of optimality.

This gap is addressed in this chapter by proposing a bi-level airspace management heuristic, composed of a dynamic airspace segmentation algorithm which governs a trajectory planning heuristic that optimises UAV operations in terms of energy consumed and mission durations. The vehicle flight plans are discretised into sections, with each segment being improved progressively as the vehicle completes its trip. An optimal airspace structure is generated by a dynamic clustering algorithm based on a hybrid K-means PSO algorithm.

This combined approach, to the best of the author’s knowledge, has not been proposed previously in the literature. We apply the model to the UAV operation optimised in Chapter 5, which aims to maximise relief delivery in the aftermath of a natural disaster.

The chapter is structured as follows: the model framework is presented in Section 6.1, followed by the airspace segmentation model in 6.2.1. We formulate and propose a solution to the trajectory planning problem in 6.3. Section 6.4 provides the results and discussions, with final thoughts captured in Section 6.5.

6.1 Framework

The overall problem workflow consists of two main components: a vehicle routing algorithm and a trajectory management module. The former generates and evaluates problem solutions based on pre-defined key performance indicators and metrics (as described in Chapter 5). The trajectory management module uses these results as inputs to create feasible trajectories for the vehicle fleet. Together, the algorithms generate mission and supporting infrastructure requirements.

The trajectory management module contains two processes: a dynamic clustering algorithm and a trajectory planning heuristic. The dynamic clustering procedure discretises the airspace

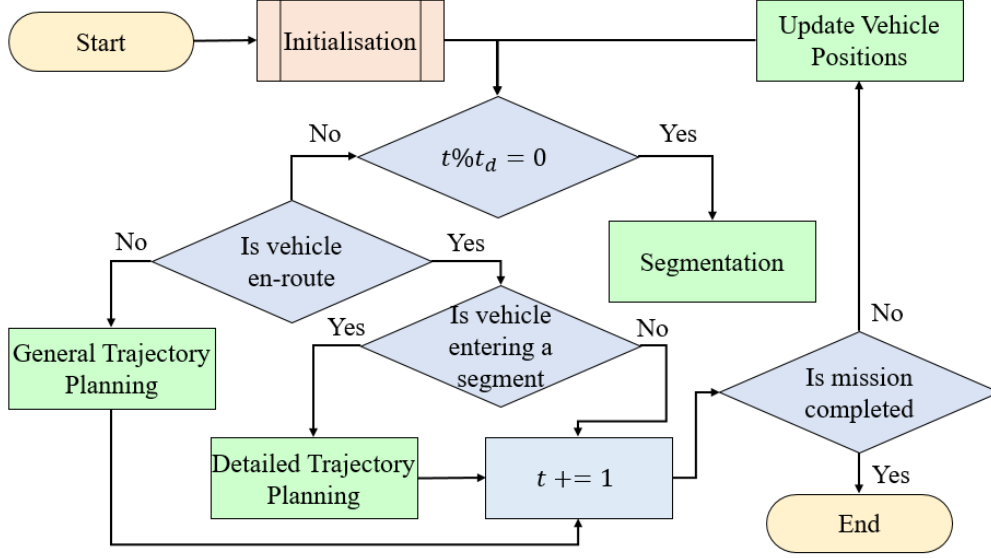


Figure 6.1: Model workflow.

into distinct segments considering traffic, weather and geography. This way, the number of conflicting vehicles and trajectories to consider during trajectory planning is reduced by limiting the analysis to obstacles within the cluster. Vehicles are given a location and approximate allowable time window to enter and/or exit a cluster that allows for modifications in their trajectories if required.

The trajectory planning heuristic creates UAV paths and modifies routing decisions based on the level of activity in an area, battery levels and demand requirements. A battery consumption minimisation is proposed whilst ensuring safety by maintaining minimum separation with other vehicles in accordance with PBN and RNP requirements.

Trajectories are assumed to follow the profile shown in Figure 6.2. Three distinct phases of flight are identified: ascent, cruise and descent, as well as a transition stage before and after the cruise phase. The trajectory planning heuristic described in this Chapter focuses on the cruise stage, as ascent and descent is assumed vertical in line with Assumption 4.1 in Chapter 5.

6.2 Airspace Segmentation

The airspace sectorisation problem is solved using a bi-step solution method. First, a clustering algorithm selects optimal centroids, and then a graph partitioning heuristic generates the sector

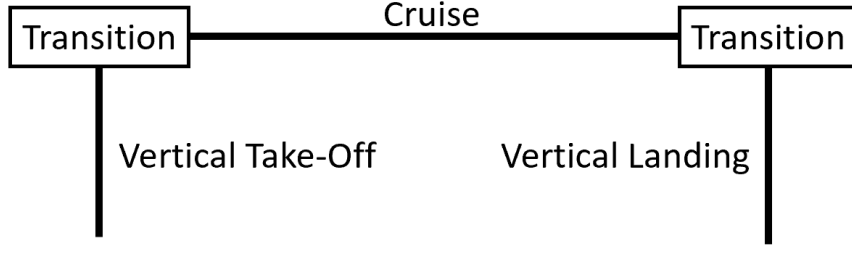


Figure 6.2: Trajectory profile.

boundaries. In the reviewed literature, GAs are used to select optimal centroid locations (Tang et al. 2012, Sergeeva 2017), and in Xue (2009), an additional process is added to reduce the heuristic runtime.

Instead, we propose a hybrid K-means Particle Swarm Optimisation (PSO) algorithm where the K-means heuristic provides an initial solution to the PSO algorithm, which further explores the solution space. The hybrid method combines the speed and local optimality of the K-means algorithm, and the ability to consider multiple non-geometric factors of the PSO. In general, PSOs have been applied to clustering problems previously with consistent better performance than locally-optimal heuristics (Esmin et al. 2015, Bopche & Jain 2016).

6.2.1 Formulation

Indices	Sets
i = cluster	S = cluster set
v = vehicle	V = vehicle set $\mathcal{D} \subset \mathcal{N}$
k = waypoint	K = waypoint set
Parameters	Variables
L, U = Coordinate space [-]	a_i = Boolean: vehicle v traverses cluster i
t_d = Decision timestep [s]	x_{ivk} = Boolean: vehicle v traverses cluster i through waypoint k
σ = Contribution parameter [-]	q_{ivk} = crossing of sector i by vehicle v at waypoint k [\$]
c_{vk} = Coordinate location of waypoint k	y_i = Cluster centroid coordinates
t_{vk} = Waypoint timestamp [s]	s_{ivk} = Waypoint k distance to centroid of cluster i
γ = Objective weight parameter [-]	r_{ivk} = travel between k and $k + 1$ waypoints

d_{vk}	= distance point of waypoint k	p_{vk}	= weight value of every waypoint k and $k + 1$ waypoints
T	= cluster flight time		

The mathematical formulation of the problem is presented:

$$\text{Minimise } Z = \gamma_1 D + \gamma_2 W + \gamma_3 P \quad (6.1)$$

$$D = \sum_{i \in S, v \in V, k \in K} p_{ivk} s_{ivk} \quad (6.2)$$

$$W = \sum_{i \in S} \left| \sum_{v \in V} a_{iv} - \frac{\sum_{i \in S, v \in V} a_{iv}}{N} \right| \quad (6.3)$$

$$P = \sum_{i \in S, v \in V, k \in K} q_{ivk} \quad (6.4)$$

Subject to:

$$\sum_{k \in K} q_{ivk} \leq 2 \quad \forall i \in S \quad \forall v \in V \quad (6.5)$$

$$\sum_{k \in K} x_{ivk} \leq a_{iv} M \quad \forall i \in S \quad \forall v \in V \quad (6.6)$$

$$\sum_{k \in K} x_{ivk} \geq a_{iv} \quad \forall i \in S \quad \forall v \in V \quad (6.7)$$

$$\sum_{k \in K} r_{ivk}(d_{v,k+1} - d_{vk}) \leq T \quad \forall i \in S \quad \forall v \in V \quad (6.8)$$

$$q_{ivk} = |x_{iv,k+1} - x_{ivk}| \quad \forall i \in S \quad \forall v \in V \quad \forall k \in K \quad (6.9)$$

$$s_{ivk} = |c_{vk} - y_i| \quad \forall i \in S \quad \forall v \in V \quad \forall k \in K \quad (6.10)$$

$$r_{ivk} \leq x_{ivk} \quad \forall i \in S \quad \forall v \in V \quad \forall k \in K \quad (6.11)$$

$$r_{ivk} \leq x_{iv,k+1} \quad \forall i \in S \quad \forall v \in V \quad \forall k \in K \quad (6.12)$$

$$r_{ivk} \geq x_{ivk} + x_{iv,k+1} - 1 \quad \forall i \in S \quad \forall v \in V \quad \forall k \in K \quad (6.13)$$

$$p_{vk} = 1 - \sigma \frac{|t_{vk} - (t + \frac{1}{2}t_d)|}{t_d} \quad \forall i \in S \quad \forall v \in V \quad \forall k \in K \quad (6.14)$$

$$x_{ivk} = \begin{cases} 1 & \text{if } s_{ivk} \leq s_{jvk} \quad \forall j \in S : j \neq i \\ 0 & \text{else} \end{cases} \quad \forall i \in S \quad \forall v \in V \quad \forall k \in K \quad (6.15)$$

$$a_{iv} = \{0, 1\} \quad \forall i \in S \quad \forall v \in V \quad (6.16)$$

$$L \leq y_i \leq U \quad \forall i \in S \quad (6.17)$$

The objective function (6.1) is composed of three components defined in (6.2-6.4). Equation (6.2) evaluates the relative distance between the centroids and waypoints, where p_{vk} represents the weight value associated to each waypoint.

The objective ensures that the centroid position is located at a mean distance of the vehicle waypoints. Equation (6.3) evaluates the total number of unique vehicles in each cluster i .

Parameter W determines the cumulative deviation against the mean, such that a reduction of this value results in a more balanced workload between the clusters. Finally, equation (6.4) calculates cluster crossings to minimise path re-planning.

The problem is constrained by Equations (6.5-6.17). Constraint (6.5) ensures shape convexity by limiting all cluster crossings twice (at entry and exit). Equations (6.6-6.7) determine whether vehicle v has traversed through cluster i . Constraint (6.8) sets the minimum cluster flight time at T . Equation (6.9) calculates the cluster crossings based on variable x_{ivk} . The distance between centroid i and waypoint k is calculated in (6.10). Equations (6.11-6.13) linearise the behaviour of (6.8), ensuring that waypoints $k + 1$ and k belong to sector i . The behaviour of the weight parameter p_{ik} is described in Equation (6.14). Constraints (6.15-6.17) bound x_{ivk} and a_{iv} as Booleans, and y_i as a coordinate within a maximum coordinate U and minimum L .

Concerning the weight parameter in Equation (6.14), the value of p_{ik} is also dependent on parameter σ , that is bounded between 0 and 1, where 0 ensures the contribution of all coordinates is equal, and 1 represents maximum weight value. An alternative method to evaluate equation (6.2) would consist on discretising the planned path into fixed distance waypoints, and evaluating the position to each coordinate. However, the approach would result in increased runtime.

6.2.2 Solution Method

Initially, the number and location of centroids are defined using a K-means clustering approach. Once the location of centroids has been optimised, we apply Fortune's sweep-line algorithm to compute the Voronoi Diagram, subdividing the volume into discrete regions (Fortune 1987). Each subdivision corresponds to a centroid, with region boundaries being equidistant to neighbouring centroids. This process ensures shape convexity and creates non-overlapping sectors. Convex airspace sectorisation reduces vehicle re-entries to a region and simplifies airspace monitoring, while the latter prevents issuing multiple instructions to the affected aircraft (Sergeeva 2017).

The main limitation of the steps above relates to their inability to evaluate complex performance metrics. The K-means clustering approach solely considers relative distance, while the resultant Voronoi graph features equidistant boundaries between selected centroids. By using a

PSO algorithm we can modify the location of the boundary vertices to alter the shape of the cells. In doing so, we identify two types of boundary vertices: fixed and free. The former is only modified along a predefined boundary line, while the latter can move in any direction. This mechanism allows us to introduce cells of fixed sizes, representing no-fly zones and extreme weather areas. A visualisation of the algorithm implementation is presented in Fig. 6.3.

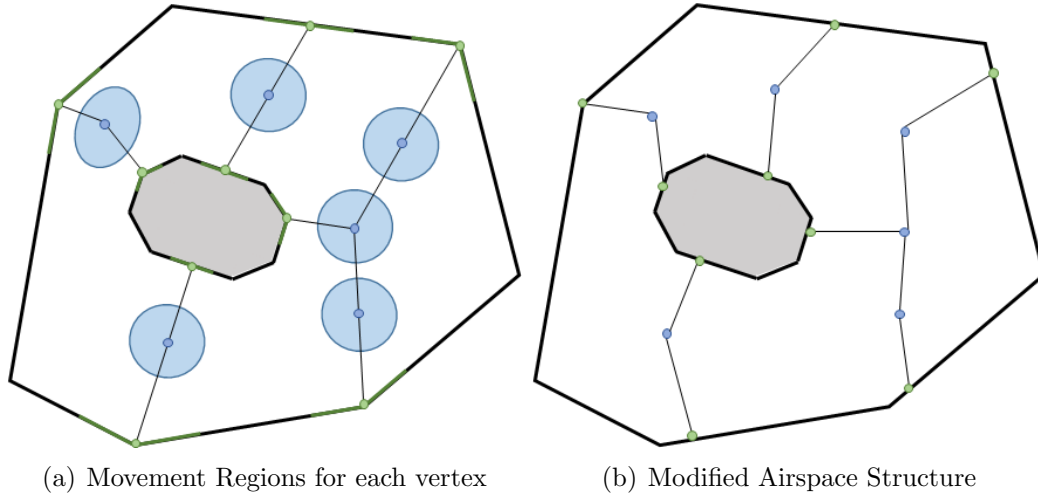


Figure 6.3: Airspace modification vertices. The green and blue nodes indicate fixed and free vertices respectively. The shaded region represents an immutable no-fly zone.

The PSO is a population-based stochastic metaheuristic where each particle, x , represents a position in the solution space, and where a population of particles is referred to as the swarm s (Poli et al. 2007). In the context of this problem, every particle contains the boundary vertices c_i . Particles are assumed to navigate the solution space independently according to a velocity vector v , unique to each particle x . Additionally, every particle x contains information of their personal best p solution, which monitors the historical position that provides the best fitness f . The fitness value is a unique scalar parameter determined by computing equation (6.1) that quantifies the optimality of the solution. Using this value, the position of the population-wide best, b , is also captured.

The algorithm commences by generating a uniformly randomised initial swarm s based on the initial K-cluster solution, and a velocity vector v . Every solution is evaluated, and the personal p and global best solutions b are determined. Given the personal best p and population best b , the velocity v is re-calculated, and the position of the swarm is updated. This cycle

is repeated until any of the two stopping criteria is met: a maximum number of iterations γ is reached, or the solution has converged. Convergence is evaluated determining whether the average velocity exceeds a minimum tolerance η . The full algorithm pseudocode is presented in Algorithm 4.

Particle movement is defined in equation 6.18, and contains three components:

1. **Inertia** is defined as the vehicle speed from the previous time step v_t and an inertia coefficient n that quantifies the contribution of inertia to the rest of the components.
2. **Personal bias** directs each particle to their personal best position p_t . This behaviour is reflected by the vector $p_t - s_t$. A uniformly random number $r_1 = U[0, 1]$ is introduced to preserve movement stochasticity. At $t = 1$, $p_t - s_t = 0$.
3. **Social bias** behaves similarly to the personal bias: particles move towards the best-known solution b_t in the neighbourhood. In this application, we assume all particles belong to the same neighbourhood, so the swarm has knowledge of the global best. Similarly, a stochastic parameter r_2 is introduced, analogous to r_1 .

The inertia n and bias parameters w_1 and w_2 quantify the relative contribution of each component to the velocity v . A high inertia parameter value compared to bias parameters allow free roaming of the particle before being pulled back close to the known best, while relative low inertia values mean that the solution orbits around the personal and global best solutions. The velocity v is also bounded by a user defined maximum allowed velocity v_{max} such that $\sum_{x \in S} v_x \leq v_{max}$. v_{max} determines the maximum search resolution allowed. An exceedingly high value would increase the likelihood of the particle to pass over good solutions, while a small value leads to insufficient exploration of the solution space.

$$v_{t+1} = nv_t + r_1w_1(p_t - s_t) + r_2w_2(b_t - s_t) \quad (6.18)$$

Compared to a Genetic Algorithm, the PSO allows the best global solution to transfer information to all the population, with varying weight (w_2). The GA can only achieve this through crossover operations, which can incorporate sections of sub-optimal solutions. Furthermore, the constant adjustment towards the best-known solution results in fast PSO convergence. According

to Hassan et al. (2005), the PSO generally provides better solutions with less variance than the GA in non-linear problems with continuous variables, including clustering problems. Additionally, the PSO has fewer parameters to adjust by the user, further facilitating convergence.

Algorithm 4 Hybrid K-Means Particle Swarm Optimisation

```

1: function H-PSO
2:   Input: Expected vehicle trajectories  $r$ , centroids  $k$ , waypoint weight parameter  $\sigma$ ,
   tolerance  $\eta$ 
3:   Output: optimal sectors  $g^*$ 
4:    $i = 0$ 
5:   average velocity  $\Delta = 1000$ 
6:   centroids  $c \leftarrow \text{K-means}(r, \eta)$ 
7:   swarm  $s \leftarrow \text{Generation}(c, [U|(x_{min}, x_{max}), (y_{min}, y_{max})])$ 
8:   velocity  $v \leftarrow U|x, y|$ 
9:   fitness  $f \leftarrow \text{Fitness}(s, r, \sigma)$ 
10:  personal best  $p \leftarrow \text{GetPersonalBest}(s, f)$ 
11:  best  $b \leftarrow \text{GetBest}(s, f)$ 
12:  while  $i < \gamma$  and  $\Delta < \eta$  do
13:     $v \leftarrow \text{NextVelocity}(s, v, p, b)$ 
14:     $s = s + v$ 
15:     $f \leftarrow \text{Fitness}(s, r, \sigma)$ 
16:     $p \leftarrow \text{GetPersonalBest}(s, f)$ 
17:     $b \leftarrow \text{GetBest}(s, f)$ 
18:     $\Delta \leftarrow \text{Average}(v)$ 
19:     $i = i + 1$ 
20:  end while
21:  return  $g^* \leftarrow \text{VoronoiGraph}(b)$ 
22: end function

```

6.3 Real-time Path Planning Component

Indices

i, j, k, l = Node
 t = Time step
 e, s = Section

Sets

$\mathcal{N}$ = Node set
 $\mathcal{T}$ = Time step set
 $\mathcal{N}$ = Trajectory Section set

State Variables

U = Airspeed [metres/second]
 SOC = State of charge [-]
 x, y, z = Cartesian coordinates [km]
 C_L = Lift coefficient [-]
 C_D = Drag coefficient [-]
 n = Load factor [-]
 n_v, n_h = Vertical/Horizontal load factor [-]

Control Variables

T = Thrust force [Newtons]
 ψ = Heading angle [radians]
 γ = Pitch angle [rads]

Force Parameters

D	=	Drag force [metres/second]
L	=	Lift [N]
W	=	Weight [N]
m	=	Drone Mass (with payload) [kg]
g	=	Acceleration due to gravity [m/s ²]
a	=	Acceleration [m/s ²]

Aerodynamic Parameters

K	=	Drag polar [-]
ρ	=	Air density [kg/m ³]
S	=	Wing area [m ²]
q	=	Dynamic pressure: $\frac{1}{2}\rho U^2$ [kg/ms ²]
C_{D0}	=	Parasitic drag coefficient [-]

Battery Parameters

P	=	Power [Watts]
I	=	Current [Amps]
V	=	Voltage [Volts]
B	=	Battery capacity [Amps-Hour]
Δe	=	Section size [s]
μ	=	Aggregate efficiency [-]

Problem Parameters

Δt	=	Trajectory length [s]
R_{xy}	=	Horiz. safety radius
R_z	=	Vert. safety radius

The model presented in this section determines energy-optimal trajectories and shares many similarities with the trajectory optimisation algorithm in Section 5.2.1. However, while that problem is tactical in nature, the algorithm developed in this section includes obstacle avoidance constraints. The formulation is defined below, adopting the assumptions presented in Section 5.2.1 as well as the following:

Assumption 6.1: Vehicle fleet is managed by a single actor.

Vehicles share information on trajectory and speed, facilitating collision avoidance. This assumption is made given the deployment of the UAV fleet by a single actor, so it is reasonable to assume that data sharing is bi-directional.

$$\text{Minimise } Z = \sum_{i,j \in \mathcal{V}, t \in \mathcal{T}} SOC_{ijte_f} \quad (6.19)$$

Subject to:

$$x_{ijt(e+1)} = x_{ijte} + \Delta e_{ijt} U_{ijte} \cos(\gamma_{ijte}) \cos(\psi_{ijte}) \quad \forall i, j \in \mathcal{N}, \quad \forall t \in \mathcal{T}, \quad \forall e \in \mathcal{E} \quad (6.20)$$

$$y_{ijt(e+1)} = y_{ijte} + \Delta e_{ijt} U_{ijte} \cos(\gamma_{ijte}) \sin(\psi_{ijte}) \quad \forall i, j \in \mathcal{N}, \quad \forall t \in \mathcal{T}, \quad \forall e \in \mathcal{E} \quad (6.21)$$

$$z_{ijt(e+1)} = z_{ijte} + \Delta e_{ijt} U_{ijte} \sin(\gamma_{ijte}) \quad \forall i, j \in \mathcal{N}, \quad \forall t \in \mathcal{T}, \quad \forall e \in \mathcal{E} \quad (6.22)$$

$$SOC_{ijt(e+1)} = SOC_{ijte} - \frac{\Delta e_{ijt} T_{ijte} U_{ijte}}{\mu BV} \quad \forall i, j \in \mathcal{N}, \quad \forall t \in \mathcal{T}, \quad \forall e \in \mathcal{E} \quad (6.23)$$

$$m U_{ijte} = T_{ijte} - D_{ijte} + W \sin(\gamma_{ijte}) \quad \forall i, j \in \mathcal{N}, \quad \forall t \in \mathcal{T}, \quad \forall e \in \mathcal{E} \quad (6.24)$$

$$D_{ijte} = \frac{1}{2} \rho U_{ijte}^2 S C_{D_{ijte}} \quad \forall i, j \in \mathcal{N}, \quad \forall t \in \mathcal{T}, \quad \forall e \in \mathcal{E} \quad (6.25)$$

$$C_{D_{ijte}} = C_{D0} + K C_{L_{ijte}}^2 \quad \forall i, j \in \mathcal{N}, \quad \forall t \in \mathcal{T}, \quad \forall e \in \mathcal{E} \quad (6.26)$$

$$C_{L_{ijte}} = \frac{n_{ijte}W}{\frac{1}{2}\rho U_{ijte}^2 S} \quad \forall i, j \in \mathcal{N}, \quad \forall t \in \mathcal{T}, \quad \forall e \in \mathcal{E} \quad (6.27)$$

$$n_{ijte}^2 = n_{v_{ijte}}^2 + n_{h_{ijte}}^2 \quad \forall i, j \in \mathcal{N}, \quad \forall t \in \mathcal{T}, \quad \forall e \in \mathcal{E} \quad (6.28)$$

$$\psi_{ijt(e+1)} = \psi_{ijte} + \Delta e_{ijt} \frac{gn_{h_{ijte}}}{U_{ijte} \cos(\gamma_{ijte})} \quad \forall i, j \in \mathcal{N}, \quad \forall t \in \mathcal{T}, \quad \forall e \in \mathcal{E} \quad (6.29)$$

$$\gamma_{ijt(e+1)} = \gamma_{ijte} + \Delta e_{ijt} \frac{g}{U_{ijte}} (n_{v_{ijte}} - \cos(\gamma_{ijte})) \quad \forall i, j \in \mathcal{N}, \quad \forall t \in \mathcal{T}, \quad \forall e \in \mathcal{E} \quad (6.30)$$

$$\Delta t_{ijte} = \sum_{e \in \mathcal{E}} \Delta e_{ijte} \quad \forall i, j \in \mathcal{N}, \quad \forall t \in \mathcal{T} \quad (6.31)$$

$$(x_{ijte} - x_{klts})^2 + (y_{ijte} - y_{klts})^2 \geq R_{xy}^2 \quad \forall i, j, k, l \in \mathcal{N}, \quad \forall t \in \mathcal{T}, \quad \forall e, s \in \mathcal{E} \quad (6.32)$$

$$(z_{ijte} - z_{klts})^2 \geq R_z^2 \quad \forall i, j, k, l \in \mathcal{N}, \quad \forall t \in \mathcal{T}, \quad \forall e, s \in \mathcal{E} \quad (6.33)$$

$$|\gamma_{ijte}| \leq \frac{\pi}{2}, |\psi_{ijte}| \leq \pi, 0 \leq T_{ijte} \leq T_{max}, U_{ijte} = U_c \quad \forall i, j \in \mathcal{N}, \quad \forall t \in \mathcal{T}, \quad \forall e \in \mathcal{E} \quad (6.34)$$

$$f(x_{ijte}, y_{ijte}) + z_{min} \leq z_{ijte} \leq f(x_{ijte}, y_{ijte}) + z_{max} \quad \forall i, j \in \mathcal{N}, \quad \forall t \in \mathcal{T}, \quad \forall e \in \mathcal{E} \quad (6.35)$$

$$|\psi_{ijt(e+1)} - \psi_{ijte}| \Delta e_{ijt} < \dot{\psi}_{max}, \quad \forall i, j \in \mathcal{N}, \quad \forall t \in \mathcal{T}, \quad \forall e \in \mathcal{E} \quad (6.36)$$

$$|\gamma_{ijt(e+1)} - \gamma_{ijte}| \Delta e_{ijt} < \dot{\gamma}_{max}$$

Equations (6.20-6.34) define UAV flight constraints under the 3D Cartesian coordinate system defined in (6.20-6.22). Battery depletion during flight is described in (6.23). Vehicle thrust is calculated using the point mass model flight equilibrium equation defined in (6.24). Equation (6.25) describes the drag force relation to the drag coefficient, the latter defined as a function of C_L in (6.26).

Equation (6.27) relates C_L and load factor n , which varies with turning as described by (6.28-6.30). Total flight time is calculated by equation (6.31). Constraints (6.32-6.33) ensure non-conflicting trajectories are developed, considering horizontal and vertical safety distances respectively.

We limit flight path angle to 90 degrees, and assume that thrust T is positive and lower than the maximum thrust level provided by motors in (6.34). Additionally, the target speed U_{cr} is predetermined and fixed throughout the cruise stage. We consider a variable terrain geography in constraint (6.35), described by function $f(x_{ij}, y_{ij})$, assuming a fixed ground clearance z_{min} and maximum altitude z_{max} . Finally, turning is limited to maximum turning rates $\dot{\psi}_{max}$ and $\dot{\gamma}_{max}$ in equation (6.36).

6.3.1 Sequential Implementation

Given the complexity of the formulation presented in the previous section and the large number of vehicles expected to operate within the same airspace, determining the optimal trajectory of UAVs is not a trivial task. Path planning runtime increases exponentially with the addition of each obstacle as a consequence of evaluating the distance to each obstacle individually, regardless of the actual risk of collision.

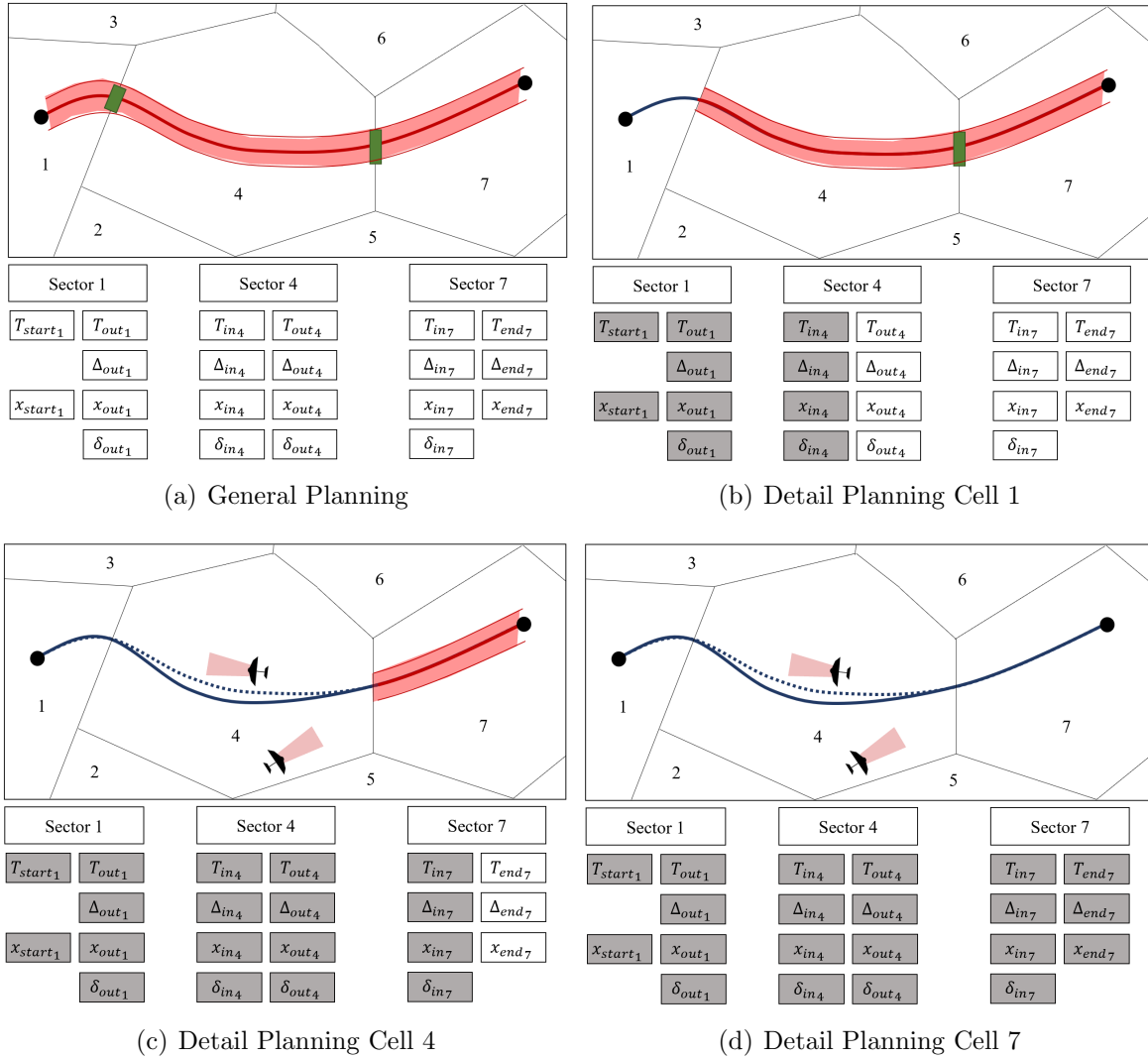


Figure 6.4: Sequential Trajectory Optimisation. Red line area indicates general planned route volume, with blue line representing the detailed trajectory considering airborne traffic. The shaded parameters indicate that their values have been determined by the planner.

We overcome this limitation by developing a sequential trajectory optimisation algorithm governed by two processes: a general path planning heuristic, and a detailed intra-sector

trajectory optimisation algorithm. Initially, a general path is developed to determine the airspace segments the vehicles will travel through, with approximate time window T and approximate entry and exit coordinates x . These are allowed to vary, and their uncertainty (Δ and δ referring to time and distance uncertainty respectively) is related to the proximity to the vehicle.

When entering each sector, the vehicle generates a detailed trajectory, accounting only for potential conflicts of vehicles inside the sector, reducing the problem size substantially. The algorithm structure is shown in Figure 6.4. The general and detailed path planning algorithms are described below:

- **General Path Planning.** The airspace is discretised into a grid based on the sector structure, and an A* algorithm is used to determine the shortest path. Path cost is composed of three components in addition to travelling distance: a segment score according to their workload during the planned path, crossing score based to minimise crossings between segment, and link occupancy based on the number of trajectories utilising the network link. This stage does not consider conflict avoidance, but instead routes the vehicle towards low congestion areas.
- **Detailed Trajectory Optimisation.** A tree-based algorithm with dynamic collision detection (which considers 4D routes of other vehicles) is used to design non-conflicting trajectories. Biased-sampling is used to discourage travel in proximity to sector boundaries. Additionally, each waypoint p is associated with a coordinate set and a heading angle, such that for any waypoint $p = x, y, \phi$. The new framework is used to connect each state using Dubins Paths assuming two flight segments: straight segments, and radius-fixed (RF) segments connecting the fly-by waypoints developed by the algorithm in accordance to PBN (ICAO 2009). The Dubins-curve developed can be of two forms: CSC (Curve-Segment-Curve) or CCC (Curve-Curve-Curve) as shown in Figure 6.5.

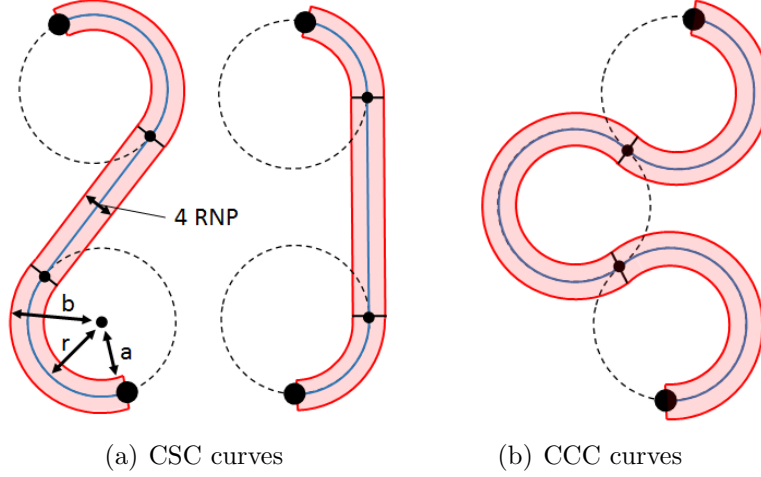


Figure 6.5: Dubins trajectory types. The blue line indicates the centre-line such that the distance to each border is 2 RNP. In RF, $b = r + 2\text{RNP}$ and $a = r - 2\text{RNP}$.

6.3.2 Heuristic Approach

The generation of potential conflict-free paths is performed using a custom-exploring tree algorithm (refer to Algorithm 5). As described in Section 6.3.1, the approach contains two levels: an A* algorithm that returns a feasible corridor that serves as an input for the lower layer, a local search Random Tree that generates a detailed trajectory of greater resolution.

Rapidly-exploring Random Trees (RRTs) were first developed by LaValle (LaValle 1998). Commonly, they commence by generating a random state x_r and the closest node x_n to x_r is found. A new node x_s is generated at distance r_{step} from x_n in the direction towards x_r . Provided the link between x_n and x_s contains no collisions, x_s is added to the node tree.

Our approach deviates in some key ways from its basic form. Our algorithm first selects the available node that is closest to the destination x_n . A set of step nodes at distance r_{step} is generated using random bearing angles that are kinematically feasible, from which the best is selected and added to the tree. In the case all step nodes are in conflict with an obstacle, the current node is added to a conflict list C , and the tree reverts to a previous state and seeks an alternate route to by-pass the obstacle.

Among the differences with the basic RRT approach, our algorithm considers kinematic constraints limiting UAV flight given constraints (6.34-6.36). In particular, vehicles are limited on their yaw rate by $\dot{\psi}_{max}$, and pitch rate $\dot{\gamma}_{max}$.

Next, the explorable area is delimited by the corridor generated by the General Path Planner. Thirdly, while the closest node x_n is found using Euclidean distance, we connect the candidate node x_s to the existing tree using battery consumption evaluating factor based on equations (6.23-6.27) as shown in Select Best procedure.

Finally, the algorithm considers multiple options for path extension at every step, providing each with a score based on their proximity to obstacles and destination expressed by (6.37). The unselected paths are kept in memory in case the algorithm has to revert to a previous state. This may occur if conflict cannot be avoided from its latest position.

$$F_n = -(x_d - x_n)^2 + (y_d - y_n)^2 + \sum_{o \in \mathcal{O}} ((x_o - x_n)^2 (y_o - y_n)^2) \quad \forall n \in \mathcal{N} \quad (6.37)$$

$$P_n = \frac{F_n}{\sum_{n \in \mathcal{N}} F_n} \quad (6.38)$$

Algorithm 5 Random Tree

```

1: function MAINRT
2:   Input: origin node  $x_o$ , destination node  $x_d$ ,
   vehicle  $v$ , maximum iterations  $\theta$ , step radius  $r_{step}$ ,
   neighbour radius  $r_n$ , obstacles  $O$ 
3:   Output: path  $p$ 
4:    $U = \text{List}(x_o)$ 
5:    $C = \text{List}()$ 
6:   for iteration  $i$  in  $\theta$  do
7:      $x_t \leftarrow \text{SelectCurrentStep}(U, x_d)$ 
8:      $x_s \leftarrow \text{CreateStepNodes}(x_t, r_{step})$ 
9:     if  $\text{IsEmpty}(x_s)$  then
10:      continue
11:    end if
12:     $x_b \leftarrow \text{SelectBest}(x_s, x_d, O)$ 
13:    if  $\text{InConflict}(x_b)$  then
14:       $x_t, C, i \leftarrow \text{Revert}(C, U, x_t, i)$ 
15:    end if
16:     $U.\text{Add}(x_c)$ 
17:    if  $\text{IsGoalReached}(x_c, x_d)$  then
18:      break
19:    end if
20:  end for
21:  return  $p \leftarrow \text{ExtractPath}(U)$ 
22: end function

23: function CREATESTEPNODE
24:   Input: current node  $x_r$ , random  $r$ , vehicle  $v$ ,
   step radius  $r_{step}$ 
25:   Output: step node  $x_s$ 
26:   pitch  $\gamma \leftarrow \text{GetPitch}(r, v)$ 
27:   bearing  $\psi \leftarrow \text{GetBearing}(r, v)$ 
28:   return  $x_s \leftarrow \text{GetNode}(x_r, r_{step}, \psi, \gamma)$ 
29: end function

30: function SELECTBEST
31:   Input: current node  $x_r$ , step node list  $X_s$ ,
   vehicle  $v$ , obstacles  $O$ 
32:   Output: step node  $x_s$ 
33:   for  $x_n$  in  $X_s$  do
34:     cost  $c \leftarrow \text{CalculateBatteryUse}(x_n, x_r, v)$ 
35:     distance  $d \leftarrow \text{CalculateDistance}(x_n, x_r)$ 
36:     if  $d > r_n$  then
37:       continue
38:     end if
39:     if  $\text{KinematicallyInfeasible}(x_n, x_r, v)$  then
40:       continue
41:     end if
42:     if  $\text{CheckObstacles}(x_r, O)$  then
43:       penalty  $P \leftarrow \text{CalculatePenalty}(x_r, O)$ 
44:     end if
45:     if  $c_{x_r} \leq c + p + d$  then
46:       continue
47:     end if
48:     step node  $x_s = x_n$ 
49:   end for
50:   return  $x_s$ 
51: end function

```

6.4 Results and Discussion

The algorithms developed in this section are applied to a selection of tests to ensure their usability in large problem instances. Three tests are carried out:

1. Trajectory planning tests under static obstacle and dynamic obstacles.
2. Airspace clustering model
3. Simulation of a response operation to monitor the algorithm runtime of the proposed algorithm under large drone networks.

6.4.1 Path Planning Results

We test our path improvement algorithm in a $50 \times 50 \text{km}^2$ area under four conditions: in the presence of a static obstacle, multiple static obstacles a dynamic obstacle, and static and dynamic obstacles.

Static Obstacle Avoidance

The first static obstacle scenario considers the movement of a drone between (6,16) and (39,39), with an obstacle of radius 5km positioned at (22, 30). Figure 6.6a, c and e show the 3D trajectory, top-down view of the trajectory and distance to the obstacle.

Figure 6.6c shows that the path lies within the corridor provided and avoids the obstacle region successfully, as the safety distance to the obstacle is always maintained.

The algorithm is then tested using multiple obstacles. The starting and ending coordinates are kept the same, but instead three obstacles at (27,21), (11,24) and (29, 37) are created, with safety radius of 10, 5 and 5 km respectively. The solutions are depicted in Figure 6.6b, d and f, which show the 3D trajectory, top-down view of the trajectory and distance to the obstacle, respectively.

Despite the increase in airspace complexity, the algorithm creates a feasible trajectory between the obstacles, always maintaining the required safety distance. Note that Figure 6.6f shows the safety distance requirement of the closest obstacle, hence the sudden increment of 5 to 10 km at around 10 min time stamp.

The path planner has successfully generated conflict-free trajectories under static obstacles. Hence, we carry out experiments that include dynamic obstacles next.

Dynamic Obstacle Avoidance

For these tests, it is assumed that the obstacle moves from (21, 30) to (20, 48) and the main vehicle must traverse from (3, 40) to (38, 40). Both consider a safety distance of 1 km between the vehicle and the moving obstacle to facilitate visualisation. A top view is provided at different time steps in Appendix D.

In the first scenario, the UAVs successfully avoids incoming targets while minimising deviation

from the optimal path and stay within the provided corridor. The minimum distance observed between the vehicles is 2.5 km, which is sufficient in this example. It is assumed as well that information regarding the other obstacles trajectory is known. Results are presented Figure 6.7a, c and e, showing the 3D trajectory, top-down view of the trajectory and proximity to the obstacle.

In the second scenario, two additional static obstacles are located at (16,41) and (27, 34) with safety radius of 5 and 4 km respectively. In this case the vehicle still successfully finds de-conflicted trajectories for the vehicles that share the same volume. In this particular instance, the minimum safety distance is reached 15 min after vehicle 1 takes off.

The results show that the trajectory planner produces feasible trajectories for UAVs in the presence of static and dynamic obstacles. We then proceed to evaluating the airspace segmentation algorithm in the next section.

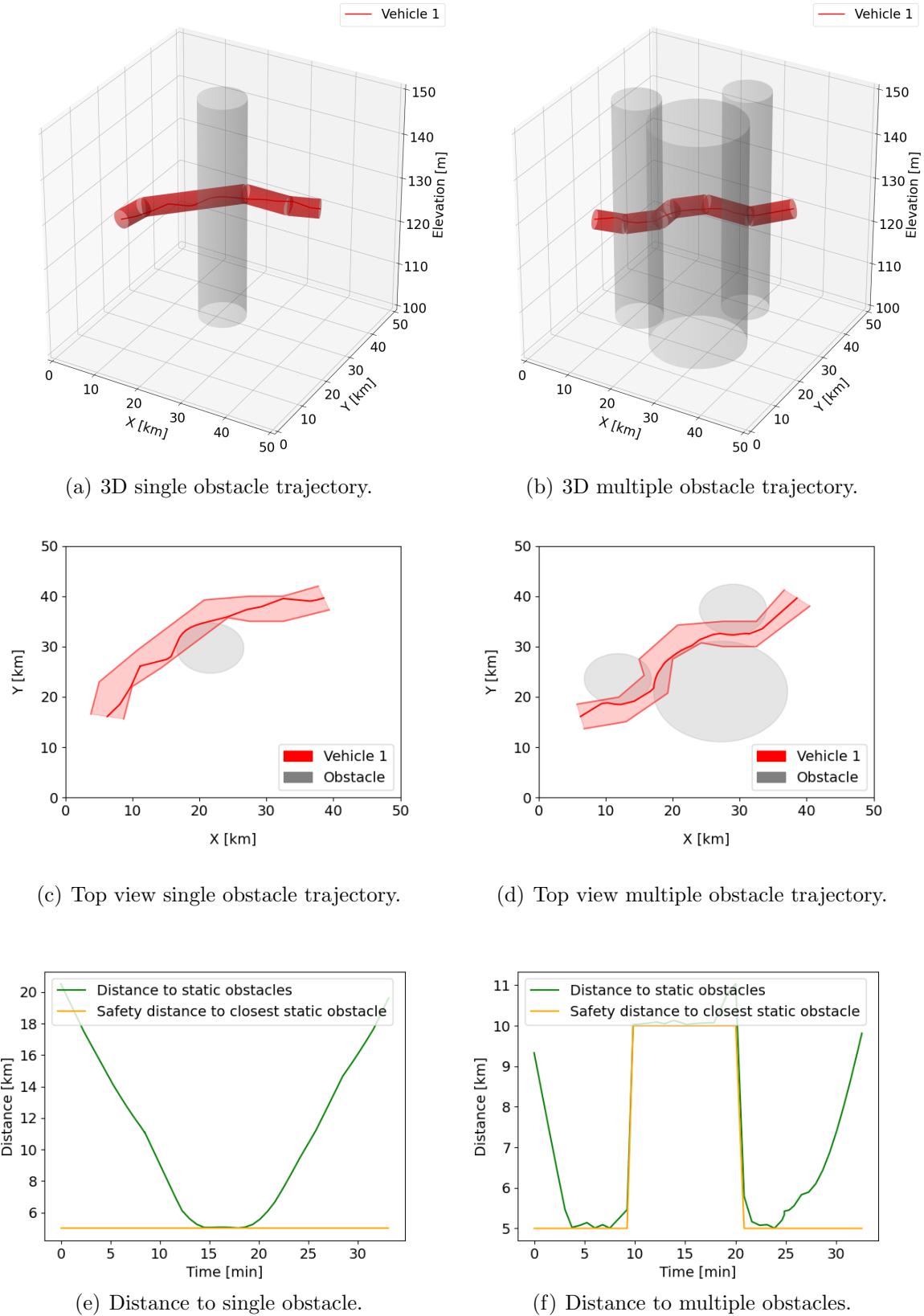


Figure 6.6: Static obstacle results.

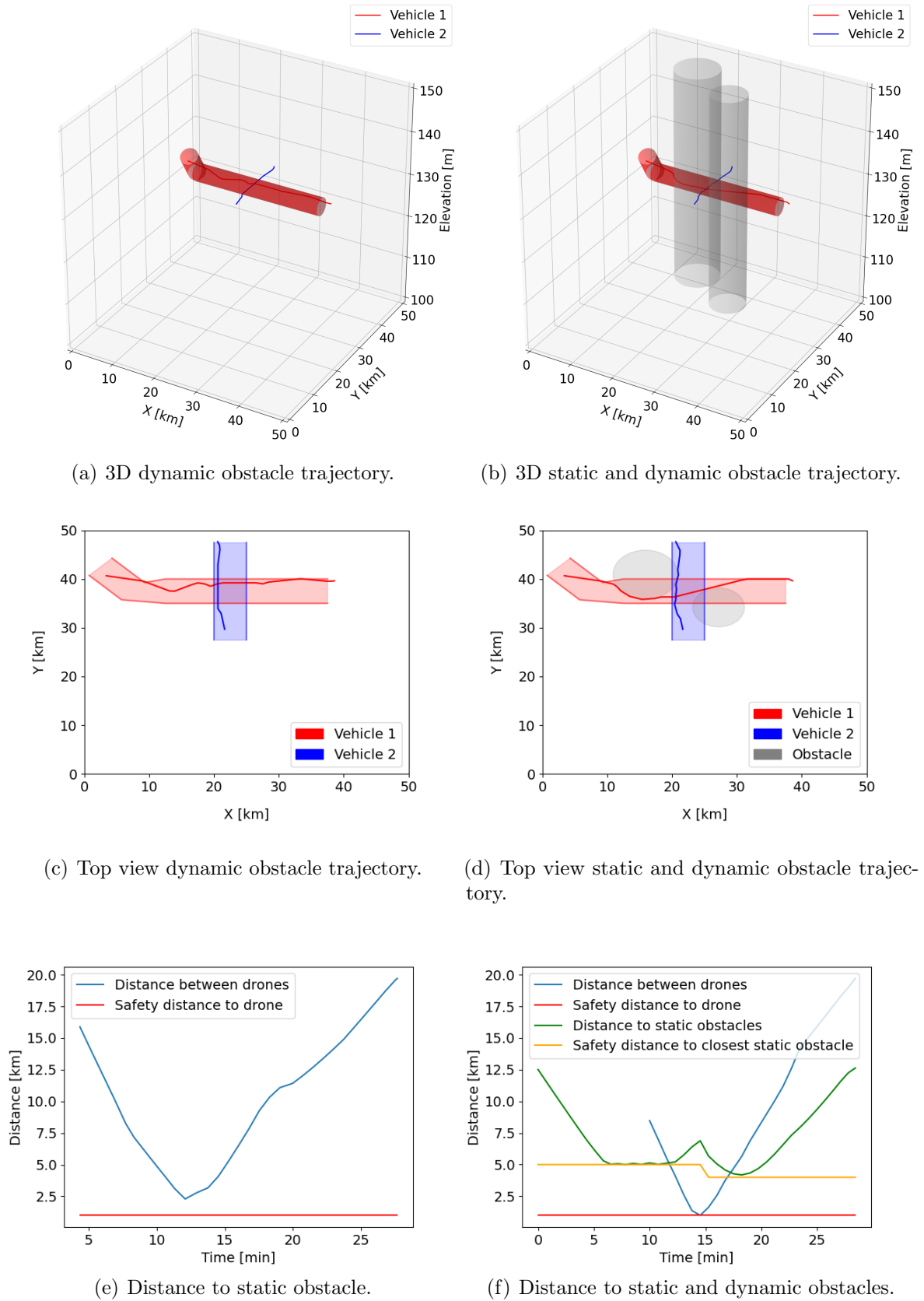


Figure 6.7: Dynamic obstacle results.

6.4.2 Airspace Segmentation Results

The algorithm developed in this section is applied to the Nanto County network, adapted from the 1999 Chi-Chi earthquake. For convenience, the network is shown in Figure 6.8. We generate a set of trajectories based on a hypothetical aerial relief response to the Chi-Chi earthquake (Escribano Macias et al. 2020). The final network, including node demand, is presented in Table 5.6.

We employ the RNP categorisation developed by Ren, Castillo-Effen, Yu, Johnson, Yoon, Takuma & Ippolito (2017) and assume 30 RNP conditions during cruise, and use Sun et al. (2018) to determine RNP requirements during VTOL. Therefore, the response takes place under 120m minimum lateral separation requirement between each vehicle in cruise.

Before applying the sequential trajectory optimisation algorithm to the response operation, we derive suitable PSO parameters using manual tuning. Informed by previous studies Trelea (2003), Paterlini & Krink (2006), we test the following parameter ranges: inertia $n = [0.4, 1]$, population size $N = [100]$, maximum velocity $V_{max} = [5] \times N$ and bias values $w_1 = [0.4, 1.5]$ and $w_2 = [0.4, 1.5]$. The results are summarised in Appendix D.

Each value combination is run twenty times and average fitness values are gathered, assuming the following objective function coefficients: $\gamma_1 = 1$, $\gamma_2 = 1$ and $\gamma_3 = 1$. Our analysis shows that optimal values for PSO are $n = 0.4$, $p = 0.6$, $s = 1$.

Table 6.3: Segmentation Algorithm Results.

Model	Crossings	Aircraft Count		Flight Time	Fitness
	(-)	Avg (-)	Dev (-)	Deviation (min)	
Proposed Approach	5,545	1,546	1,774	4.969	20.7
Genetic Algorithm	7,988	1,813	1,795	4.040	23.7
K-means	9,844	2,026	1,608	2.677	25.3

We compare the results from the PSO with a Genetic Algorithm, and a K-means approach that solely seeks to optimise the position of the voronoi centroids. The visual discretisation is shown in Figure 6.9, with detailed results shown in Table 6.3. The results obtained are the best solution of the twenty repetitions carried out in the experiment. The proposed methodology provides significant improvements in workload reduction and balance.

Firstly, the heuristic lowers sector crossings, reducing workload as the sequential trajectory

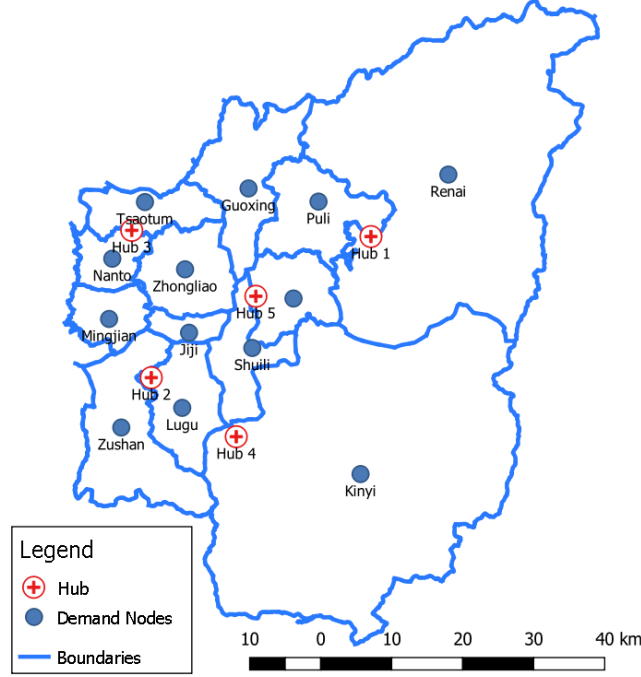


Figure 6.8: Nanto County Map, with the relief locations indicated with a blue icon, and depots shown in red.

optimisation executes when vehicles enter a new sector. Additionally, both average aircraft count and average variance are lowered, resulting in improved workload balance between the sectors. Given the relationship between workload and sector aircraft count is exponential, reductions in maximum vehicle counts lead to substantial runtime accelerations. Finally, despite the reductions in aircraft count and crossings, the flight time variance increases in the PSO compared to the GA and K-means, suggesting that vehicles spend longer time in clusters with less traffic.

6.4.3 Sequential Trajectory Results

We now evaluate the performance of the sequential trajectory optimisation algorithm given a routing schedule created using the algorithm developed in Escribano Macias et al. (2020). We run the dynamic airspace model under three re-segmentation decision times $t_d = 480, 980, 1440$ minutes and three vehicle fleet sizes $V = 25, 30, 35$, with results presented in Table 6.4.

It is expected that increasing re-segmentation frequency results in longer algorithm runtime. However, for different vehicle fleets, this relationship varies. This is a result of demand variation

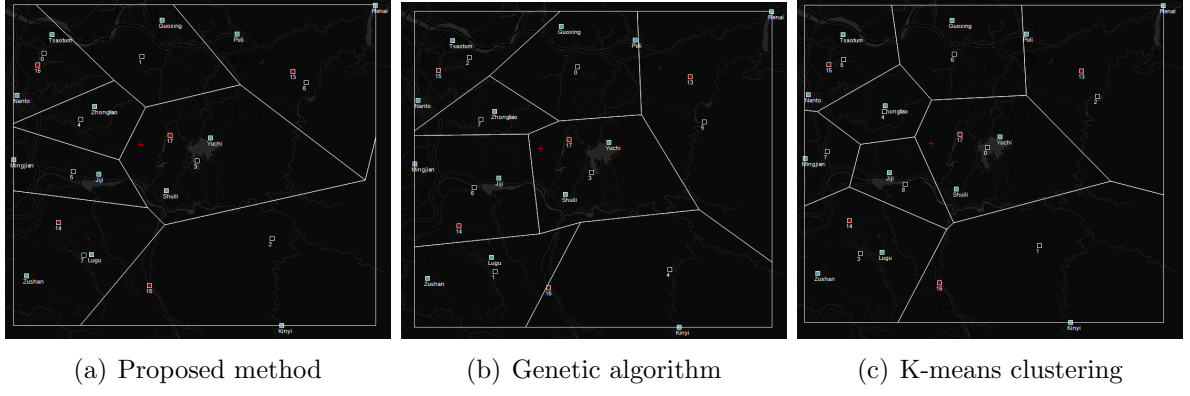


Figure 6.9: Segmentation solution for static demand case.

during the mission, resulting in increased crossing requirements and higher sector aircraft count number, which are both the greatest factors contributing to algorithm runtime.

Algorithm runtime decreases with higher re-segmentation times, as it allows less frequent and shorter trajectory optimisations. However, for 960 and 1440 re-segmentation times, the runtime slightly increases or plateaus, suggesting that the benefits of re-structuring do not overcome the overhead costs of running the airspace segmentation algorithm. In terms of mission parameters (mission completion, inter-arrival time and flight times) varying the re-segmentation time has little effect.

We compare the results of our dynamic airspace segmentation (DAS) model with a single-segment (SS) solution, which assumes no airspace segmentation, similar to the conditions assumed in Escribano Macias et al. (2020). The results are summarised in Table 6.4. Under all fleet sizes, inter-arrival and drone flight times marginally increase when no segmentation is carried out, leading to mission completion time increments of 3-4 hours (2-3% compared to DAS).

The largest improvements are seen in algorithm runtime, where the proposed method accelerates the algorithm by approximately 10 hours or 30%. The reduction in relative acceleration is a result of the larger problem size, leading to slower segmentation and smaller airspace demand variation. It was expected that the sequential improvement algorithm would result in marginally sub-optimal results given its discretised nature. However, the results have proven clear benefits with minimal drawbacks.

Table 6.4: Humanitarian response with segmentation. A total of 19,800 kg were delivered.

UAVs	Segmentation Freq. (min)	Completion (hr)	Inter-arrival (avg-min)	Flight Time (avg-min)	Number Flights -	Runtime (hr)
25	480	113.0	14.3	9.8	16169	4.5
	960	113.8	14.4	9.8	16251	2.9
	1440	114.1	14.4	9.9	16250	3.5
	SS	117.4	14.9	10.1	16253	15.6
30	480	98.2	12.1	9.9	16343	4.6
	960	97.4	12.1	9.9	16341	3.1
	1440	97.5	12.1	9.9	16341	3.3
	SS	100.3	12.7	10.2	16342	14.7
35	480	84.8	10.1	9.8	16157	5.1
	960	85.1	10.2	9.8	16157	3.5
	1440	84.6	10.1	9.7	16157	3.5
	SS	86.0	10.8	10.1	16157	15.2

6.4.4 Discussion

Our analysis has shown that our sequential trajectory planning procedure provides important improvements in terms of algorithm workload with minimal effects on solution optimality. Other advantages of the approach is the ability to isolate problematic areas due to communication errors, weather or prohibited access.

Some barriers still limit implementation. While the proposed algorithm considers UAV safety volume boundaries based on RNP, it is assumed that within the airspace segment the flight information of other obstacles is known. This deviates from reality in three key ways.

Firstly, it assumes that vehicles will communicate their entire planned trajectory or intent, which may not be desired by users. As drone regulations and supporting infrastructure is still under development, it may become standard to share this information widely. Platforms already exist in the form of Unifly or AirMap that aim to manage unmanned air traffic that have access to this information and can share it widely (Unifly 2019).

Secondly, the algorithm does not consider deviations once the final flight plan is established. To reduce the impact of this assumption, a corridor volume is provided to UAVs that allow them to modify their route in the presence of unexpected obstacles. UAVs, therefore, require sense-and-avoid capabilities for BVLOS to detect these obstacles. The creation of the sense-and-avoid mechanism is outside the scope of this study.

Finally, the sequential trajectory algorithm proposed follows RNP and PBN regulations as discussed in Chapter 3. However, further testing is required to determine the appropriate RNP for humanitarian missions. This requires measuring the trajectory performance in terms of accuracy, integrity, availability and continuity, and incorporating the errors that emerge during UAV flight. Further modifications may be required in the case of upcoming UAV-specific law, which may include the provision of contingency volumes and alternative landing areas in the case of partial failure.

Improvements can be made by carrying out flight tests outside the simulation environment, which would require to integrate the software with communications and vision hardware. In terms of application, more complex environments that consider variable demand over time, parallel humanitarian operations (search and rescue, for instance) can be implemented.

6.5 Concluding Remarks

Organisations are increasingly investing in developing UAV for on-demand commercial transportation and humanitarian response. However, current modelling approaches consider the existence of a static pre-defined network. This method is not applicable when new obstacles emerge, or in the case where the network structure changes during deployment, which may be the case in operations where initial data is uncertain, such as humanitarian missions.

To this effect, we present a novel dynamic trajectory planning algorithm that uses a Hybrid K-means Particle Swarm Optimisation metaheuristic to re-structure airspace segments and a custom sequential trajectory planning algorithm that determines battery-efficient collision-free paths. Our analysis suggests that the method significantly improves algorithm run-time with minimal loss of optimality compared to existing heuristics.

While there is a record of permitted BVLOS deployments, their approval rate is low and require long waiting time. As an example, the FAA currently approves 10% of their applications, with a mean response time of 30 days (Rochon et al. 2018). However, FAA and EASA aim to accelerate BVLOS operations approval without jeopardising airspace safety. Besides, WFP is developing coordination mechanisms for the fast, safe and regulated deployment of UAVs in

humanitarian contexts.

The approach discussed in this chapter plans aircraft trajectories through optimal airspace management. Current limitations of the algorithm include the assumption of no unfavourable weather conditions, which could be isolated in separate sectors using the presented sectorisation framework. In addition, the model does not consider the presence of other aircraft types.

Chapter 7

Endogenous stochastic optimisation for relief distribution assisted with Unmanned Aerial Vehicles

This chapter has been adapted from the following publication: Escibano J, Goldbeck N, Hsu PY, Angeloudis P, Ochieng W (2020). Endogenous stochastic optimisation for relief distribution assisted with Unmanned Aerial Vehicles. *OR Spectrum* 42, 1089–1125.

The previous chapters have outlined the planning and operation of UAV for disaster relief delivery. The limitations of UAV in terms of payload capacity make them ideal to transport small and high value material, like medicine or high caloric value food to areas with limited accessibility. However, the bulk of the relief is still delivered using road-based transportation.

For this reason, damage analysis and assessment have been the focus of humanitarian organisations and researchers alike (Bravo & Leiras 2015). The small weight, easy transportation and high data quality of UAVs provide a unique opportunity for respondents to collect and assess infrastructure damage during disaster response operations. A recent example is the 2016 Ecuador earthquake, where UAVs were deployed days after the disaster to assess road damage and provided essential data to the Ministry of Transport and National Secretariat for Risk Management, informing their relief routing decisions (DuPlessis 2016).

Despite successful pilot applications and technological improvements, some barriers remain for the widespread use of UAVs for post-disaster damage assessment. For instance, image processing and analysis are still notably time-consuming, as exemplified during the 2016 Ecuador deployment, where processing the survey results required 12 to 24 hours of continuous effort (FSD 2016).

For an efficient use of the limited human and technological resources available in the immediate aftermath of a disaster, UAV deployments must be prioritised when and where they provide the most benefits for the affected community, including the assessment of damages to key infrastructure and the estimation of the population needs that ensure the provision of precise and rapid response. Moreover, flight times should be minimised given energy shortages and safety risks (risk of injury due to malfunction, vandalism, for example) in humanitarian settings (Van Wassenhove 2006, Van Wassenhove & Pedraza Martinez 2012, Meier et al. 2016). Overall, this necessitates the development of bespoke modelling approaches that determine the optimal time and location for road damage data collection, considering cumulative knowledge and informing future relief distribution decisions.

The literature review presented in Section 7.1 reveals that the problem of UAV-based network assessment and relief distribution is yet to be formulated and solved simultaneously. Moreover, existing optimisation methods assume deterministic or exogenous stochastic conditions.

To address this gap, this chapter presents a mathematical optimisation problem for the provision of humanitarian relief supported by UAV-based damage assessment. The problem formulation presented assumes that the extent of damage to the road network is uncertain, with UAVs deployed to survey the network and reduce the uncertainty related to vehicular travel times and travel costs. Simultaneously, ground vehicles are routed through the network, delivering relief cargo.

Therefore, the problem is defined as endogenously stochastic, where the uncertainty in the network depends on the routing decisions of UAVs and ground vehicles. A greedy sequential solution algorithm and a metaheuristic (a Genetic Algorithm) are presented to determine optimal routing itineraries at discrete time steps. Thus, the contribution of this chapter is threefold:

1. To the best of the authors' knowledge, it is the first to formalise an optimisation problem

to model the UAV-assisted endogenous stochastic travel time relief distribution. Therefore, it proposes a metric to quantify the benefit of integrating UAV road assessment with humanitarian response.

2. A greedy exact solution procedure is proposed. However, additional simplifications are required to ensure reasonable runtime due to its high complexity. A relaxed version of the problem is solved using a meta-heuristic, which is generalised and can solve multiple problem instances.
3. It demonstrates the solution methods' applicability by solving a realistic case study adapted from the 2010 Haiti earthquake. The results show that important benefits are obtained when introducing UAVs in humanitarian response operations.

We analyse the current literature in Section 7.1, followed by the mathematical model defined in Section 7.2. The solutions methods described in Section 7.3 are applied to a numerical case study focusing on a hypothetical response to an earthquake based in Haiti in Section 7.4. Conclusions and recommendations for future work are provided in Section 7.5.

7.1 Current Methods for Road Damage Assessment

The following section presents an overview of the current research on endogenous stochastic optimisation methods and humanitarian relief optimisation models that consider road network damages or accessibility. Despite the two aspects being related, no study thus far incorporates both of them simultaneously. As such, the review analyses the state-of-the-art separately.

7.1.1 Post-disaster relief optimisation

This review focuses on studies that optimise the routing decisions of relief distribution or network repair vehicles. Generally, these decisions are captured by the Vehicle Routing Problem (VRP), which models vehicle capacity constraints, stochastic demand and travel time. Duque & Sörensen (2011) developed a greedy randomised adaptive search procedure for allocating engineering resources to repair road networks. A similar problem instance is solved by Aksu &

Ozdamar (2014), who presented an Integer Programming (IP) model relaxing the VRP to a path-based algorithm by ignoring travel times.

The "debris removal problem" is further studied by Sahin et al. (2016). A recursive Mixed Integer Linear Programming model (MILP) is developed to maximise the cumulative network accessibility whilst minimising clearing time. Similarly, Akbari & Salman (2017) conceived an exact MILP model to generate a synchronised road clearing schedule for multiple engineering teams. Kim et al. (2018) adopted a MILP model to minimise the weighted sum of total damages and the completion time while considering time-dependent damage characteristics.

Balcik et al. (2008) proposed a two-step approach to solve the relief distribution problem. Their method generates vehicle routes using the Travelling Salesman Problem (TSP) formulation in the upper level and vehicle itineraries in the second step. Özdamar (2011) incorporates evacuation and refuelling considerations to the relief distribution problem. The proposed algorithm relaxes the VRP problem to a static integer flow formulation, significantly improving its runtime. Vahdani et al. (2018b) conceived a two-phase, multi-objective, multi-period and multi-commodity mixed integer programming model which locates the distribution centres and plans vehicle routes.

The combined problem of relief distribution and network repair has been studied by Yan & Shih (2009). The proposed multi-objective MILP model is capable of planning emergency road repair operations and the subsequent relief distribution deployments for multiple commodities simultaneously. Ransikarbum & Mason (2016) developed a Goal Programming-based (GP) multi-objective integrated response and recovery model that optimises network recovery and relief distribution simultaneously. Finally, Li & Teo (2018) conceived a multi-period bi-level programming model to optimise road repair and relief optimisation. They employ a genetic algorithm to maximise relief delivery quantity and network accessibility.

The algorithms reviewed share limitations relating the disregard of partial damages and assumption that repair times and the extent of the damages are all known. To consider the latter, stochastic optimisation algorithms must be developed.

In this case, the source of uncertainty plays an important role in the formulation and solution method. Exogenous stochastic algorithms focus on problems where uncertainty emerges from

sources unconnected to the choices of the decision-maker, while endogenous stochastic problems consider uncertainty changes based on the choices of the user. The review will explore the developments in both types of problem in the next sections.

7.1.2 Exogenous stochastic optimisation

During the time period immediately after the disaster, the state road network is uncertain and should be assessed before planning debris removal or supplies distribution. Many studies consider these uncertainties by adopting the two-stage stochastic programming (2SP) modelling technique for depicting the post-disaster humanitarian logistics response under an uncertain demand of relief supplies.

Most studies reviewed that consider exogenous stochastic conditions fall under the Hub Location-Allocation Problem (HLP). In our review of HLPs in Section 2.5, we highlight Rawls & Turnquist (2012), Bozorgi-Amiri et al. (2013), Bozorgi-Amiri & Khorsi (2016) as relevant to this section. Van Hentenryck et al. (2010) focused on large scale disaster relief and recovery optimisation modelled as a single commodity allocation problem. A similar framework is presented considering multiple commodity types by Tofighi et al. (2016).

Several studies solve the on warehouse location-demand allocation problem, with particular focus on hurricanes (Rawls & Turnquist 2012, Pacheco & Batta 2016), earthquakes (Mete & Zabinsky 2010), and floods (Garrido et al. 2015). Noyan (2012) studied risk-adverse location-allocation optimisation instead of common risk-neutral problems, while Bozorgi-Amiri et al. (2013) presented a multi-objective robust stochastic programming problem considering uncertain demand, supply, cost and transportation.

The common goal of the established stochastic models is to minimise the total expected cost among all possible scenarios. The operation cost generally consists of ordering and shipping costs, and a parametric penalty in case of shortages under extreme scenarios.

The optimal logistics configuration in almost all the studies was identified by obtaining the exact solution to the 2SP model through an optimal solver. The reason for adopting the optimisation solver is that only a reasonable number of possible scenarios were derived from the adopted case study and the probability of a scenario s occurring is independent of the decisions

made in the previous time steps. In other words, the uncertainty was not defined as endogenous. In addition, it is worth mentioning that the calculation of transportation cost in most of the previous studies only considered the shipping quantity, instead of evaluating the uncertain road condition which may have a prominent influence toward the punctuality of the delivery.

7.1.3 Endogenous stochastic programming

The following section explores the research that focuses on endogenous stochastic problems, where the decisions directly affect the level of uncertainty (Jonsbraten 1998). This includes studies that consider the fluctuation in road condition as another major source of uncertainty during the modelling process.

Khaligh & Mirhassani (2016) solved a single-vehicle routing problem with stochastic demands, where the actual demand of a customer is revealed only once the customer is visited. Considering the problem of hospital operation room scheduling, Hooshmand et al. (2018) developed a genetic algorithm to solve the endogenous time-stochastic scheduling problem. Vayanos et al. (2011) studied the production planning problem in offshore oil fields considering endogenous uncertainties.

Nohadani & Sharma (2008) conceived a generalised robust linear optimisation model applied to a shortest path problem. The algorithm considers arc lengths to be stochastic and affected by routing decisions. Similar research has been carried out by Boyles & Waller (2010), who use a minimum cost flow problem.

On another thread, several researches have focused on reducing the number of non-anticipativity constraints (NACs) in multi-stage stochastic programming models with endogenous uncertainties. The methods adopted included sequential scenario decomposition heuristics and Lagrangean decomposition (Gupta & Grossmann 2011, Apap 2017).

The studies analysed above focus on the reduction of NACs, which grow exponentially in size with larger problem instances. In practice, the algorithms developed are applied to scenarios that deviate significantly to our own.

Only Khaligh & Mirhassani (2016), Nohadani & Sharma (2008) and Boyles & Waller (2010) study the VRP or similar formulations, while others focus on scheduling problems. However,

Khaligh & Mirhassani (2016) solved the stochastic demand VRP, and used simplifications based on vehicle payload to reduce the scenario tree. Travel time is modelled as a stochastic parameter by Nohadani & Sharma (2008) and Boyles & Waller (2010), but both studies propose a static problem where variable uncertainties do not condition future decisions.

Thus, an important contribution of this chapter is the proposal of a mathematical formulation for the time-discretised endogenous stochastic travel time multiple VRP, assisted with aerial assessment. A unique aspect of our work is that both agents (UAV and truck) are capable of reducing network uncertainty. The above research was consulted when formulating the mathematical model presented in the next section.

7.2 Model Description

The model presented in this section captures the key decisions at the operational stage of humanitarian relief distribution in the aftermath of a natural disaster. The distribution of relief is carried out by ground vehicles assisted by UAVs, which assess the damage of the network. The objective is to reduce operational costs of delivering cargo by optimising UAV and ground vehicle routes over the mission time horizon. Note that the operational costs are calculated based on the vehicle travel times, such that a reduction in costs leads to a reduction in mission time.

A graph $G_v = \{V, E_v\}$ composed of nodes V and edges E_v represents the road network connecting urban settlements. It is assumed that the road network has been subjected to damage, the degree of which is stochastic at the mission start. This stochastic component relates to each edge $ij \in E_v$, modifying the travel time and travel cost of the link. In this problem, we consider a time horizon that is represented as a discrete set of time periods t such that $t \in T$. The stochastic component is time independent.

At every time step, UAVs are deployed to collect information on the road network, revealing the actual cost associated with damages of the network. The UAV exploration is captured by the binary decision variable b_{ij}^t , indicating that a link is explored at time step t_i if $b_{ij}^{t_i} = 1$ and $b_{ij}^t = 0 \ \forall t < t_i$. A scenario tree $s \in S$ captures all the potential routing options of the UAV.

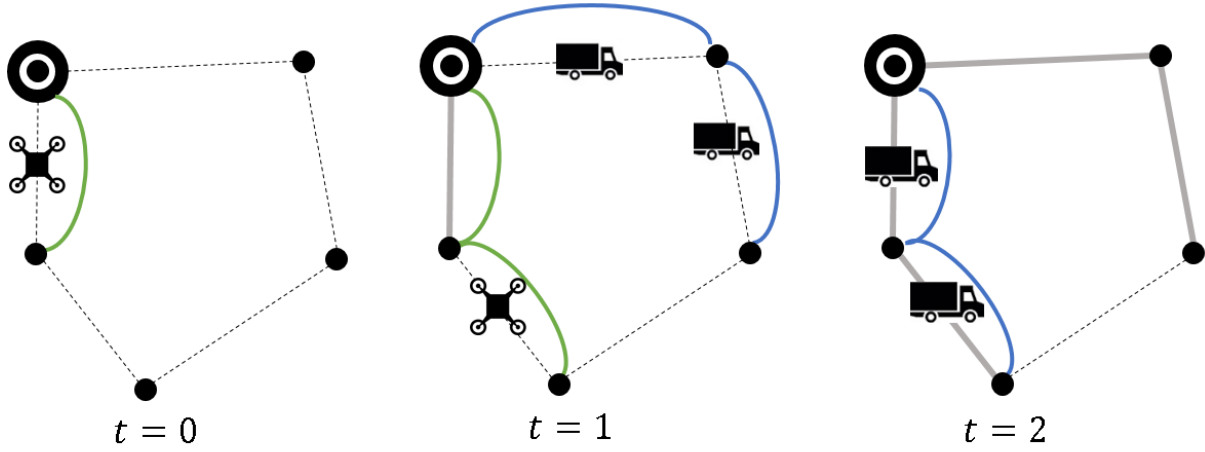


Figure 7.1: Truck and UAV routing example. A dashed line indicates an unexplored link, a gray line determines that the link has been explored when the vehicle is being routed, while the green and blue indicate the route followed by the drone and the truck respectively.

Note that UAVs are assumed to operate in Beyond Visual Line of Sight (BVLOS) conditions in spite of current drone regulations disallowing them to do so without prior clearance (Stöcker et al. 2017). However, UAV deployments have been permitted previously for road damage assessment in Ecuador and, at time of writing, WFP is collaborating with national authorities to develop regulations for humanitarian UAV deployments (DuPlessis 2016, Vornic 2017).

After the UAV is deployed at each time step, cargo vehicles traverse G_v transporting relief to every demand node $\forall i \in N$. Given common post-processing and analysis related to UAV damage assessment (FSD 2016), it is assumed that explorations at time step t_i inform routing decisions at time steps $t > t_i$. This behaviour is shown in Figure 7.1, where a drone is deployed at time $t = 0$, and the information gathered by that drone is available for routing at $t = 1$.

We adopt the following modelling assumptions:

Assumption 7.1: Ground and aerial vehicles must return to the depot at the end of the time step. A time step t corresponds to a single UAV and truck journey. While the UAV is assumed to be deployed before the truck, we consider that flight preparation, battery recharge, data consolidation and analysis take place during the same time step. This assumption is based on current case studies that require several hours to collect and analyse UAV data (DuPlessis 2016, FSD 2016). Consequentially, it is also assumed that the UAV battery is fully charged before each deployment.

Assumption 7.2: Average vehicle speeds are related to the Modified Mercalli Intensity (MMI). The MMI is a standardised measurement of the ground movement intensity. Hence, it is assumed that areas with greater ground movements are exposed to more severe damages, and lead to slower average vehicle speeds. At the start of the mission, respondents have no knowledge of the extent of link damages or the MMI score, thus assuming an average vehicle speed v_{avg} .

Assumption 7.3: Vehicle speeds are normally distributed. Variations in spot velocities are commonly modelled using normal distributions (Adeke et al. 2018, Naidu 2019). This model assumes that the same standard deviation occurs regardless of damage, but the distribution mean of each link is calculated based on Assumption 7.2.

The model presented requires two graphs $G_v = \{V, E_v\}$ and $G_d = \{V, E_d\}$ as inputs. The former represents the road network, and contains information on the deterministic pre-disaster transport travel time g_{ij} , deterministic damage travel time contribution d_{ij} , and stochastic damage time contribution $\hat{d}_{ij}$. G_d forms a fully directed graph that the UAV uses to traverse through the network. Edge set E_d contains information on the flight times f_{ij} and the approximate battery consumption q_{ij} , defined as percentage battery use. Vehicular payload capacity and node demand compose the remaining required inputs.

7.2.1 Mathematical formulation

Table 7.1: Variable, parameter and index definitions

Indices		Sets	
i, j	= nodes	E	= arc set
t	= time step	T	= time step set
s	= scenario	S	= scenario set
k	= vehicle/drone	K_v	= vehicle set
		K_d	= drone set
		V	= vertices
		D	= depots $D \subset V$
		N	= demand nodes $N \subset V$
Parameters		Variables	
f_{ij}	= flight time from i to j [hours]	x_{ijk}^t	= Boolean: indicates drone k explores link from i to j

e_{ij}	= exploration time from i to j [h]	$y_{ijk}^{s,t}$	= Boolean: indicates vehicle k explores link from i to j
g_{ij}	= travel time from i to j [h]	$r_i^{s,t}, h_i^{s,t}$	= excess demand at node i
q_{ij}	= battery consumption from i to j [-]	o_{jk}^t	= percentage battery charge of vehicle k at node j
d_{ij}^s	= damage travel time contribution from i to j [h]	w_{ijk}^s	= cargo flow from i to j by vehicle k
Q	= payload capacity [-]	b_{ij}^t	= Boolean: explored link
a_i^t	= demand of node i [-]	u_{ij}^t	= Boolean: new link explored at time t
p^s	= scenario probability [-]		
γ_{ij}^s	= scenario-link relation [-]		
c	= value of time [£/h]		
P	= demand penalty [£]		

The problem described above is consolidated into a single mathematical formulation. The reader is referred to Table 7.1 for the definitions of the variables and parameters used in the remainder of this section. The linearisation of the problem is presented in the Endogenous Stochastic Research Appendix.

$$Z = \sum_{t \in T} (Z_{uav}^t + Z_{exp}^t + Z_{prob}^t) \quad (7.1.1)$$

$$Z_{uav}^t = \sum_{ij \in E, k \in K_d} x_{ijk}^t f_{ij} c \quad \forall t \in T \quad (7.1.2)$$

$$Z_{exp}^t = \sum_{ij \in E} u_{ij}^t e_{ij} c \quad \forall t \in T \quad (7.1.3)$$

$$Z_{prob}^t = \sum_{s \in S} p^s \sum_{ij \in E_v} \gamma_{ij}^s u_{ij}^t (Z_{veh}^{t,s} + Z_{dmg}^{t,s} + Z_{unc}^{t,s} + Z_{pen}^{t,s}) \quad \forall t \in T \quad (7.1.4)$$

$$Z_{veh}^{t,s} = \sum_{ij \in E, k \in K_v} y_{ijk}^{s,t+1} g_{ij} c \quad \forall t \in T \quad \forall s \in S \quad (7.1.5)$$

$$Z_{dmg}^{t,s} = \sum_{ij \in E, k \in K_v} y_{ijk}^{s,t+1} d_{ij}^s b_{ij}^{t+1} c \quad \forall t \in T \quad \forall s \in S \quad (7.1.6)$$

$$Z_{unc}^{t,s} = \sum_{ij \in E, k \in K_v} y_{ijk}^{s,t+1} \hat{d}_{ij}^s (1 - b_{ij}^{t+1}) c \quad \forall t \in T \quad \forall s \in S \quad (7.1.7)$$

$$Z_{pen}^{t,s} = \sum_{i \in V} (r_i^{s,t} + h_i^{s,t}) P \quad \forall t \in T \quad \forall s \in S \quad (7.1.8)$$

The overall objective function is presented in equation (7.1.1), consisting of seven components

defined in (7.1.2-7.1.8). Equations (7.1.2-7.1.3) describe the two scenario independent objectives: UAV travelling cost and route exploration cost, respectively. The scenario dependent components are evaluated in Equation (7.1.4) and are regulated by p^s and γ_{ij}^s , indicating the probability for scenario s to occur and relating the explored link ij with scenario s , respectively. This approach allows us to only consider the damage levels of the explored link in the scenarios during truck routing. As such, the remaining components relate to the costs associated with truck routing.

Equations (7.1.5-7.1.7) calculate the total vehicle routing cost. The deterministic pre-disaster cost component is evaluated in (7.1.5), while damage contributions are considered in (7.1.6) and (7.1.7). The former uses a deterministic component based on explored links, while the latter considers a stochastic cost distribution as the traversed link was not previously explored.

It is important to note that the cost values determined in (7.1.2-7.1.7) are proportional to the vehicle travel time, as they are calculated using a value of time parameter c . Therefore, the formulation serves as a response time minimisation objective.

Finally, a penalty associated with undelivered demand is proposed in equation (7.1.8). The undelivered demand is calculated using the slack variables r_i^{st} and h_i^{st} , and parameter P relates to the value of life, which serves as a measurement of the social costs related to the response operation (Holguin-Veras et al. 2013).

The full mathematical problem is described as follows:

$$\text{Minimise } Z \tag{7.2.1}$$

Subject to:

$$\sum_{j \in V} x_{ijk}^t = 1 \quad \forall i \in D \quad \forall t \in T \quad \forall k \in K_d \tag{7.2.2}$$

$$\sum_{i \in V} x_{ijk}^t \leq 1 \quad \forall j \in V \quad \forall t \in T \quad \forall k \in K_d \tag{7.2.3}$$

$$x_{iik}^t = 0 \quad \forall i \in V : i \notin D \quad \forall t \in T \quad \forall k \in K_d \tag{7.2.4}$$

$$\sum_{i \in V} x_{jik}^t + \sum_{i \in V} x_{ijk}^t = 0 \quad \forall j \in V \quad \forall t \in T \quad \forall k \in K_d \tag{7.2.5}$$

$$o_{jk}^t = \begin{cases} 0 & \text{if } j \in D \\ \sum_{i \in V} x_{ijk}^t (q_{ij} + o_{ik}^t) & \text{if } j \notin D \end{cases} \quad \forall j \in V \quad \forall t \in T \quad \forall k \in K_d \quad (7.2.6)$$

$$1 \geq \sum_{i \in V} x_{ijk}^t (q_{ij} + o_{ik}^t) \quad \forall j \in V \quad \forall t \in T \quad \forall k \in K_d \quad (7.2.7)$$

$$\sum_{j \in V} y_{ijk}^{s,t+1} \geq 1 \quad \forall i \in D \quad \forall k \in K_v \quad \forall t \in T \quad \forall s \in S \quad (7.2.8)$$

$$y_{ijk}^{s,t+1} = 0 \quad \forall i \in N \quad \forall t \in T \quad \forall s \in S \quad (7.2.9)$$

$$\sum_{j \in V} y_{ijk}^{s,t+1} \leq 2 \quad \forall i \in V \quad \forall t \in T \quad \forall s \in S \quad (7.2.10)$$

$$\sum_{j \in V} y_{jik}^{s,t+1} - \sum_{j \in V} y_{ijk}^{s,t+1} = 0 \quad \forall i \in V \quad \forall k \in K_v \quad \forall t \in T \quad \forall s \in S \quad (7.2.11)$$

$$w_{ijk}^{s,t+1} \leq y_{ijk}^{s,t+1} Q \quad \forall ij \in E \quad \forall t \in T \quad \forall s \in S \quad (7.2.12)$$

$$\sum_{j \in V, k \in K_v} (w_{jik}^{s,t+1} - w_{ijk}^{s,t+1}) + r_i^{s,t+1} - h_i^{s,t+1} = a_i^{t+1} \quad (7.2.13)$$

$$\forall i \in V \quad \forall t \in T \quad \forall s \in S$$

$$u_{ij}^t \leq \sum_{k \in K_d} x_{ijk}^t \quad \forall ij \in E \quad \forall t \in T \quad (7.2.14)$$

$$u_{ij}^t = b_{ij}^{t+1} - b_{ij}^t \quad \forall ij \in E \quad \forall t \in T \quad (7.2.15)$$

$$b_{ij}^{t+1} \geq b_{ij}^t \quad \forall ij \in E \quad \forall t \in T \quad (7.2.16)$$

$$\text{Non - Anticipativity Constraints} \quad (7.2.17)$$

$$x_{ijk}^t = \{0, 1\} \quad \forall ij \in E \quad \forall k \in K_v \quad \forall t \in T \quad (7.2.18)$$

$$b_{ij}^t, u_{ij}^t = \{0, 1\} \quad \forall ij \in E \quad \forall t \in T \quad (7.2.19)$$

$$w_{ijk}^{s,t} \geq 0 \quad \forall ij \in E \quad \forall k \in K_v \quad \forall t \in T \quad \forall s \in S \quad (7.2.20)$$

$$y_{ijk}^{s,t} = \{0, 1\} \quad \forall ij \in E \quad \forall k \in K_d \quad \forall t \in T \quad \forall s \in S \quad (7.2.21)$$

$$0 \leq o_{jk}^t \leq 1 \quad \forall j \in V \quad \forall k \in K_v \quad \forall t \in T \quad (7.2.22)$$

$$r_j^{s,t}, h_j^{s,t} \geq 0 \quad \forall j \in V \quad \forall t \in T \quad \forall s \in S \quad (7.2.23)$$

The problem constraints are defined by (7.2.2-7.2.23). UAV movement is controlled by (7.2.2-7.2.5): constraint (7.2.2) ensures that the UAV starts at the depot, while (7.2.3) limits the UAV to visit a node once only. Constraint (7.2.4) allows UAVs to wait only at the depot, meaning

that the UAV may not be deployed if $i = j \forall i, j \in D$. An equal number of outgoing and incoming UAVs at each node i is defined in (7.2.5).

We limit UAV range based on the battery consumption matrix q_{ij} , implemented in (7.2.6-7.2.7). The variable o_{jk} is defined in (7.2.6) and represents the percentage battery consumption of the UAV k when arriving to node j . A value of 1 indicates that the battery has been fully consumed, while a value of 0 represents a fully charged battery. The value of o_{jk}^t is bounded to 0 when $j \in D$ to replicate its recharging behaviour. Constraint (7.2.7) ensures that battery levels never exceed their capacity.

Constraints (7.2.8-7.2.13) monitor the vehicle and cargo flows throughout the network, with (7.2.8-7.2.11) being the truck-equivalent to (7.2.2-7.2.5). Constraint (7.2.8) ensures that the vehicles start at the depot, (7.2.9) prohibits all vehicles to travel to the same demand node, (7.2.10) allows only two visits to each node, and (7.2.11) ensures vehicle flow equilibrium.

Constraints (7.2.12) and (7.2.13) monitor cargo flow through the network. The first limits cargo transportation to vehicle payload, while the second represents the cargo equilibrium constraints considering the slack variables ($r_i^{s,t}$ and $h_i^{s,t}$). These variables indicate the quantity of cargo that remains undelivered at each time step, and are used by the objective component (7.1.8).

Constraint (7.2.14) ensures that explored link u_{ij}^t can only be selected if a drone has traversed said link. The explored link variable b_{ij}^t is updated by (7.2.15), giving ground vehicles access to the actual link cost as per (7.1.6-7.1.7). Constraint (7.2.16) ensures that an explored link at time t_i remains explored in future time steps $t \geq t_i$. The Non-Anticipativity Constraints (NACs) in (7.2.17) are described in detail in Section 7.2.2.

The remaining equations (7.2.18-7.2.23) define the variable boundaries: $x_{ijk}^{s,t}$, b_{ij}^t , u_{ij}^t and $y_{ijk}^{s,t}$ are booleans, battery charge o_{jk}^t is bounded from 0 to 1, and cargo flows $w_{ij}^{s,t}$ and $r_i^{s,t}$, $s_i^{s,t}$ demand variables are positive.

7.2.2 Non-Anticipativity Constraints

The proposed problem formulation reveals actual link damage only if explored by a UAV or traversed by a vehicle in a previous time step. In other words, the timing of the revealed

information is decision-dependent, and depends on the available information at the start of the time step. Therefore, given two scenarios s and s' at time step t , if the information revealed is the same for s and $s' \forall t' \in T : t' < t$, the scenarios are defined as indistinguishable. NACs ensure that decisions made in indistinguishable scenarios s and s' at time step t remain identical.

The phenomenon is illustrated in Fig. 7.2, which presents a sample scenario tree with three stages and nine scenarios, where each branch represents the three damage levels of a newly explored link (based on 3-point approximation of normal distribution). In Fig. 1, all scenarios s at $t = 0$ are indistinguishable, illustrating the behaviour of our algorithm. Given that no damage has yet been assessed and no routing decisions have been made, all scenarios must be indistinguishable at this stage. Therefore, the NAC at $t = 0$ is:

$$x_{ijk}^{s,0} = x_{ijk}^{s',0} \quad \forall ij \in E \quad \forall k \in K_d \quad \forall s, s' \in S : s' < s \quad (7.3)$$

For $t > 0$, NACs determine both truck, cargo and UAV routing decisions, respectively. By referring again to Fig. 7.2, the NAC at $t = 1$ would state that $x_{ijk}^{1,t} = x_{ijk}^{2,t} = x_{ijk}^{3,t}$, as they all share the same scenario history. In the problem formulation above, the decision depends on the revealed link damages and unsatisfied demand nodes. For convenience, we define a variable $z_i^{s,t} = r_i^{s,t} + h_i^{s,t}$ that calculates all undelivered demand in scenario s at time step t . The NACs for $t > 0$ are defined in equations (7.4-7.6), considering drone, truck and cargo routing respectively.

$$\begin{aligned} & (if \sum_{ij \in E, t' \in T : t' < t} |d_{ij}^s u_{ij}^{t'} - d_{ij}^{s'} u_{ij}^{t'}| + \\ & \sum_{i \in V} |z_i^{s,t''} - z_i^{s',t''}| = 0 \text{ then } x_{ijk}^{s,t} = x_{ijk}^{s',t}) \end{aligned} \quad (7.4)$$

$$\forall ij \in E \quad \forall k \in K_d \quad \forall s, s' \in S : s' < s \quad \forall t, t'' \in T : t'' = t - 1, t > 0$$

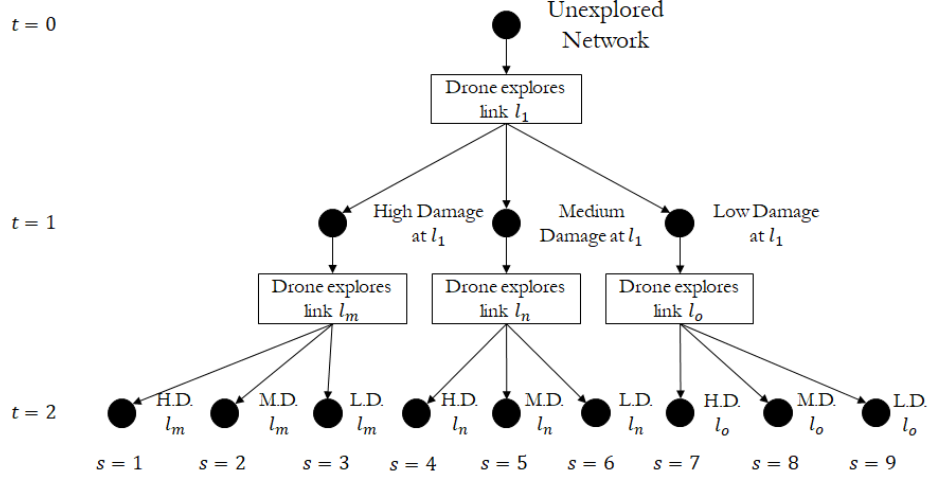


Figure 7.2: Scenario tree example where the normal distribution is modelled using the three-point estimation method.

$$\begin{aligned}
 & (if \sum_{ij \in E, t' \in T: t' < t} |d_{ij}^s u_{ij}^{t'} - d_{ij}^{s'} u_{ij}^{t'}| + \\
 & \sum_{i \in V} |z_i^{s, t''} - z_i^{s', t''}| = 0 \text{ then } y_{ijk}^{s, t} = y_{ijk}^{s', t}) \quad (7.5)
 \end{aligned}$$

$$\forall ij \in E \quad \forall k \in K_d \quad \forall s, s' \in S : s' < s \quad \forall t, t'' \in T : t'' = t - 1, t > 0$$

$$\begin{aligned}
 & (if \sum_{ij \in E, t' \in T: t' < t} |d_{ij}^s u_{ij}^{t'} - d_{ij}^{s'} u_{ij}^{t'}| + \\
 & \sum_{i \in V} |z_i^{s, t''} - z_i^{s', t''}| = 0 \text{ then } w_{ijk}^{s, t} = w_{ijk}^{s', t}) \quad (7.6)
 \end{aligned}$$

$$\forall ij \in E \quad \forall k \in K_d \quad \forall s, s' \in S : s' < s \quad \forall t, t'' \in T : t'' = t - 1, t > 0$$

7.3 Solution Method

Exact solution methods of the full scenario tree and NACs result in an exceedingly large problem size even for small instances. Equation (7.7) shows how the number of variables $\mathcal{N}$ relate to the number of time steps $\mathcal{T}$, edges $\mathcal{E}$, vertices $\mathcal{V}$, scenarios $\mathcal{S}$, drones $\mathcal{K}_d$ and ground vehicles $\mathcal{K}_v$. In turn, the number of scenarios equate to the total number of possible combinations of

link damages at each time step as defined by equation (7.8), where n indicates the number of link damage levels. As it can be observed, the number of scenarios increases non-linearly as the network size increases (number of links), and hence quickly becomes unsolvable for exact solvers.

$$\mathcal{N} = \mathcal{ET}(2 + \mathcal{K}_d + 2\mathcal{K}_v\mathcal{S}) + \mathcal{VT}(1 + 2\mathcal{S}) \quad (7.7)$$

$$\text{where } \mathcal{S} = n^{\mathcal{E}} \quad (7.8)$$

The formulation of the proposed problem instance is similar to a Time-Dependent Vehicle Routing Problem (TD-VRP), where the aim is to assign vehicles to a route to minimise total distance travelled. The problem is NP-complete, as it is a generalisation of the Travelling Salesman Problem, so any exact solution method is unscalable, and is not applicable to realistic problem instances.

Previous research relies on heuristics and meta-heuristics to overcome the scalability issues of exact solution methods for the TD-VRP (Gendreau et al. 2015, Kok et al. 2012) and multi-stage endogenous stochastic problems (Gupta & Grossmann 2011, Hooshmand & MirHassani 2016, Hooshmand et al. 2018, Apap 2017). Thus, the pursuit of exact solution for these kinds of problems is unfeasible within acceptable computational time and resource. Therefore, we propose two solution approaches: a greedy sequential method presented in Section 7.3.1, and a meta-heuristic described in Section 7.3.2.

7.3.1 Sequential algorithm

We reduce the number of variables to consider by using a greedy sequential model, solving each time step $t \in T$ sequentially. A workflow of the model is presented in Figure 7.3.

The process commences by generating an initial problem instance with randomised network damages s_p . An exact optimisation solver is used to optimise the routing decisions that minimise the objective function (7.1.1) for $t = 0$. The resulting drone and truck routings are used to update the network demand and network damages in order to carry forward the information gathered at $t = 0$. The process continues with increasing values of t until all demand is delivered.

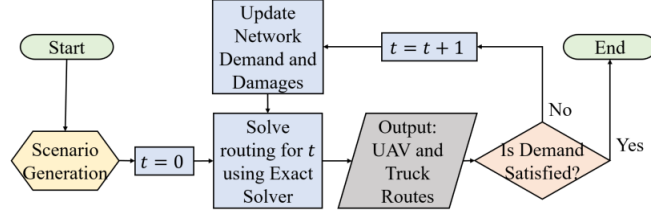


Figure 7.3: Sequential algorithm workflow

The proposed solution ensures that at each step t_i the exact solver minimises the operational cost given the cumulative knowledge gathered for $t < t_i$. However, the main problem of the sequential method is that it ignores potential future decisions for $t > t_i$. Thus, the greedy decisions made at t_i , while optimal for the proposed time step, may result in deviations from the optimal problem solution.

In exchange for a reduced solution optimality, the greedy sequential model eliminates the need to implement NACs and reduces the scenario tree complexity. The latter is shown in equations (7.9)-(7.10), where the number of combinations is dependent on the number of links $\mathcal{E}$, and the number of explored links per time step $\mathcal{B}$. As a result, the approach considers only a limited number of explorations at every time step. The potential number of scenarios becomes quickly infeasible given the total combinations of link explorations.

$$\mathcal{N} = \mathcal{E}(2 + \mathcal{K}_d + 2\mathcal{K}_v\mathcal{S}) + \mathcal{V}(1 + 2\mathcal{S}) \quad (7.9)$$

$$\text{where } \mathcal{S} = \mathcal{E}^{\mathcal{B}} + 1 \quad (7.10)$$

Hence, we enforce $\mathcal{B} = 1$, following current UAV range limitations, image post-processing and data analysis requirements for damage assessment (Soesilo & Sandvik 2016, FSD 2016). Furthermore, the greedy nature of the approach provides a lower-bound solution of the original problem. By developing a Genetic Algorithm, we can extend the problem to multiple drones, vehicles and explorations per time step, as well as eliminate its greedy nature.

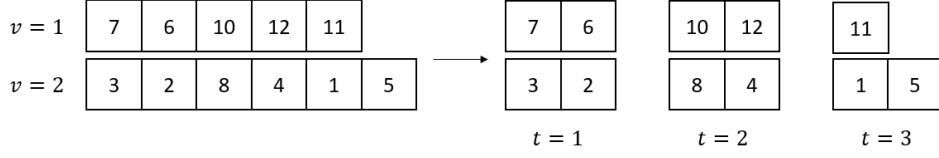


Figure 7.4: Chromosome representation.

7.3.2 Genetic Algorithm

A Genetic Algorithm (GA) is a widely used meta-heuristic developed in the 1970s that employs probabilistic-based search to explore the solution space (Davis 1991). GAs have been used extensively to solve the VRP due to its linear scaling with problem size (Karakatič & Podgorelec 2015). Its performance level has been shown to be on par with other heuristics and for some particular problem instances, have provided the best-known solutions ever recorded (Berger & Barkaoui 2003, Guezouli & Abdelhamid 2017). Therefore, we selected the GA as an appropriate solution method for our proposed problem.

Five core components govern the behaviour of the GA: solution generation, evaluation, selection, crossover and mutation. The length of the search, governed by the number of iterations σ , and the population size z alter the optimality level the GA can achieve. The structure of the components is presented in Algorithm 6.

The generation phase creates an initial solution set. Every solution is then evaluated using equation (7.1.1), which returns a scalar fitness value indicating its optimality. The implementation of the evaluation stage is presented in Algorithm 2, which is called by Algorithm 7 in Line 7.

The selection phase determines which solutions are passed down to the next steps. We propose a modified roulette method that associates a selection probability to each solution based on their fitness value, such that more optimal solutions have higher probability of being selected. This allows poorer solutions to be selected in benefit of gene diversity.

Finally, crossover and mutation processes constitute the main solution manipulation methods. The former combines two solutions, while the latter alters a solution individually. The probability of occurrence of each process is controlled by parameters p_c and p_m , respectively. These randomised alterations aid the GA to avoid local optima, allowing it to navigate through

complex search spaces.

Algorithm 6 Genetic Algorithm	Algorithm 7 Fitness Evaluation
<pre> 1: function GA 2: Input: network G, vehicles n_k, drones n_d, generations σ, crossover and mutation probabilities p_c, p_m, population size z 3: Output: optimal solution s^* 4: $i = 0$ 5: $s \leftarrow \text{Generation}(z, n_k, n_d)$ 6: while $i \leq \sigma$ 7: $f \leftarrow \text{Evaluation}(s, G)$ 8: $s \leftarrow \text{Selection}(s, f)$ 9: $s \leftarrow \text{Crossover}(s, p_c)$ 10: $s \leftarrow \text{Mutation}(s, p_m)$ 11: $i = i + 1$ 12: return $s^* \leftarrow \text{GetBest}(s, f)$ 13: end function </pre>	<pre> function EVALUATION Input: solution s, network G Output: fitness f time steps $T \leftarrow \text{GetMissionTime}(s)$ discretised mission $m \leftarrow \text{Discretise}(s, T)$ $f = 0$ for t in T do $G \leftarrow \text{UpdateExploration}(m_{t-1}, G)$ paths $p \leftarrow \text{Dijkstra}(m_t, G)$ drone cost $c_d \leftarrow \text{DroneCosts}(m_t, G)$ truck cost $c_t \leftarrow \text{TruckCosts}(m_t, p)$ $f += c_d + c_t$ end for return f end function </pre>

Every solution in the GA contains two integer arrays indicating truck and drone routes respectively, where each drone and link is given a unique identification value. Assuming that each demand node has to be visited once, the solutions can be structured as a travelling salesman problem, where the only information contained is the schedule specifying the route sequence. Given a maximum payload allowance, the single array is then broken down into the separate time steps, as illustrated in Fig. 7.4. The same structure is followed in the drone routing, which determines the links to explore and discretises the route into time steps given a maximum battery capacity.

While the route followed by the UAV is easy to generate, given that the UAVs traverse a fully connected graph, trucks are restricted to a pre-defined road network. Therefore, at each time step t , the Dijkstra's algorithm (Dijkstra 1959) returns the shortest paths for each schedule using the estimated damage if $b_{ij}^t = 0$ and actual link damages if $b_{ij}^t = 1$.

The proposed structure facilitates solution generation and manipulation. Truck route generation consists on shuffling the order of demand nodes, and randomly assigning a node to each vehicle. A similar process is followed for drone routing generation, albeit not all links need to be explored.

Crossover selects a starting and ending point in a solution and exchanges the genes between the points with another solution. Finally, mutation contains generic operations that affect both

drone and truck routing, and specific operations applied to each independently.

The generic mutator operators include: inversion, which inverts the order of the vehicle route, self-swap, that exchanges two genes of the vehicle route, and intervehicle-swap, which exchanges a gene with a different vehicle's of the same solution. Specific to truck routing is a nearest neighbour operator that re-orders a set of genes by using the nearest neighbour heuristic given expected cost damages. Drone routing mutation contains insertion and deletion steps based on the links most traversed by the trucks.

The application of any of the mentioned processes is randomly selected. Each process contains a probability of activation, namely p_h , p_s , p_v , p_n , p_i and p_d for inversion, self-swap, intervehicle-swap, nearest neighbour, insertion and deletion, respectively. A single operation on both truck and drone routing is permitted for every mutation process.

7.4 Case study

We demonstrate our approach in a case study based on the road network of Port-au-Prince, the capital of Haiti. The region has been affected by numerous disasters in its history, and was subject to one of the deadliest earthquakes ever recorded in 2010 (Government of the Republic of Haiti 2010).

The case study area contains 25 cities that are interconnected by the highway system, consisting of 66 uni-directional links (Fig. 7.5). The international airport of Port-au-Prince represents the only port of entry to the network, and is established as the distribution depot. At every time step, UAVs and ground vehicles must leave and return to this depot, given constraints (7.2.2), (7.2.5), (7.2.8) and (7.2.11).

The humanitarian mission consists of delivering materials from the depot to the 25 cities. It is assumed that a single visit satisfies initial demand necessities, and that payload capacity limits delivery to two nodes in a single trip.

Vehicles must traverse a damaged network. Where damages are unknown, respondents assume a velocity $\hat{v}_{avg} = 40$. Otherwise, links are assigned an average velocity $v_{avg,MMI}$ dependent on the MMI measurements reported by USAID (2010), as shown in Table 7.2. The

Table 7.2: Model parameters.

Parameter	Symbol	Value
Underivered Cargo Penalty [$\mathcal{L}$]	P	500
Expected Vehicle Velocity [km/h]	$\hat{v}$	40
Expected Velocity Deviation [km/h]	σ	5
Payload Capacity [-]	Q	2
Node Demand [unit]	a_i^0	1
Value of time [$\mathcal{L}/h$]	c	55
Average Speed at MMI 5 [km/h]	$v_{avg,5}$	60
Average Speed at MMI 6 [km/h]	$v_{avg,6}$	50
Average Speed at MMI 7 [km/h]	$v_{avg,7}$	40
Average Speed at MMI 8 [km/h]	$v_{avg,8}$	30
Average Speed at MMI 9 [km/h]	$v_{avg,9}$	20
Average Speed at MMI 10+ [km/h]	$v_{avg,10}$	10

selected speeds have previously been recorded in post-earthquake environments (Mas et al. 2012, Wisetjindawat et al. 2014).

Damaged link velocities are generated using a normal distribution, where the mean of the distribution coincides with the average velocities shown in Table 7.2. Note that these do not mean that the vehicle travels at that speed throughout the whole link, but represents the delay caused by the network damages.

7.4.1 Sequential Algorithm Results

A smaller problem instance was prepared given the limitations of the sequential greedy algorithm outlined in Section 7.3. The eastern region of the network is evaluated, which consists of 10 nodes and connected by 22 links (refer to Figure 7.6). As an initial test run, we assume a constant MMI of 7.

We run the computational experiment on a workstation with a 64-bit Intel Core i7-4790 CPU and 8 GB RAM. Python 3 and Gurobi 8.0 were used to model and solve the algorithm. The experiment is repeated for 1000 scenarios, where link damages are randomly and independently assigned using a normal distribution. Each iteration runs in 16 seconds, with some variation.

To provide an insight into the solution structures, we present the routing behaviour of the worst-case scenario (largest percentage network damage) in Fig. 7.6, with a detailed breakdown

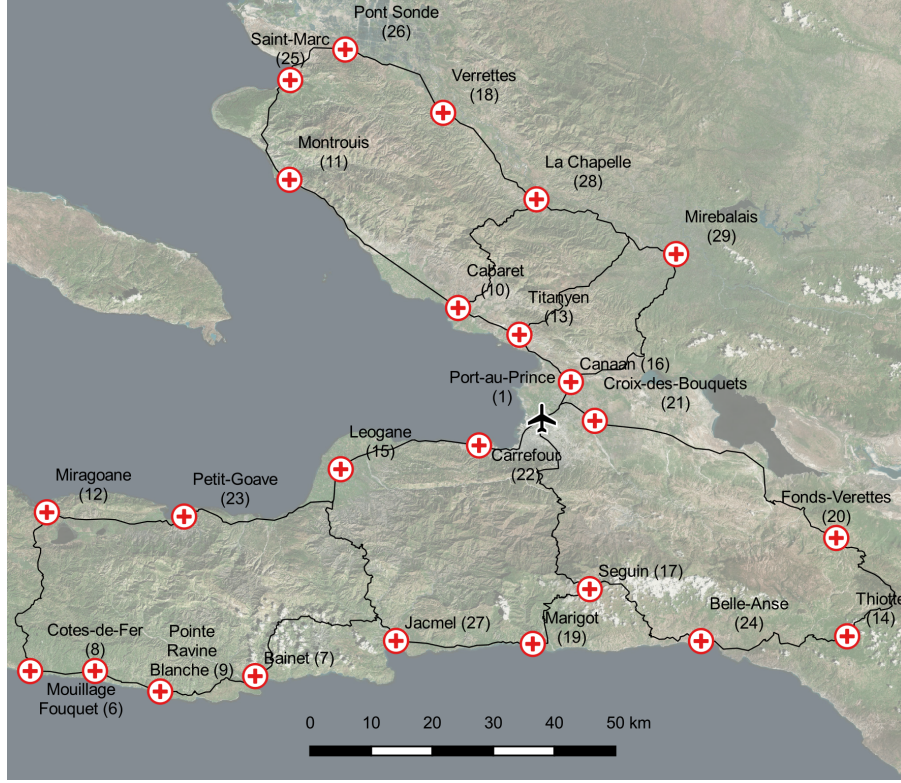


Figure 7.5: Case study road network in Haiti. Data from OpenStreetMap.

of the routing decisions presented in Table 7.3. Note that each direction of the link is assumed to be explored separately. To show the information that trucks have access to, we present UAV explorations at time step t that correspond to UAV deployments at $t - 1$. In other words, a UAV deployed at time step 0 is shown in Fig. 7.6 to occur at time step 1.

It can be observed that the UAV collects the relevant information for the vehicle, as in most cases, the route explored by the drone is immediately utilised by the ground vehicle in the following time step. The only exception is the use of the UAV to explore link 8-4 at time step 4, which provides an alternate route to enter and exit the depot in case the damage levels of neighbouring links is too high.

We compare the performance of our algorithm with a deterministic approach, where routing decisions are made based on expected link costs, with our results summarised in Fig. 7.7. We observe that the stochastic approach provides significant travel time savings in all scenarios, with larger differences observed at greater network damages. On average, the stochastic method provided 25% reductions in truck routing travel time, as well as lower cost variance.

A similar behaviour is shown in Fig. 7.8, which presents the results of a further run carried

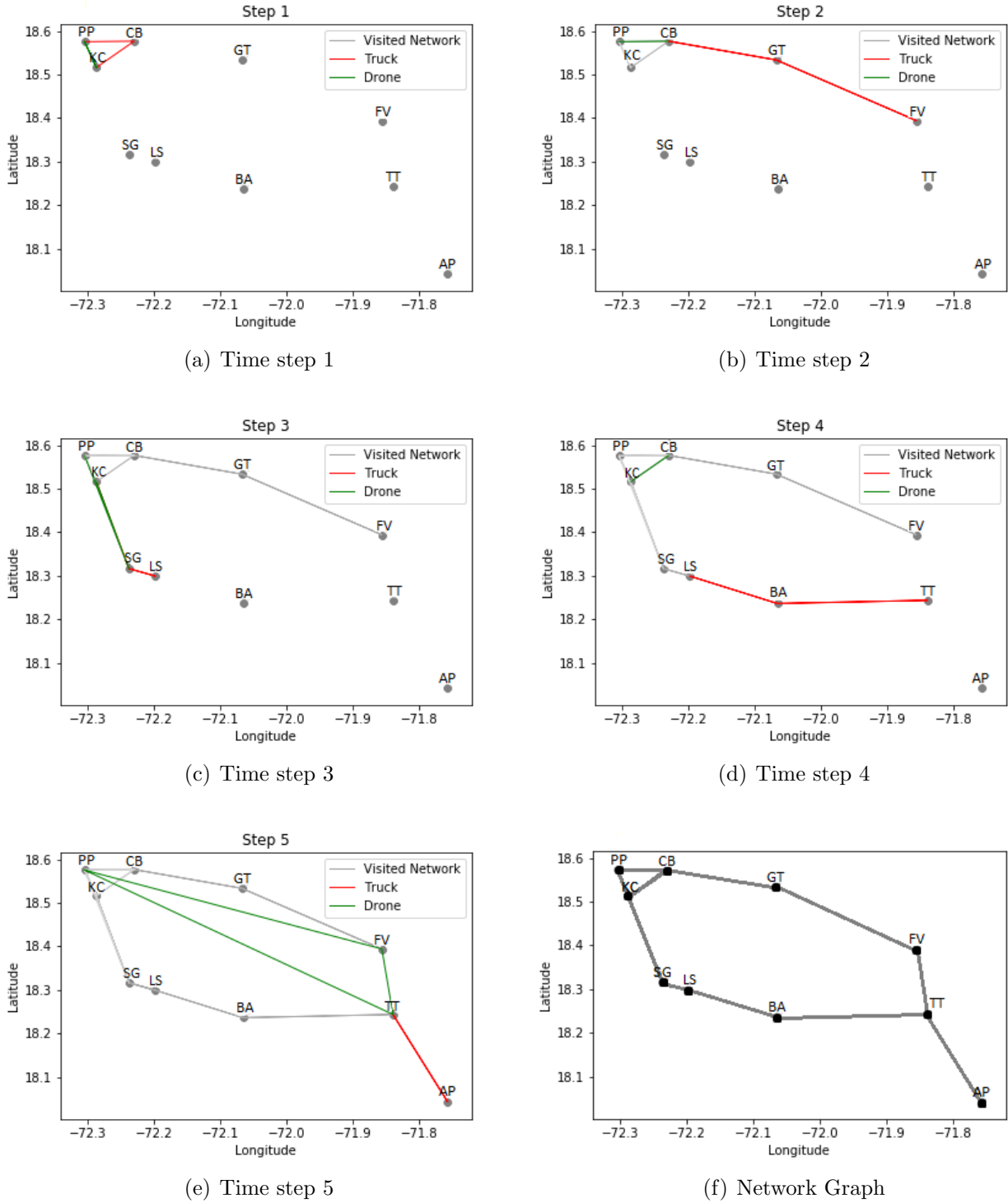


Figure 7.6: Truck and UAV routing per time step.

Nodes:

Port Au Prince (PP), Croix-des-Bouquets (CB), Ganthier (GT) Fonds-Verrettes (FV), Thiotte (TT), Anse-a-Pitre (AP), Belle-Anse (BA), La Serre (LS), Seguin (SG), Kenscoff (KC).

Table 7.3: Route itineraries in worst case scenario. Explored links in every time steps are highlighted in bold. Node numbers refer to Fig. 7.6

Time step	Drone route	Truck route
1	PP-KC -PP	PP- KC - CB -PP
2	PP-CB -PP	PP- CB - GT - FV - GT - CB -PP
3	PP- SG - KC -PP	PP- KC - SG - LS - SG - KC -PP
4	PP- CB - KC -PP	PP- KC - SG - LS - BA - TT - BA - LS - SG - KC -PP
5	PP- FV - TT -PP	PP- CB - GT - FV - TT - AP - TT - BA - LS - SG - KC -PP

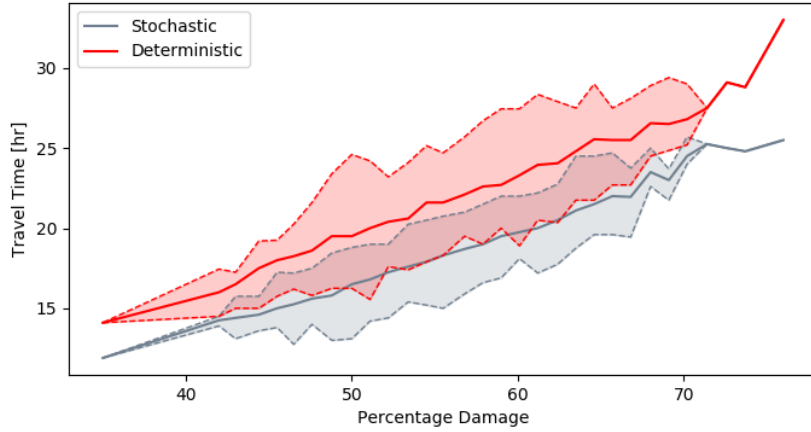


Figure 7.7: Deterministic and stochastic average vehicle routing travel times with network damage. The shaded region indicates the solution range.

out with and without UAV support, but still considering cumulative network knowledge based on previous truck routing decisions. In this case, the UAV-supported response also provides shorter travel times and lower deviations under all network damage configurations compared to the unsupported relief distribution, with average travel time savings of 20%.

7.4.2 Genetic Algorithm results

The Genetic Algorithm (GA) is developed using Python 3 and executed on the same workstation. Parameter values of $p_c = 0.5$, $p_m = 0.1$ and $z = 100$ are used based on the results of manual tuning, presented in the Appendix. Mutation operator parameters are tuned such that all operators have the same probability of being activated. The number of generations is kept to a constant 100. The GA runs without an exploration limit - the UAV is capable of exploring multiple links in the same time step.

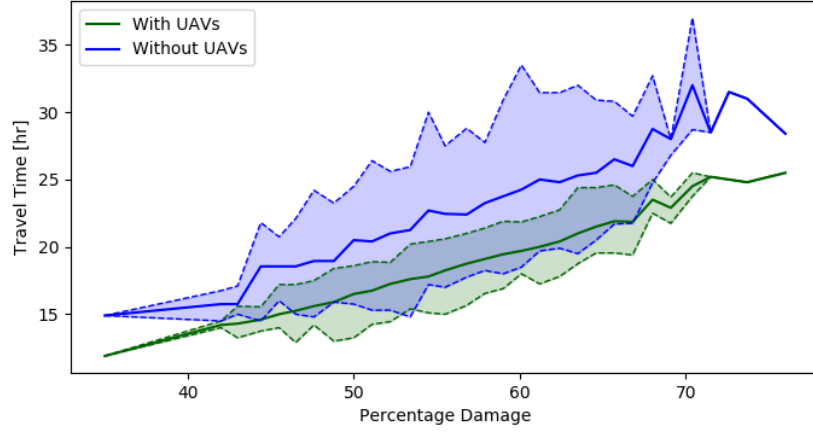


Figure 7.8: Stochastic average vehicle routing travel times with network damage under UAV support. The shaded region indicates the solution variability.

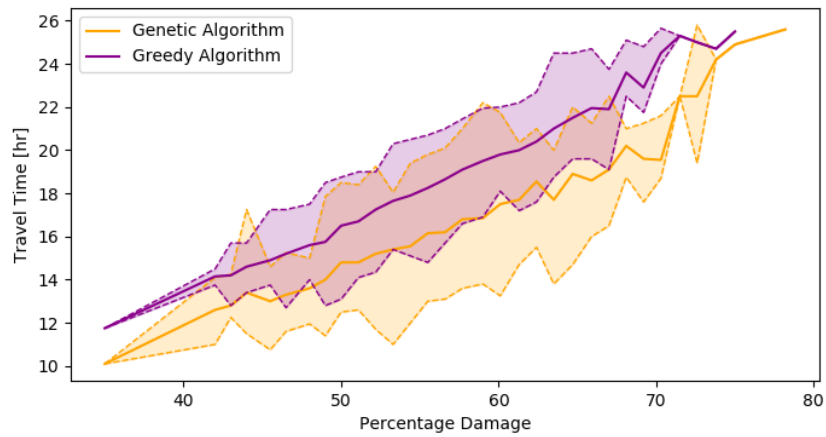


Figure 7.9: Average routing costs with network damage for the sequential greedy and GA solution methods.

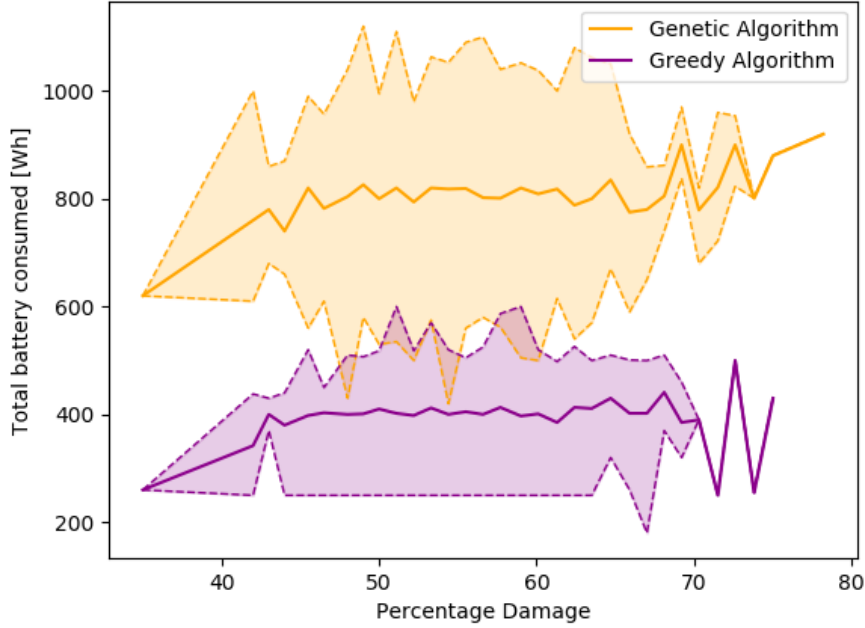


Figure 7.10: Average energy use by drones for the sequential greedy and GA solution methods.

In terms of total travel time, the GA provides significant performance improvements compared to the sequential greedy algorithm. Reductions between 10-20% can be observed in Fig. 7.9 depending on network damage levels. The main disadvantage of the GA method is the increased variability in solutions as a result of its probability-based search. As an example, the GA provides upper bound solutions at a similar level to the greedy algorithm between 50-60% network damage, despite the average travel cost being approximately 20% lower.

Fig. 7.10 shows that battery discharge rates increase significantly when using the GA in comparison with the sequential algorithm. The ability to visit multiple links in the same time step triples the total battery consumption compared to the greedy case. This is expected, as logically the exploration of multiple links in a single deployment requires longer flight times, hence greater battery use than the greedy approach. Similar to Fig. 7.9 the GA algorithm provides a greater solution variability due to its stochastic nature.

UAV performance

In this section we compare the performance of the GA with and without UAV assistance. To achieve this, we define perceived cost as the cost estimation made by respondents to traverse a

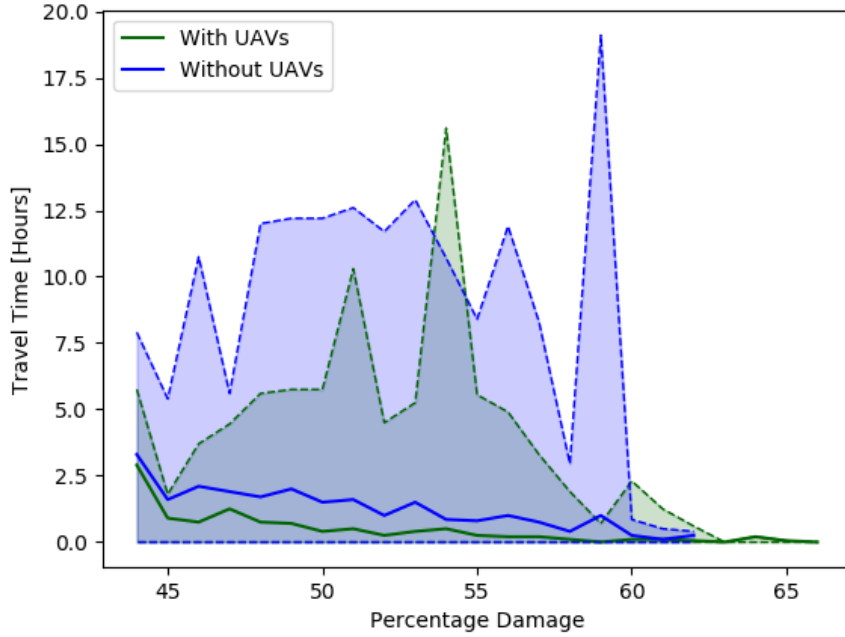


Figure 7.11: Deviation between perceived and actual path cost with and without drone assisted reconnaissance.

certain path given the cumulative collected knowledge of the network damages. This means that a path of cost c may be perceived as c_v , and if $c_v < c$, the respondents may be routed to a worse path than expected.

Figure 7.11 presents the percentage deviation between the perceived costs and the actual cost for the traversed truck paths. One can see that the drone-assisted respondents have a lower deviation between perceived and actual routing costs than unassisted respondents. In addition, the maximum cost deviations are lowered by 5-8% on average.

While the improvements do not seem significant at first, the current figure does not take into account how the UAV aids with route selection. Figure 7.12 presents the cost deviations between selected and best available path. As observed, the UAVs provide near 0% deviation from the best path at all damage levels, meaning that the vehicles are able to select the best route possible to each destination in most cases. Furthermore, the average path travel time deviation ranges 2 and 6%, and the cost deviation variability is also significantly reduced.

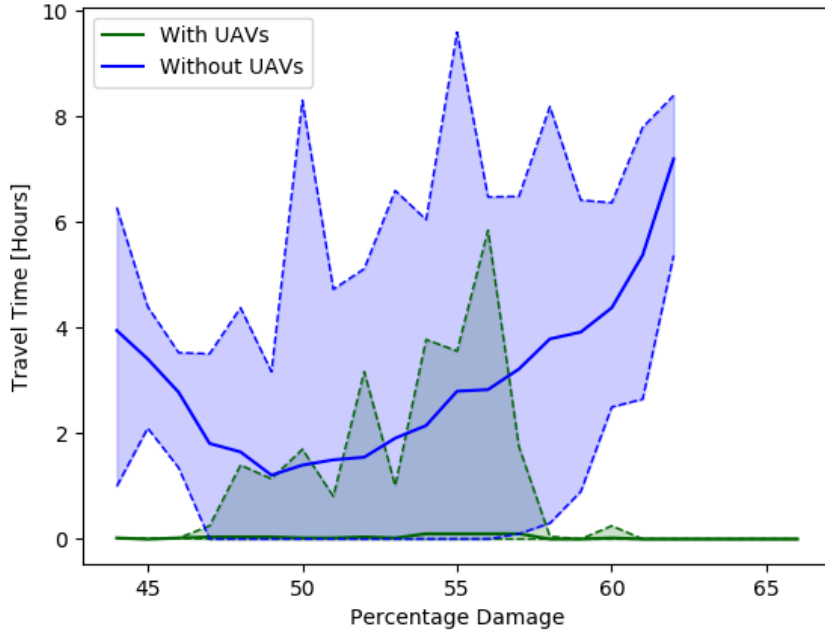


Figure 7.12: Deviation between chosen and best possible path traversed with and without drone assisted reconnaissance.

Penalty parameter variation

The penalty parameter P quantifies the cost of delaying the delivery of relief at a node. The intended purpose of the parameter is to minimise the mission length. As a result, it has an effect on the routing decisions made by the algorithm, as greater P values incite faster responses. Conversely, this allows less time for the UAV to explore the network, and may result in higher uncertainty encountered by the vehicle.

We evaluate the mission length with variations in the P parameter in Figure 7.13. As expected, low values of P result in significantly longer mission times (a maximum mission time of 20 days is allowed). The greatest reduction is observed between 50 and 150, where the penalty exceeds the exploration costs.

Large study case

A final set of experiments is carried out that simulates the relief response to the full 25 node network. A penalty parameter $P = 500$ is selected, as informed by the results in Section 7.4.2. Initially, we assume that two vehicles and two drones are deployed simultaneously given the

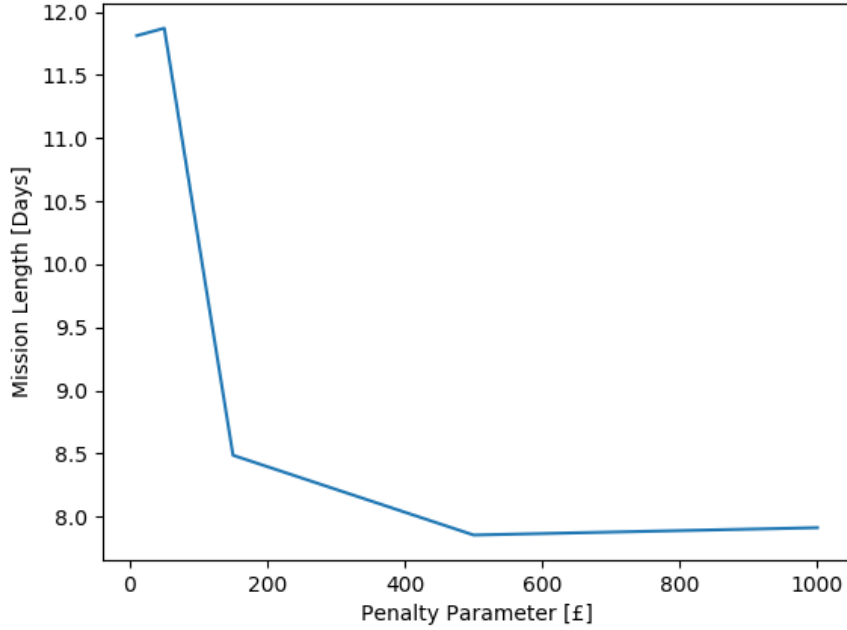


Figure 7.13: Average mission time variation with penalty parameter value.

larger network size.

Similarly to the small case study, both the UAV-assisted and the non-assisted problem instance have comparable deviations between the perceived and actual path cost for their chosen itineraries (Figure 7.14). However, Figure 7.15 shows that the smaller deviations are observed when estimating the cost of the best available path. As a result, the UAV assisted response provides shorter paths to the destination. As seen in Figure 7.16, the UAV assisted response reduces the cost between the best possible path and the actual path in all damage levels. The constant 2-3% benefit mirrors the difference observed in Figure 7.15.

Consequently, the UAV-assisted run not only reduces the average routing travel time by 10-20% at all damage levels, but the deviation decreases significantly throughout. Therefore, the UAV does generate a benefit to the humanitarian response, as it consistently provides information that allows trucks to select better routes to each destination consistently.

When comparing the response with different number of UAVs, we obtain evident improvements when using two UAVs compared to one (Figure 7.19), while additional UAVs provide very similar results (Figure 7.20). A critical level of knowledge of the network is obtained when two UAVs are used, and adding more UAVs has minimal impact on the optimality of the routing.

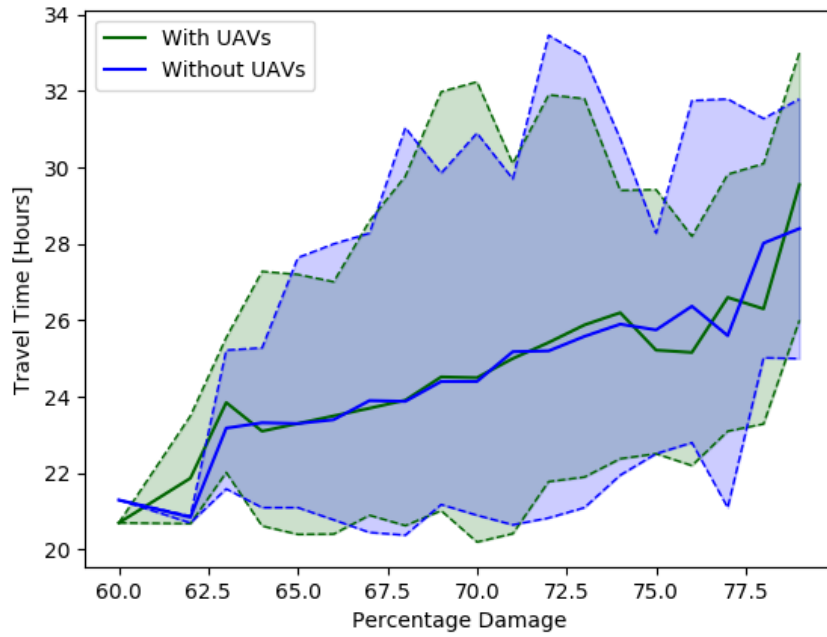


Figure 7.14: Deviation between perceived and actual path cost with and without drone assisted reconnaissance in 25 node case.

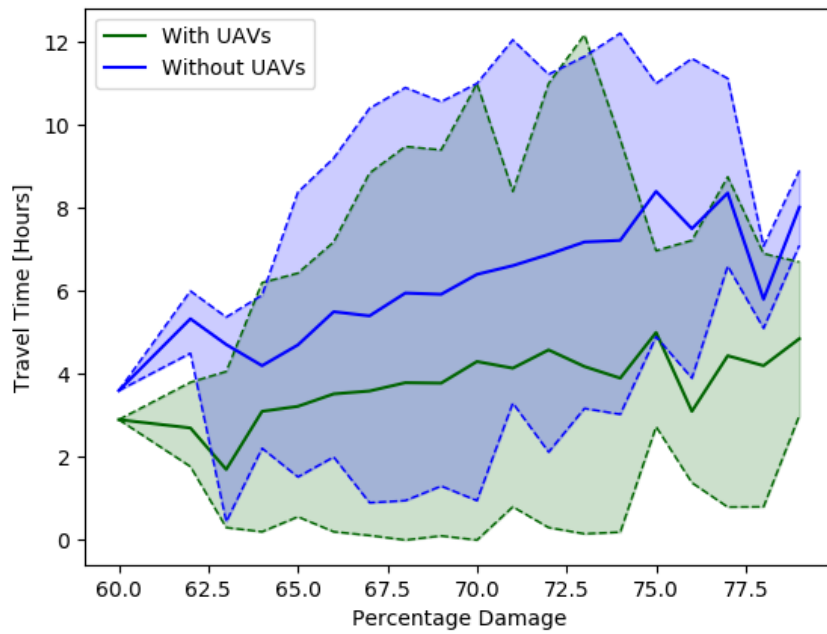


Figure 7.15: Deviation between perceived and best path cost with and without drone assisted reconnaissance in 25 node case.

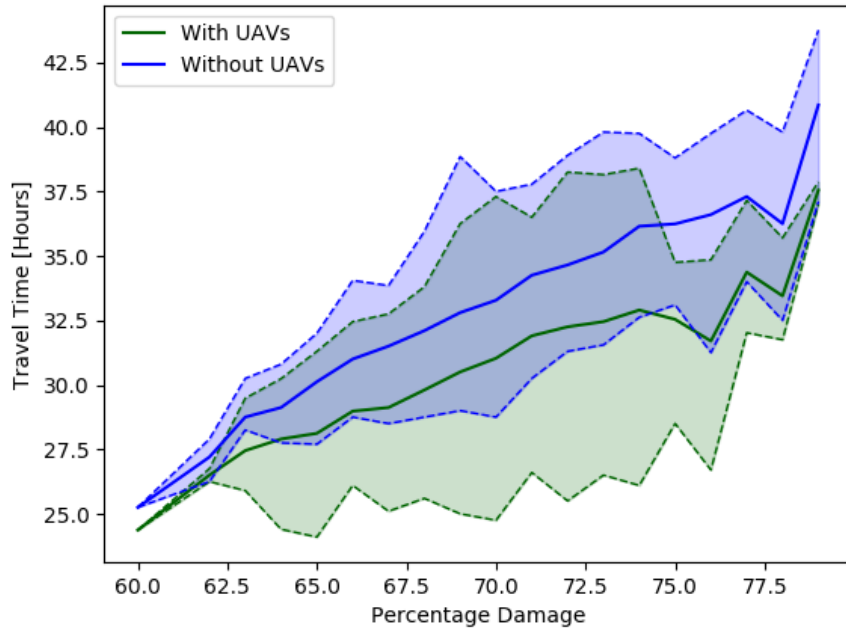


Figure 7.16: Deviation between chosen and best possible path traversed with and without drone assisted reconnaissance in 25 node case.

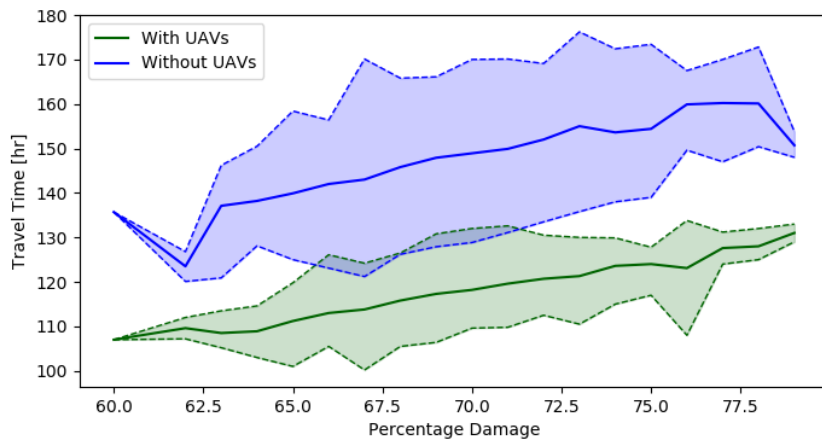


Figure 7.17: Routing costs with and without UAVs in 25 node case.

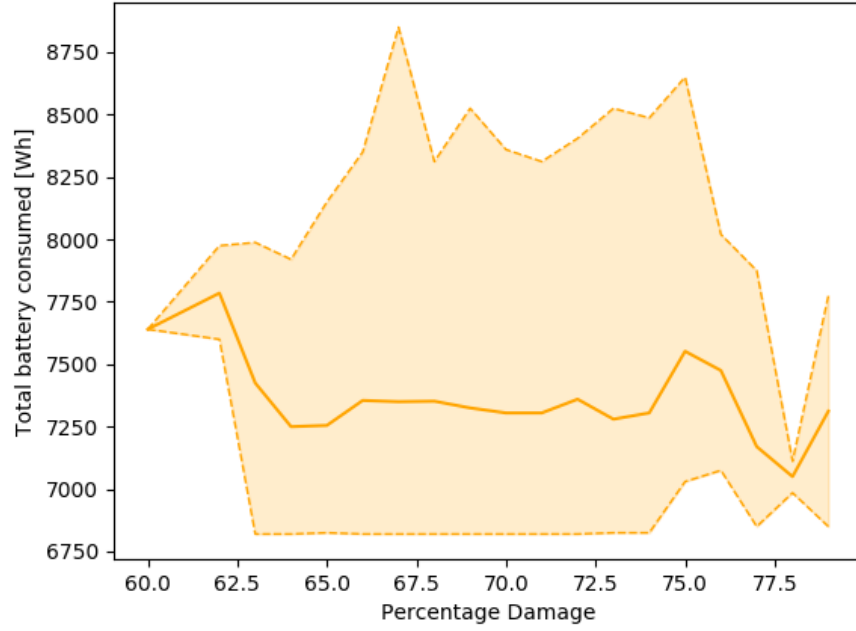


Figure 7.18: Average energy use by drones with network damage for the GA in the 25 node case.

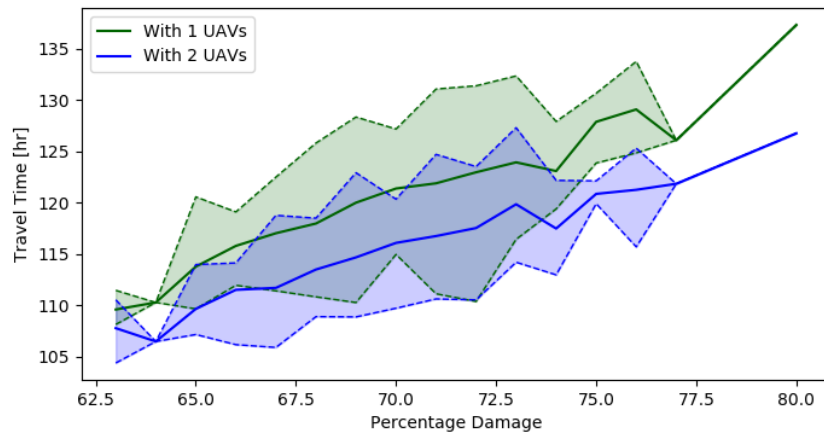


Figure 7.19: Routing costs with 1 and 2 UAVs in 25 node case.

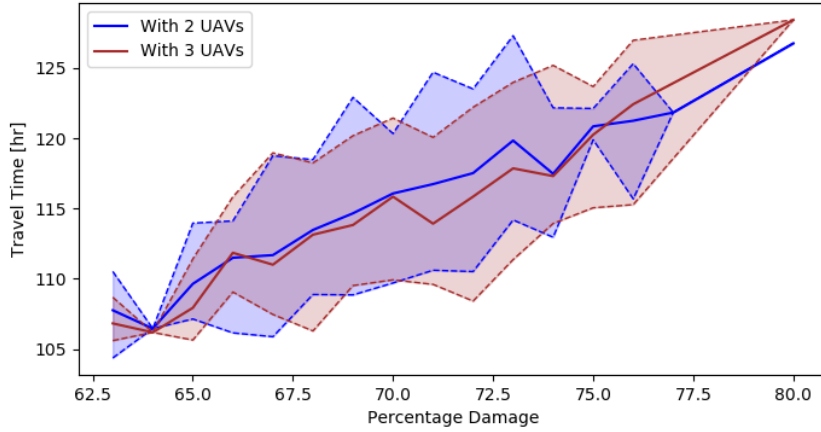


Figure 7.20: Routing costs with 2 and 3 UAVs in 25 node case.

7.4.3 Discussion

Our analysis has shown that UAVs provide significant improvements to humanitarian response operations by reducing uncertainty in the road network state and informing routing decisions. However, several barriers limit their widespread implementation. In first instance, current regulations limit UAV flight in BVLOS conditions (Stöcker et al. 2017). Despite permissions being provided by aviation authorities on a case-by-case basis, current processes are long and would inevitably lag humanitarian operation timelines.

Further simplifications with respect to aerial conflicts have been made. Given the low altitude requirements of road damage inspection, the latter assumption is reasonable in areas where drone activity is low. However, the rapid increase of drone use will necessitate efficient trajectory management algorithms for humanitarian use.

Additionally, our current methodology assumes optimal weather conditions, meaning that flight range deviations as a consequence of temperature variability, precipitation and wind are not considered. Furthermore, localised weather events in the form of rain or low visibility may reduce routing options at different time steps, forcing the UAV and vehicle to explore alternative regions. This simplification limits the applicability of the algorithm as it cannot capture energy requirements accurately, especially in weather-related disasters.

In the first instance, our exact approach considers that a single link is explored at every time step, as to limit the number of scenarios to be optimised. While the assumption may

be appropriate when only one UAV is used, given its range limitations and data processing timelines, the use of multiple UAVs for widespread exploration requires a relaxation of this rule. However, traffic management systems and regulations that will permit such operations are in development at the time of writing, and are expected to be operational in the near future (EASA 2015, Kopardekar et al. 2016, Chakrabarty et al. 2019). Thus, we expect that concerns regarding BVLOS drone operations will be addressed in due course by the relevant authorities.

Further extending the UAV range would require the location of recharge hubs. This includes allowing UAVs to depart from and return to different locations, and selecting optimal locations for the recharge hubs.

From an algorithmic perspective, the improvements presented above would increase algorithm runtime exponentially, as any of the relaxations above results in new exploration combinations. In addition, providing non-greedy solutions to this problem would require the development of intricate scenario-trees that are dependent on routing decisions made throughout the mission time horizon.

7.5 Concluding Remarks

Road damage remains a key factor that delays relief distribution. To overcome this problem, humanitarian organisations are increasingly relying on UAVs for rapid damage assessment. While UAV routing has been explored by researchers in the context of logistics, this chapter seeks to formalise and solve the UAV-assisted vehicle routing problem with endogenous uncertainty.

In particular, the algorithm presented in this chapter coordinates UAV and truck deployments simultaneously over the course of a relief distribution operation, where trucks deliver relief resources and UAVs assess the network damages, which are assumed to be unknown at the start of the mission. A sequential greedy stochastic optimisation method and an adapted genetic algorithm are applied to a case study based on the 2010 Haiti earthquake. Our analysis indicates that our algorithms provide significant performance gains compared to deterministic methods, which are widely used in the literature.

Technological limitations still hinder widespread adoption of UAVs in humanitarian opera-

tions. Their susceptibility to changing weather patterns, as well as maintenance and required access to electricity bring further uncertainty given potential failure states mid-flight. Collision avoidance further adds an additional operational risk. Taking them into account necessitates the development of trajectory management algorithms for humanitarian operations, such as the algorithms developed in Chapter 6. In the metropolitan context, the Urban Air Mobility (UAM) concept has been introduced to manage UAV operations. These aspects have not been considered in this chapter.

The proposed problem and solution method represent a foundation from which future research may be conducted. Improvements can be carried out to relax several simplifications that limit implementation. In the context of earthquake response, aftershocks may induce further network damages, meaning data collected by UAVs may become outdated. Introducing this time-dependent and stochastic damage component would require the development of a complex link damage model.

Further extensions to the algorithm presented in this paper could implement multi-depot operations, expand the case study size, and provide alternative objective functions, such as evaluating service equity and safety. Aspects of coordination can be evaluated by including multiple independent organisations. Another stream of research could seek to reduce the NACs, which would provide significant improvements to the applicability of exact solutions methods. These will compose our future work.

Chapter 8

Conclusions and future work

This chapter presents the findings of the thesis and compares them against the research objectives described in Chapter 1, analysing the contributions and limitations of the research and their impact on the different stakeholders. Suggestions for future work are also provided.

8.1 Research Objectives Analysis

The analysis carried out in this section considers four key stakeholders during humanitarian response: humanitarian actors, governments, industry partners, and academics. The findings of this thesis are mapped to these stakeholders and the research objectives introduced in Chapter 1, which are presented below for the reader's convenience:

- Obj. 1. Develop a quantitative framework for measuring UAV-based relief distribution which considers ethical principles including equity and fairness (Bagshaw 2012). Further UAV-specific features are incorporated that relate to safety, energy use and access, which are implemented using complex objective functions.
- Obj. 2. Propose a last-mile relief distribution design algorithm for UAV-based humanitarian relief operation. The algorithm optimises key decisions of hub location and routing, and reflects the sequential decisions made during humanitarian operations, thus considering uncertainty and imperfect data (Boyles & Waller 2010, Özdamar et al. 2014).

Obj. 3. Develop trajectory design algorithms with accurate energy expenditure estimations.

While the linearisation of fuel consumption relationships is reasonable for road vehicles, this assumption misrepresents the effectiveness and operational range of UAVs, particularly when consideration variations due to weather and payload sizes. Hence, the nonlinear behaviour of battery use is no longer negligible, and have to be considered in planning.

Obj. 4. Test and calibrate the proposed framework in large-scale scenarios. The case studies proposed are based on realistic disaster in scale and scope, which allows us to compare the effectiveness of UAV-based response with traditional vehicles.

Chapter 5 proposes a quantitative framework for UAV-based relief distribution described by Objective 1. The proposed framework is tested using variations of the objective function to consider ethical principles of distribution equity. In addition, the proposed framework incorporates aspects of energy expenditure and recharge which are UAV-specific, and their impact on the final solution is evaluated in this thesis.

Objective 2 is addressed in Chapters 5 and 7. A bi-level network flow quantitative model is proposed to design the last-mile distribution stage of relief in the aftermath of a natural disaster. The two levels optimise drone operations and humanitarian operations, respectively. The former contains a trajectory optimisation stage that seeks the reduction of UAV battery and flight time, while the humanitarian stage considers the design of the relief distribution system using the trajectories developed in the drone operations as inputs.

The model is formulated as a linear programming model that reduces operation costs, mission duration and energy use. This determines optimal depot locations and vehicle itineraries considering energy allocation and storage, battery management, and distribution fairness.

A stochastic optimisation algorithm is also developed to model the deployment of UAV for road damage assessment to support relief delivery. The ability to to determine infrastructure damages in real-time results in an endogenous stochastic problem as UAV assessment influence routing decisions.

Improvements to the drone operations stage are presented in Chapter 6 as part of Objective

3. An operational sequential trajectory optimisation procedure proposed isolates airspace regions based on their demand level. The approach plans UAV trajectories using a custom Tree algorithm that generates the detailed trajectory at each airspace region given an initial corridor designed by an A* algorithm.

The algorithms in Chapter 6 can also be used to isolate dangerous weather events that prevent the operation of UAV. The effects of weather on the UAV operations are varied. Temperature affects the battery performance, with lower temperatures creating internal resistances and reducing battery capacity. Wind speed and direction is of particular importance as it alters the aerodynamics of the aircraft flight, and given the low mass of UAVs, supposes a risk of accident. The operations outlined in this thesis assume that they are undertaken under the conditions outlined by the manufacturers (Vertical Technologies 2017).

The algorithms and processes developed in this thesis have are validated through rigorous testing in problem instances of different scale and their optimality evaluated against exact solution methods. Finally, the framework has been applied to two real-life case studies: one based on the 1999 Chi-Chi earthquake in Taiwan, and the 2010 Haiti earthquake. In both cases, the large area affected and extensive transport damage delayed road-based relief distribution, making them adequate for UAV-based response. The Chi-Chi network considers 13 demand points, one for each township in the Nanto County, with a total population of 545,874, and 19,483 of vulnerable persons. Data for these case studies were obtained from published studies (Sheu 2007, Government of the Republic of Haiti 2010, USAID 2010) and publicly available material (Vertical Technologies 2017).

The contribution of this thesis to the theory of optimisation is two-fold. Chapter 5 proposes the first solution method to simultaneously solve the location-routing problem considering split delivery, energy management and recharge. The computationally efficient solution method includes a modified route assignment heuristic built upon earlier work by Özdamar (2011) and Dorling et al. (2017). To the best of the author's knowledge, Chapter 5 is the first to define the endogenous stochastic VRP with travel time. Greedy solution technique, while fast is suboptimal and can only handle one UAV agent (due to large scenario generation due to NACs). Here lies an important area of improvement in terms of optimisation, which is the development

of improved optimisation procedures for this type of problem.

This thesis provides a useful tool for humanitarian actors to design and evaluate current humanitarian operations and analyse whether the integration of UAV-related solutions is beneficial to their current practice. The case studies demonstrate the applicability of the proposed approaches in realistic applications, and Chapter 6 presented a generic trajectory management approach for UAV routing useful for other applications.

Industry partners can utilise the frameworks and methodologies developed in this thesis to determine the limiting factors of UAV design and manufacture that affect the operational performance. This allows the industry to develop the key components that provide the greatest marginal performance increase.

Governments can assist humanitarian operations by facilitating access to the required assets, both in-country and inflow of goods (for example, batteries and drone platforms). Furthermore, the proposed methods can demonstrate the potential of the technology, and inform on the appropriate legislation to enable their use in humanitarian contexts.

While current regulations governing UAV operations are flexible and open-ended to allow operations to take place given clearance with the respective Civil Aviation Authority, the speed of humanitarian operations requires clearance to be given within an extremely short time window. The United States' FAA allows clearance for UAV operations in a month which renders the use of UAVs impossible for relief distribution, which occurs within the first few days after a disaster event.

The main problem relates to the prohibition of BVLOS operations and cargo transportation using UAVs. The latter is restricted mainly due to the risk of fall of the item and the hazardous nature of the potential cargo. BVLOS operation regulation will follow ICAO operation-based regulatory framework (see Chapter 3), meaning that the rulings will be based on the risk perceived by the operation. For this reason, the framework presented (particularly Chapter 6) is designed to comply with the RNP frameworks consistent with international regulation (ICAO 2008).

In terms of transferability, the humanitarian response process outlined in this thesis is transferable to any natural disaster provided the conditions are within the UAV specifications.

This means that weather-related disasters such as typhoons and hurricanes would not allow for UAV operations until the high wind speeds have subsided. Similarly, an underlying assumption of this thesis is the permission of UAV use which may be jeopardised in non-natural disasters, where the independence and ethics of humanitarian operations may be questioned by locals. Furthermore, the devised framework is also transferable to other humanitarian operations given some changes. For example, the approach can be used to transport material during the recovery stage, in which case some minor changes to the objective functions may be necessary to reduce the importance of mission time. Having the drone operation comply with RNP allows the framework to be used for other operations given that the appropriate RNP specifications are provided. Therefore, the processes described in this thesis, while tested for humanitarian use, remain applicable for different operations.

To summarise, the contributions of this thesis are as follows:

- Con. 1. Development and implementation of a holistic humanitarian relief distribution framework using UAS for the analysis of humanitarian response. The framework considers unique operational aspects that relate to UAVs, and that distinguish them from conventional humanitarian operations models. This contribution allows practitioners in the humanitarian sector (e.g. WFP, and independent NGOs) to design UAS-based humanitarian response.
- Con. 2. Quantitative assessment of drone-based humanitarian operations. This contribution aids to move the field forward by providing researchers with methodologies to evaluate UAV-based humanitarian relief. The metrics developed were used to measure the impact of UAV-based humanitarian relief framework specified by Contribution 1.
- Con. 3. Development and implementation of novel trajectory management algorithm for UAV fleets. The provision of humanitarian assistance is dependent on the design of trajectories and management of vehicle fleets. This contribution allows practitioners to manage and deploy UAS in humanitarian environments. The algorithm proposed by this contribution focuses on isolating hazardous areas

and ensure minimum safety requirements are met throughout the mission. The assumptions on the latter are informed by Contribution 4.

Con. 4. Conformity with PBN framework (Ren, Castillo-Effen, Yu, Johnson, Yoon, Takuma & Ippolito 2017, Sun et al. 2018). ICAO proposed the PBN that defines the system requirements of aircraft must comply measured in terms of accuracy, integrity, continuity, availability and functionality (ICAO 2008). This framework allows flight restrictions and requirements to be performance-based, thus eliminating sensor-specific routes and procedures. However, PBN is yet to be extended to UAV aircraft, and thus necessitates the provision of new RNP frameworks that correspond to UAV operations. By conforming to the PBN framework, the algorithm can be generalised to consider multiple aircraft types, operations and trajectory profiles.

Contributions 1 and 2 conform the algorithms for UAV-based humanitarian operation design and optimisation, focusing on translating UAV constraints and objectives to the humanitarian context (detailed in Chapter 5 and 7). Contributions 3 and 4 incorporate the trajectory and airspace management constraints, which are a unique requirement of UAV-based operations (Chapter 6).

The presented contributions provide novel methodological approaches in the fields of UAV operations and humanitarian response. The proposed framework serves as a decision support tool for practitioners to manage and design UAV-based humanitarian relief distribution systems. Furthermore, the ability to test a range of problem instances allows the measurement of the impact caused by policy changes on UAV operations, thus informing best policy in terms of quantifiable metrics, such as humanitarian response time, inter-arrival times, among others. The results gathered in this thesis are replicable, as are based on existing problem instances, and data used is facilitated in the respective Chapters.

8.1.1 Future work

This section presents recommendations on further work to address some limitations of the presented thesis, expand upon the conclusions of this research, and interesting topics related to the work undertaken.

- While the development of the proposed frameworks have focused on the application of UAVs for humanitarian relief, its application to other operations in the humanitarian response process (safety monitoring, population control, surveying, etc.) can also be investigated. This requires the modification of the objective functions presented in Chapter 5.
- The trajectory management algorithm presented in Chapter 6 can be expanded to incorporate alternative applications, which requires further experimentation and extensive data. Furthermore, UAV compliance with RNP and PBN is still an active area of research, as the proposition of appropriate RNP and PBN levels depend on the operation type, context, and flight configuration (Jung et al. 2016, Ren, Castillo-Effen, Yu, Yoon, Nakamura, Johnson & Ippolito 2017, Ren, Castillo-Effen, Yu, Johnson, Yoon, Takuma & Ippolito 2017). To achieve this, the performance of the algorithm must be evaluated in terms of accuracy, integrity, continuity and availability.
- The presented thesis assumes that a single coordinator or organisation is responsible for the design and operation of humanitarian response operations. While the assumption is widely adopted in the literature (see Chapter 2) to provide system-optimal results, the actual humanitarian context relies on the cooperation of multiple entities of different sizes, objectives, and capabilities. Further work should delve on developing algorithms that account for these entities, and determine how cooperation, coordination or competition affect the performance of the humanitarian response operation.
- To the best of the author’s knowledge, Chapter 7 presented the first humanitarian application of endogenous stochastic optimisation. The use of UAVs can provide key information at discrete times and locations to inform decision-making during response. Further research should propose extensions to the proposed model that consider other

humanitarian applications including demand estimation and road repair.

Bibliography

- Abdelgawad, H. & Abdulhai, B. (2009), ‘Emergency evacuation planning as a network design problem: a critical review’, *Transportation Letters: International Journal of Transportation Research* **1**(December), 41–58.
- Abdelgawad, H. & Abdulhai, B. (2010), ‘Managing Large-Scale Multimodal Emergency Evacuations’, *National Evacuation Conference* **2**(2), 122–151.
- Abounacer, R., Rekik, M. & Renaud, J. (2014), ‘An exact solution approach for multi-objective location-transportation problem for disaster response’, *Computers and Operations Research* **41**(1), 83–93.
URL: <http://dx.doi.org/10.1016/j.cor.2013.08.001>
- Adeke, P. T., Zava, A. E. & Ato, A. A. (2018), ‘Spot Speed Study of Vehicular Traffic on Major Highways in Makurdi Town’, *Civil and Environmental Research* **10**(6).
- Afrotech EPFL (2016), ‘Red Line’.
URL: <http://afrotech.epfl.ch/page-115280-en.html>
- Ahmadi, M., Seifi, A. & Tootooni, B. (2015), ‘A humanitarian logistics model for disaster relief operation considering network failure and standard relief time: A case study on San Francisco district’, *Transportation Research Part E: Logistics and Transportation Review* **75**, 145–163.
- Akbari, V. & Salman, F. S. (2017), ‘Multi-vehicle synchronized arc routing problem to restore post-disaster network connectivity’, *European Journal of Operational Research* **257**(2), 625–640.
URL: <https://linkinghub.elsevier.com/retrieve/pii/S0377221716305987>

- Aksu, D. T. & Ozdamar, L. (2014), ‘A mathematical model for post-disaster road restoration: Enabling accessibility and evacuation’, *Transportation Research Part E: Logistics and Transportation Review* **61**, 56–67.
- Al Theeb, N. & Murray, C. (2017), ‘Vehicle routing and resource distribution in postdisaster humanitarian relief operations’, *International Transactions in Operational Research* **24**(6), 1253–1284.
- Albareda-Sambola, M., Díaz, J. A. & Fernández, E. (2005), ‘A compact model and tight bounds for a combined location-routing problem’, *Computers and Operations Research* **32**(3), 407–428.
- American Red Cross (2015), Drones for Disaster Response and Relief Operations, Technical Report April, American Red Cross.
- Apap, R. M. (2017), ‘Models and Computational Strategies for Multistage Stochastic Programming under Endogenous and Exogenous Uncertainties’, *ProQuest Dissertations and Theses* p. 143.
URL: <https://search.proquest.com/docview/1953255761?accountid=188395>
- Argollo, S. R., Bandeira, R. A., Mello, L. C. & Campos, V. (2014), ‘Humanitarian supply chain: an analysis of response operations to natural disasters’, *European Journal of Transport and Infrastructure Research* **14**(14), 290–310.
- Arrieta-Camacho, J. J., Biegler, L. T. & Subramanian, D. (2007), ‘Trajectory control of multiple aircraft: An NMPC approach’, *Lecture Notes in Control and Information Sciences* **358**(1), 629–639.
- Aslan, E. & Çelik, M. (2019), ‘Pre-positioning of relief items under road/facility vulnerability with concurrent restoration and relief transportation’, *IIE Transactions* **51**(8), 847–868.
URL: <https://doi.org/10.1080/24725854.2018.1540900>
- Şatir Akpunar, Ö. & Akpınar, e. (2021), ‘A hybrid adaptive large neighbourhood search algorithm for the capacitated location routing problem’, *Expert Systems with Applications* **168**(March 2020).

Bagshaw, S. (2012), Humanitarian Principles, Technical report, UN OCHA, Geneva, Switzerland.

URL: https://www.unocha.org/sites/dms/Documents/OOM-humanitarianprinciples_eng-June12.pdf

Balcik, B., Beamon, B. & Smilowitz, K. (2008), ‘Last Mile Distribution in Humanitarian Relief’, *Journal of Intelligent Transportation Systems* **12**(April), 51–63.

Banomyong, R., Varadejsatitwong, P. & Oloruntoba, R. (2017), ‘A systematic review of humanitarian operations, humanitarian logistics and humanitarian supply chain performance literature 2005 to 2016’, *Annals of Operations Research* pp. 1–16.

Barbarosoğlu, G. & Arda, Y. (2004), ‘A two-stage stochastic programming framework for transportation planning in disaster response’, *Journal of the Operational Research Society* **55**(1), 43–53.

Barbarosoğlu, G., Özdamar, L. & Çevik, A. (2002), ‘An Interactive Approach for Hierarchical Analysis of Helicopter Logistics in Disaster Relief Operations An interactive approach for hierarchical analysis of helicopter logistics in disaster relief operations’, *European Journal of Operational Research* **140**(1), 118–133.

Barbelet, V. (2018), As local as possible, as international as necessary, Technical Report November, Overseas Development Institute, London, UK.

URL: www.odi.org/hpg.

Baronti, F., Fantechi, G., Fanucci, L., Leonardi, E., Roncella, R., Saletti, R. & Saponara, S. (2011), ‘State-of-charge estimation enhancing of lithium batteries through a temperature-dependent cell model’, *International Conference on Applied Electronics (AE)* **0**(1), 1–5.

Barry, J. & Jefferys, A. (2002), A bridge too far: aid agencies and the military in humanitarian response, Technical Report January, Humanitarian Practice Network, London, UK.

Battarra, M., Balcik, B. & Xu, H. (2018), ‘Disaster preparedness using risk-assessment methods

- from earthquake engineering', *European Journal of Operational Research* **269**(2), 423–435.
URL: <https://doi.org/10.1016/j.ejor.2018.02.014>
- BBC News (2016), 'Syria conflict : Aid convoy attack was air strike , UN expert says'.
URL: <http://www.bbc.co.uk/news/world-middle-east-37561755>
- Beamon, B. M. & Balcik, B. (2008), 'Performance measurement in humanitarian relief chains', *International Journal of Public Sector Management* **21**(1), 4–25.
- Bellingham, J., Richards, A. & How, J. P. (2002), 'Receding horizon control of autonomous aerial vehicles', *American Control Conference* **5**, 3741–3746.
- Berger, J. & Barkaoui, M. (2003), 'A new hybrid genetic algorithm for the capacitated vehicle routing problem', *Journal of the Operational Research Society* **54**(12), 1254–1262.
- Betts, J. T. (1998), 'Survey of Numerical Methods for Trajectory Optimization', *Journal of Guidance, Control, and Dynamics* **21**(2), 193–207.
- Bish, D. R., Sherali, H. D. & Hobeika, a. G. (2014), 'Optimal evacuation planning using staging and routing', *Journal of the Operational Research Society* **65**(1), 124–140.
- Bogaitsky, J. (2019), 'Matternet Partners With UPS As It Launches First Regular Drone Delivery Service in U.S.'.
URL: <https://www.forbes.com/sites/jeremybogaitsky/2019/03/26/matternet-partners-with-ups-as-it-sets-to-launch-first-regular-drone-delivery-service-in-us/>
- Bopche, K. S. & Jain, A. (2016), 'A hybrid clustering technique combining a PSO Algorithm with K-Means', *International Journal of Computer Applications* **137**(1), 40–44.
- Boyles, S. D. & Waller, S. T. (2010), 'A mean-variance model for the minimum cost flow problem with stochastic arc costs', *Networks* **56**(3), 215–227.
- Bozorgi-Amiri, A., Jabalameli, M. S., Alinaghian, M. & Heydari, M. (2012), 'A modified particle swarm optimization for disaster relief logistics under uncertain environment', *International Journal of Advanced Manufacturing Technology* **60**(1-4), 357–371.

- Bozorgi-Amiri, A., Jabalameli, M. S. & Mirzapour Al-e-Hashem, S. M. (2013), ‘A multi-objective robust stochastic programming model for disaster relief logistics under uncertainty’, *OR Spectrum* **35**(4), 905–933.
- Bozorgi-Amiri, A. & Khorsi, M. (2016), ‘A dynamic multi-objective location–routing model for relief logistic planning under uncertainty on demand, travel time, and cost parameters’, *International Journal of Advanced Manufacturing Technology* **85**(5-8), 1633–1648.
URL: <http://dx.doi.org/10.1007/s00170-015-7923-3>
- Braekers, K., Ramaekers, K. & Nieuwenhuysen, I. V. (2016), ‘The vehicle routing problem: State of the art classification and review’, *Computers & Industrial Engineering* **99**, 300–313.
URL: <http://dx.doi.org/10.1016/j.cie.2015.12.007>
- Brandt, J. B. & Selig, M. S. (2011), Propeller Performance Data at Low Reynolds Numbers, in ‘49th AIAA Aerospace Sciences Meeting’, Orlando, FL, pp. 1–18.
- Bravo, R. Z. B. & Leiras, A. (2015), Literature Review of the Applications of Uavs in Humanitarian Relief, in ‘XXXV Encontro Nacional de Engenharia de Producao’, Fortaleza, Brasil.
- Bretschneider, S. & Kimms, A. (2012), ‘Pattern-based evacuation planning for urban areas’, *European Journal of Operational Research* **216**(1), 57–69.
- Bruni, M. E., Beraldi, P. & Khodaparasti, S. (2018), ‘A fast heuristic for routing in post-disaster humanitarian relief logistics’, *Transportation Research Procedia* **30**, 304–313.
URL: <https://doi.org/10.1016/j.trpro.2018.09.033>
- Brusco, M. J. & Stahl, S. (2006), *Branch-and-Bound Applications in Combinatorial Data Analysis*, Springer Science & Business Media, New York, NY, United States.
- Burrows, J. (2012), ‘Fuel-optimal aircraft trajectories with fixed arrival times’, *Journal of Guidance, Control, and Dynamics* **6**(1), 14–19.
URL: <http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Fuel-Optimal+Aircraft+Trajectories+with+Fixed+Arrival+Times#0>

- Campbell, L. & Clarke, P. K. (2018), Making Operational Decisions in Humanitarian Response: A Literature Review, Technical report, USAID, London, UK.
URL: <https://www.researchgate.net/publication/325049212>
- Chakrabarty, A., Stepanyan, V., Krishnakumar, K. & Ippolito, C. (2019), Real-time path planning for multi-copters flying in UTM-TCL4, *in* ‘AIAA Scitech 2019 Forum’, San Diego, CA, pp. 1–16.
- Chang, F. S., Wu, J. S., Lee, C. N. & Shen, H. C. (2014), ‘Greedy-search-based multi-objective genetic algorithm for emergency logistics scheduling’, *Expert Systems with Applications* **41**(6), 2947–2956.
- Chen, X. & Zhan, F. (2008), ‘Agent-based modelling and simulation of urban evacuation : relative effectiveness of simultaneous and staged evacuation strategies.’, *Journal of the Operational Research Society* **59**(1), 25–33.
- Chien, S. I. & Korikanthimath, V. V. (2007), ‘Analysis and Modeling of Simultaneous and Staged Emergency Evacuations’, *Journal of Transportation Engineering* **133**(3), 190–197.
- Chiesa, S., Aleina, S. C., Di Meo, G. A., Fusaro, R. & Viola, N. (2014), Autonomous take-off and landing for unmanned aircraft system: Risk and safety analysis, *in* ‘29th Congress of the International Council of the Aeronautical Sciences’, St Petersburg, Russia, pp. 1–10.
- Chiu, Y. C., Zheng, H., Villalobos, J. & Gautam, B. (2007), ‘Modeling no-notice mass evacuation using a dynamic traffic flow optimization model’, *IIE Transactions (Institute of Industrial Engineers)* **39**(1), 83–94.
- Coelho, B. N., Coelho, V. N., Coelho, I. M., Ochi, L. S., K, R. H., Zuidema, D., Lima, M. S. F. & Adilson, R. (2017), ‘A multi-objective green UAV routing problem’, *Computers & Operations Research* **0**, 1–10.
- Cova, T. J. & Johnson, J. P. (2003), ‘A network flow model for lane-based evacuation routing’, *Transportation Research Part A: Policy and Practice* **37**(7), 579–604.

Cozzolino, A. (2012), Humanitarian Logistics and Supply Chain Management, *in* ‘Humanitarian Logistics: Cross-Sector Cooperation in Disaster Relief Management’, Springer-Verlag, Berlin, Germany, pp. 5–17.

URL: <http://link.springer.com/10.1007/978-3-642-30186-5>

CRED (2015), The Human Cost of Natural Disasters 2015: Global Perspective, Technical report, UNISDR, Brussels, Belgium.

CRED (n.d.), The human cost of weather-related disasters 1995-2015, Technical report, UNISDR, Geneva, Switzerland.

URL: https://www.unisdr.org/2015/docs/climatechange/COP21-WeatherDisastersReport_2015_FINAL.pdf, year = 2015

Cura, T. (2012), ‘A particle swarm optimization approach to clustering’, *Expert Systems With Applications* **39**(1), 1582–1588.

URL: <http://dx.doi.org/10.1016/j.eswa.2011.07.123>

CWGER (2008), Guidance on Early Recovery, Technical Report April, United Nation Development Programme Bureau for Crisis Prevention and Recovery, Geneva, Switzerland.

Dai, R. & Cochran, J. E. (2009), ‘Three-dimensional trajectory optimization in constrained airspace’, *Journal of Aircraft* **46**(2), 627–634.

D’Andrea, R. (2014), ‘Guest Editorial. Can Drones Deliver?’, *IEEE Transactions on Automation Science and Engineering* **11**(3), 647–648.

Davey, E., Borton, J. & Foley, M. (2013), A history of the humanitarian system Western origins and foundations.

Davis, L. (1991), *Handbook of genetic algorithms*, Van Nostrand Reinhold, New York, NY.

De Angelis, V., Mecoli, M., Nikoi, C. & Storch, G. (2007), ‘Multiperiod integrated routing and scheduling of World Food Programme cargo planes in Angola’, *Computers and Operations Research* **34**(6 SPEC. ISS.), 1601–1615.

- Delahaye, D., Tsiotras, P., Delahaye, D. & Tsiotras, P. (2014), ‘Mathematical Models for Aircraft Trajectory Design : A Survey’.
- DHL (2015), ‘Successful Trial Integration of DHL Parcelcopter into Logistics Chain’.
URL: http://www.dhl.com/en/press/releases/releases_2016/all/parcel_ecommerce/successful_trial_integration_dhl_parcelcopter_logistics_chain.html
- Dijkstra, E. W. (1959), ‘A note on two problems in connexion with graphs’, *Numerische Mathematik* **1**(1), 269–271.
- Disasters Emergency Committee (2015), ‘Haiti Earthquake Facts and Figures’.
- Dorling, K., Heinrichs, J., Messier, G. G. & Magierowski, S. (2017), ‘Vehicle Routing Problems for Drone Delivery’, *IEEE Transactions on Systems, Man, and Cybernetics: Systems* **47**(1), 70–85.
- Drexl, M. & Schneider, M. (2015), ‘A survey of variants and extensions of the location-routing problem’, *European Journal of Operational Research* **241**(2), 283–308.
URL: <http://dx.doi.org/10.1016/j.ejor.2014.08.030>
- Drone Industry Insights (2017), ‘Drone Energy Sources – Pushing the Boundaries of Electric Flight’.
URL: <https://www.droneii.com/drone-energy-sources>
- DuPlessis, J. (2016), Using drones to inspect post-earthquake road damage in Ecuador, Technical Report 13, FSD.
URL: <https://drones.fsd.ch/en/case-study-no-13-using-drones-to-inspect-post-earthquake-road-damage-in-ecuador/>
- Duque, P. M. & Sörensen, K. (2011), ‘A GRASP metaheuristic to improve accessibility after a disaster’, *OR Spectrum* **33**(3), 525–542.
- EASA (2015), A proposal to create common rules for operating drones in Europe, Technical Report September, EASA, Cologne, Germany.

URL: https://www.easa.europa.eu/system/files/dfu/205933-01-EASA_Summary_of_the_ANPA.pdf

EASA (2017a), Notice of Proposed Amendment 2017-05 (A) - Introduction of a Regulatory Framework for the Operation of Drones, Technical report, EASA.

URL: https://www.easa.europa.eu/sites/default/files/dfu/NPA_2017-05_%28A%29_0.pdf

EASA (2017b), Notice of Proposed Amendment 2017-05 (B) - Introduction of a regulatory framework for the operation of drones, Technical report, EASA.

URL: https://www.easa.europa.eu/system/files/dfu/NPA_2017-05_%28B%29.pdf

EASA (2017c), Notice of Proposed Amendment 2017-10 - Software assurance level requirements for safety assessment of changes to air traffic management / air navigation services functional systems, Technical report, EASA.

URL: <https://www.easa.europa.eu/document-library/notices-of-proposed-amendment/npa-2017-10>

EASA (2018), Opinion No 01/2018 - Introduction of a regulatory framework for the operation of unmanned aircraft system operations in the open and specific categories, Technical report, EASA.

URL: https://www.easa.europa.eu/system/files/dfu/NPA_2017-05_%28B%29.pdf

Eftekhar, M., Li, H., Van Wassenhove, L. N. & Webster, S. (2016), 'The Role of Media Exposure on Coordination in the Humanitarian Setting', *Production and Operations Management* 0(0), 1–15.

EMDAT (2020), 'OFDA/CRED International Disaster Database'.

URL: <http://www.emdat.be/>

Es Yurek, E. & Ozmutlu, H. C. (2018), 'A decomposition-based iterative optimization algorithm for traveling salesman problem with drone', *Transportation Research Part C: Emerging Technologies* 91(July 2017), 249–262.

URL: <https://doi.org/10.1016/j.trc.2018.04.009>

Escribano Macias, J., Angeloudis, P. & Ochieng, W. (2020), ‘Optimal Hub Selection for Rapid Medical Deliveries using Unmanned Aerial Vehicles’, *Transportation Research Part C: Emerging Technologies* **110**(January), 56–80.

URL: http://www.optimization-online.org/DB_FILE/2018/02/6459.pdf

Esmin, A. A., Coelho, R. A. & Matwin, S. (2015), ‘A review on particle swarm optimization algorithm and its variants to clustering high-dimensional data’, *Artificial Intelligence Review* **44**(1), 23–45.

URL: <http://dx.doi.org/10.1007/s10462-013-9400-4>

ETC (2017), ‘Humanitarian Unmanned Aerial Vehicles (UAV) Coordination Model’.

URL: <https://www.etcluster.org/discussions/humanitarian-unmanned-aerial-vehicles-uav-coordination-model>

European Parliament and Council (2018), ‘Regulation (EU) 2018/1139’.

URL: <https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32018R1139&from=EN>

Even, C., Pillac, V. & Hentenryck, P. V. (2015), Convergent Plans for Large-Scale Evacuations, in ‘Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI-15)’, pp. 1121–1127.

FAA (2012), FAR Part 333, Technical report, FAA.

FAA (2015), Concept of Operations, Technical report, FAA, Washington, DC.

FAA (2016), ‘NextGen implementation plan 2016’.

URL: https://www.faa.gov/nextgen/media/NextGen_Implementation_Plan-2016.pdf

FAA (2018), Integration of Civil Unmanned Aircraft Systems (UAS) in the National Airspace System (NAS) Roadmap Federal Aviation Administration Second Edition, Technical report, FAA.

URL: https://www.faa.gov/uas/resources/policy-library/media/Second_Edition_Integration_of_Civil_UAS_NAS_Roadmap_July_2018.pdf

- Forsmo, E. J. (2012), Optimal Path Planning for Unmanned Aerial Systems, Msc thesis, Norwegian University of Science and Technology Norway.
- Fortune, S. (1987), 'A Sweepline Algorithm for Voronoi Diagrams', *Algorithmica* **2**(2), 153–174.
- Friesz, T., Bernstein, D., R., T. & B., W. (1993), 'A variational inequality formulation of the dynamic network user equilibrium problem', *Operations Research* **41**(1), 80–91.
- FSD (2016), Using drones to create maps and assess building damage in Ecuador, Technical Report 14, FSD, Geneva, Switzerland.
- URL:** <http://drones.fsd.ch/wp-content/uploads/2016/11/14.Case-StudyEcuador.pdf>
- Fujii, K., Higuchi, K. & Rekimoto, J. (2013), Endless flyer: A continuous flying drone with automatic battery replacement, in 'Proceedings - IEEE 10th International Conference on Ubiquitous Intelligence and Computing, UIC 2013 and IEEE 10th International Conference on Autonomic and Trusted Computing, ATC 2013', pp. 216–223.
- Furini, F., Persiani, C. A. & Toth, P. (2016), 'The Time Dependent Traveling Salesman Planning Problem in Controlled Airspace', *Transportation Research Part B: Methodological* **90**, 38–55.
- Gammell, J. D., Srinivasa, S. S. & Barfoot, T. D. (2014), 'Informed RRT*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic', *IEEE International Conference on Intelligent Robots and Systems* pp. 2997–3004.
- Gao, X.-Z., Hou, Z.-X., Guo, Z., Chen, X.-Q. & Chen, X.-Q. (2014), 'Joint optimization of battery mass and flight trajectory for high-altitude solar-powered aircraft', *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* **228**(13), 2439–2451.
- Garrido, R. A., Lamas, P. & Pino, F. J. (2015), 'A stochastic programming approach for floods emergency logistics', *Transportation Research Part E: Logistics and Transportation Review* **75**, 18–31.
- Geiger, B. (2009), Unmanned aerial vehicle trajectory planning with direct methods, Doctoral thesis, Pennsylvania State University.

- Gendreau, M., Ghiani, G. & Guerriero, E. (2015), ‘Time-dependent routing problems: A review’, *Computers and Operations Research* **64**, 189–197.
- Gendreau, M. & Potvin, J.-Y. (2010), *Handbook of Metaheuristics*, 2nd edn, Springer Publishing Company, Incorporated.
- Golabi, M., Shavarani, S. M. & Izbirak, G. (2017), ‘An edge-based stochastic facility location problem in UAV-supported humanitarian relief logistics: a case study of Tehran earthquake’, *Natural Hazards* **87**(3), 1545–1565.
- Goldblatt, R. B. (2004), Evacuation Planning, Human Factors and Traffic Engineering Perspectives, in ‘Proceedings of the 2004 European Transport Conference’, pp. 1–21.
- Gottlieb, I. (1997), *Practical Electric Motor Handbook*, 1 edn, Reed Educational and Professional Publishing Ltd, Oxford, United Kingdom.
- Government of the Republic of Haiti (2010), Haiti Earthquake PDNA: Assessment of damage, losses, general and sectoral needs.
- Gracey, W. (1980), Measurement of Aircraft Speed and Altitude, Technical report, NASA, Hampton, VA.
- URL:** <http://www.dtic.mil/dtic/tr/fulltext/u2/a280006.pdf>
- Grafham, O. & Lahn, G. (2018), ‘The Costs of Fuelling Humanitarian Aid’.
- Griva, I., Nash, S. G. & Sofer, A. (2009), *Linear and Nonlinear Optimization*, 2 edn, Society for Industrial and Applied Mathematics.
- Grossmann, C. & Roos, H.-G. (2007), *Numerical Treatment of Partial Differential Equations*, 3 edn, Springer, Berlin.
- Guezouli, L. & Abdelhamid, S. (2017), ‘A multi-objective optimization of Multi-depot Fleet Size and Mix Vehicle Routing Problem with time window’, *2017 6th International Conference on Systems and Control, ICSC 2017* pp. 328–333.

- Gupta, V. & Grossmann, I. E. (2011), ‘Solution strategies for multistage stochastic programming with endogenous uncertainties’, *Computers and Chemical Engineering* **35**(11), 2235–2247.
- Ha, Q. M., Deville, Y., Pham, Q. D. & Hà, M. H. (2018), ‘On the min-cost Traveling Salesman Problem with Drone’, *Transportation Research Part C: Emerging Technologies* **86**(May 2016), 597–621.
URL: <https://doi.org/10.1016/j.trc.2017.11.015>
- Haghani, A. & Oh, S. C. (1996), ‘Formulation and solution of a multi-commodity, multi-modal network flow model for disaster relief operations’, *Transportation Research Part A: Policy and Practice* **30**(3), 231–250.
- Haidari, L. A., Brown, S. T., Ferguson, M., Bancroft, E., Spiker, M., Wilcox, A., Ambikapathi, R., Sampath, V., Connor, D. L. & Lee, B. Y. (2016), ‘The economic and operational value of using drones to transport vaccines’, *Vaccine* **34**(34), 4062–4067.
- Hakimi, L. (1964), ‘Optimum Locations of Switching Centers and the Absolute Centers and Medians of a Graph’, *Operations Research* **12**(3), 450–459.
URL: <https://www.jstor.org/stable/pdf/168125.pdf>
- Hall, N. (2016), ‘The Drag Coefficient’.
URL: <https://www.grc.nasa.gov/www/k-12/airplane/dragco.html>
- Ham, A. M. (2018), ‘Integrated scheduling of m-truck, m-drone, and m-depot constrained by time-window, drop-pickup, and m-visit using constraint programming’, *Transportation Research Part C: Emerging Technologies* **91**(July 2017), 1–14.
URL: <https://doi.org/10.1016/j.trc.2018.03.025>
- Han, K., Friesz, T. L. & Yao, T. (2013), ‘Existence of simultaneous route and departure choice dynamic user equilibrium’, *Transportation Research Part B: Methodological* **53**, 17–30.
- Hart, P. E., Nilsson, N. J. & Raphael, B. (1968), ‘Formal Basis for the Heuristic Determination of Minimum Cost Paths’, *IEEE Transactions on Systems Science and Cybernetics* (2), 100–107.

- Hart, W. E., Laird, C., Watson, J.-P. & Woodruff, D. L. (2012), *Pyomo – Optimization Modeling in Python*, Vol. 67, Springer Science & Business Media.
- Hart, W. E., Watson, J. P. & Woodruff, D. L. (2011), ‘Pyomo: Modeling and solving mathematical programs in Python’, *Mathematical Programming Computation* **3**(3), 219–260.
- Hassan, R., Cohanin, B. & de Weck, O. (2005), ‘A comparison of particle swarm optimization and the genetic algorithm’, *1st AIAA multidisciplinary design optimization specialist conference* pp. 1–13.
URL: <http://arc.aiaa.org/doi/pdf/10.2514/6.2005-1897>
- Hidalgo, J. (2019), ‘Drone delivery closer as Reno-based Flirtey OK ’ d to fly beyond line of sight’.
URL: <https://eu.rgj.com/story/money/business/2019/03/08/drone-delivery-closer-reno-flirtey-approved-fly-beyond-line-sight/3097046002/>
- Hildmann, H. & Kovacs, E. (2019), ‘Review: Using Unmanned Aerial Vehicles (UAVs) as Mobile Sensing Platforms (MSPs) for Disaster Response, Civil Security and Public Safety’, *Drones* **3**(3), 59.
- Hobeika, A. G. & Kim, C. (1998), ‘Comparison of traffic assignments in evacuation modeling’, *IEEE Transactions on Engineering Management* **45**(2), 192–198.
- Holden, J. & Goel, N. (2016), Fast-Forwarding to a Future of On-Demand Urban Air Transportation, Technical report, Uber.
URL: <https://www.uber.com/elevate.pdf>
- Holguín-Veras, J., Jaller, M., Van Wassenhove, L. N., Pérez, N. & Wachtendorf, T. (2012), ‘On the unique features of post-disaster humanitarian logistics’, *Journal of Operations Management* **30**(7-8), 494–506.
URL: <http://dx.doi.org/10.1016/j.jom.2012.08.003>
- Holguin-Veras, J., Pérez, N., Jaller, M., Van Wassenhove, L. N. & Aros-Vera, F. (2013), ‘On the

- appropriate objective function for post-disaster humanitarian logistics models', *Journal of Operations Management* **31**, 262–280.
- Homola, J., Dao, Q., Martin, L., Mercer, J., Mohlenbrink, C. & Claudatos, L. (2017), Technical capability level 2 unmanned aircraft system traffic management (UTM) flight demonstration: Description and analysis, in 'AIAA/IEEE Digital Avionics Systems Conference - Proceedings'.
- Hong, I., Kuby, M. & Murray, A. T. (2018), 'A range-restricted recharging station coverage model for drone delivery service planning', *Transportation Research Part C: Emerging Technologies* **90**(February), 198–212.
URL: <https://doi.org/10.1016/j.trc.2018.02.017>
- Hooshmand, F. & MirHassani, S. A. (2016), 'Efficient constraint reduction in multistage stochastic programming problems with endogenous uncertainty', *Optimization Methods and Software* **31**(2), 359–376.
URL: <https://doi.org/10.1080/10556788.2015.1088850>
- Hooshmand, F., MirHassani, S. A. & Akhavein, A. (2018), 'Adapting GA to solve a novel model for operating room scheduling problem with endogenous uncertainty', *Operations Research for Health Care* .
- Hosseini, S. & Mesbahi, M. (2013), 'Optimal path planning and power allocation for a long endurance solar-powered UAV', *2013 American Control Conference* pp. 2588–2593.
- Hrabar, S. (2008), 3D path planning and stereo-based obstacle avoidance for rotorcraft UAVs, in '2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS', IEEE, pp. 807–814.
- Hu, F., Xu, W. & Li, X. (2012), 'A modified particle swarm optimization algorithm for optimal allocation of earthquake emergency shelters', *International Journal of Geographical Information Science* **26**(9), 1643–1666.
- Hu, Z. H., Sheu, J. B., Yin, Y. Q. & Wei, C. (2017), 'Post-disaster relief operations considering psychological costs of waiting for evacuation and relief resources', *Transportmetrica A: Trans-*

port Science **13**(2), 108–138.

URL: <https://doi.org/10.1080/23249935.2016.1222559>

Huang, J. S. & Lien, Y. N. (2012), ‘Challenges of emergency communication network for disaster response’, *2012 IEEE International Conference on Communication Systems, ICCS 2012* pp. 528–532.

ICAO (1944), ‘Convention on International Civil Aviation (Chicago Convention)’.

URL: <https://www.refworld.org/docid/3ddca0dd4.html>

ICAO (2008), *Performance Based Navigation (PBN) Manual*, 3rd edn, ICAO, Montreal, Canada.

ICAO (2009), ‘Required Navigation Performance Authorization Required (RNP AR) Procedure Design Manual’.

ICAO (2019a), ‘Task Force on Unmanned Aircraft Systems for Humanitarian Aid and Development (TF-UHAD)’.

URL: [https://www.icao.int/safety/UA/Pages/Task-Force-on-Unmanned-Aircraft-Systems-for-Humanitarian-Aid-and-Development-\(TF-UHAD\).aspx](https://www.icao.int/safety/UA/Pages/Task-Force-on-Unmanned-Aircraft-Systems-for-Humanitarian-Aid-and-Development-(TF-UHAD).aspx)

ICAO (2019b), ‘Unmanned Aircraft Systems Traffic Management (UTM) – A Common Framework with Core Principles for Global Harmonization’.

URL: <https://www.icao.int/safety/UA/Documents/UTM-Framework.en.alltext.pdf>

Jonsbraten, T. W. (1998), *Optimization Models for Petroleum Field Exploitation*, PhD thesis, Norwegian School of Economics and Business Administration.

Jung, J., D’Souza, S. N., Johnson, M. A., Ishihara, A. K., Modi, H. C., Nikaido, B. & Hasseeb, H. (2016), Applying Required Navigation Performance Concept for Traffic Management of Small Unmanned Aircraft Systems, *in* ‘Proceedings of the 30th Congress of the International Council of the Aeronautical Sciences (ICAS2016)’, Daejeon, Korea.

Karakatič, S. & Podgorelec, V. (2015), ‘A survey of genetic algorithms for solving multi depot vehicle routing problem’, *Applied Soft Computing Journal* **27**, 519–532.

- Keach, S. (2018), ‘Amazon-style drones could be delivering packages to UK homes by 2019’.
URL: <https://www.thesun.co.uk/tech/5736054/amazon-drone-deliveries-uk-prime-air/>
- Khaligh, F. H. & Mirhassani, S. A. (2016), ‘A mathematical model for vehicle routing problem under endogenous uncertainty’, *International Journal of Production Research* **54**(2), 579–590.
URL: <http://dx.doi.org/10.1080/00207543.2015.1057625>
- Khan, H., Vasilescu, L. G. & Khan, A. (2008), ‘Disaster Management Cycle – a Theoretical Approach’, *Management & Marketing* **6**(1), 43–50.
- Kim, S., Shin, Y., Lee, G. M. & Moon, I. (2018), ‘Network repair crew scheduling for short-term disasters’, *Applied Mathematical Modelling* **64**, 510–523.
URL: <https://doi.org/10.1016/j.apm.2018.07.047>
- Kimms, A. & Maiwald, M. (2017), ‘An exact network flow formulation for cell-based evacuation in urban areas’, *Naval Research Logistics* **64**(7), 547–555.
- Kimms, A. & Maiwald, M. (2018), ‘Bi-objective safe and resilient urban evacuation planning’, *European Journal of Operational Research* **269**(3), 1122–1136.
URL: <https://doi.org/10.1016/j.ejor.2018.02.050>
- Klose, A. & Drexler, A. (2005), ‘Facility location models for distribution system design’, *European Journal of Operational Research* **162**, 4–29.
- Kok, A. L., Hans, E. W. & Schutten, J. M. (2012), ‘Vehicle routing under time-dependent travel times: The impact of congestion avoidance’, *Computers and Operations Research* **39**(5), 910–918.
URL: <http://dx.doi.org/10.1016/j.cor.2011.05.027>
- Kopardekar, P., Rios, J., Prevot, T., Johnson, M., Jung, J. & Robinson III, J. E. (2016), UAS Traffic Management (UTM) Concept of Operations to Safely Enable Low Altitude Flight Operations, in ‘16th AIAA Aviation Technology, Integration, and Operations Conference’, pp. 1–16.
URL: <http://arc.aiaa.org/doi/10.2514/6.2016-3292>

- Kothari, M., Postlethwaite, I. & Gu, D. W. (2009), ‘Multi-UAV path planning in obstacle rich environments using rapidly-exploring random trees’, *Proceedings of the IEEE Conference on Decision and Control* pp. 3069–3074.
- Kovács, G. & Spens, K. (2009), ‘Identifying challenges in humanitarian logistics’, *International Journal of Physical Distribution and Logistics Management* **39**(6), 506–528.
- Kovács, G. & Spens, K. M. (2007), ‘Humanitarian logistics in disaster relief operations’, *International Journal of Physical Distribution & Logistics Management* **37**(2), 99–114.
- Kryvasheyev, Y., Chen, H., Obradovich, N., Moro, E., Van Hentenryck, P., Fowler, J. & Cebrian, M. (2016), ‘Rapid assessment of disaster damage using social media activity’, *Science Advances* **2**(3), 1–12.
- Kung, H. Y., Chen, C. H. & Ku, H. H. (2012), ‘Designing intelligent disaster prediction models and systems for debris-flow disasters in Taiwan’, *Expert Systems with Applications* **39**(5), 5838–5856.
URL: <http://dx.doi.org/10.1016/j.eswa.2011.11.083>
- Kuwata, Y. & How, J. P. (2011), ‘Cooperative distributed robust trajectory optimization using receding horizon MILP’, *IEEE Transactions on Control Systems Technology* **19**(2), 423–431.
- Langelaan, J. W. (2007), ‘Long Distance / Duration Trajectory Optimization for Small UAVs’, *AIAA Guidance, Navigation and Control Conference* pp. 1–14.
- Laporte, G., Mercure, H. & Nobert, Y. (1986), ‘An Exact Algorithm for the Asymmetrical Capacitated Vehicle Routing Problem’, *Networks* **16**(1 986), 33–46.
- LaValle, S. M. (1998), Rapidly-Exploring Random Trees: A New Tool for Path Planning, Report no. tr 98-11, Iowa State University.
URL: <http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Rapidly-exploring+random+trees:+A+new+tool+for+path+planning#0>
- Lavalle, S. M. (2006), Sampling-based Motion Planning, in ‘Planning Algorithms’, Cambridge University Press, chapter 5, pp. 185–248.

Lewis, J. & Magalhaes, L. (2018), ‘Here Come Helicopters on Demand’.

URL: <https://www.wsj.com/articles/here-come-helicopters-on-demand-1498010700>

Li, Q., Wei, C., Wu, J. & Zhu, X. (2014), Improved PRM method of low altitude penetration trajectory planning for UAVs, *in* ‘2014 IEEE Chinese Guidance, Navigation and Control Conference, CGNCC 2014’, pp. 2651–2656.

Li, Q., Yang, X. & Heng, W. (2006), Integrating Traffic Simulation Models with Evacuation Planning System in a GIS Environment, *in* ‘2006 IEEE Intelligent Transportation Systems Conference’, Toronto, Canada, pp. 590–595.

Li, S. & Teo, K. L. (2018), ‘Post-disaster multi-period road network repair: work scheduling and relief logistics optimization’, *Annals of Operations Research* .

URL: <http://link.springer.com/10.1007/s10479-018-3037-2>

Li, X., Huang, B., Liu, Z., Zhang, X. & Sun, J. (2012), ‘A novel method for planning a staged evacuation’, *Journal of Systems Science and Complexity* **25**(6), 1093–1107.

Li, X., Li, Q. & Claramunt, C. (2018), ‘A time-extended network model for staged evacuation planning’, *Safety Science* **108**(May 2016), 225–236.

URL: <https://doi.org/10.1016/j.ssci.2017.08.004>

Li, X., Xie, J., Cai, M., Xie, M. & Wang, Z. (2009), ‘Path planning for UAV based on improved heuristic A* algorithm’, *ICEMI 2009 - Proceedings of 9th International Conference on Electronic Measurement and Instruments* **3**, 3488–3493.

Liang, N. J., Shih, Y. T., Shih, F. Y., Wu, H. M., Wang, H. J., Shi, S. F., Liu, M. Y. & Wang, B. B. (2001), ‘Disaster epidemiology and medical response in the Chi-Chi earthquake in Taiwan’, *Annals of Emergency Medicine* **38**(5), 549–555.

Lin, Y. & Saripalli, S. (2014), ‘Path planning using 3D Dubins Curve for Unmanned Aerial Vehicles’, *2014 International Conference on Unmanned Aircraft Systems (ICUAS)* pp. 296–304.

URL: <http://ieeexplore.ieee.org/document/6842268/>

- Lin, Y. & Saripalli, S. (2017), ‘Sampling-Based Path Planning for UAV Collision Avoidance’, *IEEE Transactions on Intelligent Transportation Systems* **18**(11), 3179–3192.
- Linden, D. (1984), *Handbook of Batteries and Fuel Cells*, McGraw-Hill Book Co., New York, NY, United States.
- Liu, Y., Lai, X. & Chang, G.-l. (2006), ‘Cell-Based Network Optimization Model for Staged Evacuation Planning’, *Transportation Research Record: Journal of the Transportation Research Board* **1964**, 127–135.
- Lu, Q., George, B. & Shekhar, S. (2005), Capacity Constrained Routing Algorithms for Evacuation Planning: A Summary of Results, in B. E. Bauzer Medeiros C., Egenhofer M.J., ed., ‘Advances in spatial and temporal databases. SSTD 2005. Lecture Notes in Computer Science’, Vol. 3633, Springer-Verlag Berlin Heidelberg, pp. 291–307.
- Mas, E., Suppasri, A., Imamura, F. & Koshimura, S. (2012), ‘Agent-based Simulation of the 2011 Great East Japan Earthquake / Tsunami Evacuation : An Integrated Model of Tsunami Inundation and Evacuation’, *Journal of Natural Disaster Science* **34**(1), 41–57.
- Meier, P. (2018), ‘Field Testing Medical Cargo Drones in the DR’.
URL: <https://werobotics.org/blog/2018/02/27/testing-cargo-drones-dr/>
- Meier, P. & Bergelund, J. (2017), Field-Testing the First Cargo Drone Deliveries in the Amazon Rainforest, Technical Report February, WeRobotics, Lima, Peru.
URL: <http://werobotics.org/wp-content/uploads/2017/02/WeRobotics-Amazon-Rainforest-Cargo-Drones-Report.pdf>
- Meier, P. & Soesilo, D. (2014), Using Drones for Medical Payload Delivery in Papua New Guinea, Technical report, FSD - Swiss Foundation for Mine Action, Geneva, Switzerland.
URL: <http://drones.fsd.ch/wp-content/uploads/2016/04/Case-Study-No2-PapuaNewGuinea.pdf>
- Meier, P., Soesilo, D. & Guerin, D. (2016), Cargo Drones in Humanitarian Contexts Meeting Summary, Technical report, FSD - Swiss Foundation for Mine Action, Sheffield, UK.

URL: <http://drones.fsd.ch/wp-content/uploads/2016/08/CargoDrones-MeetingSummaryfinal.pdf>

Mete, H. O. & Zabinsky, Z. B. (2010), ‘Stochastic optimization of medical supply location and distribution in disaster management’, *International Journal of Production Economics* **126**(1), 76–84.

URL: <http://dx.doi.org/10.1016/j.ijpe.2009.10.004>

Michini, B., Toksoz, T., Redding, J., Michini, M., How, J., Vavrina, M. & Vian, J. (2011), Automated Battery Swap and Recharge to Enable Persistent UAV Missions, in ‘Infotech@Aerospace 2011’, American Institute of Aeronautics and Astronautics.

Mitchell, J. T., Cutter, S. L. & Edmonds, A. S. (2007), ‘Improving Shadow Evacuation Management: Case Study of the Grianteville, South Carolina, chlorine spill’, *Journal of Emergency Management* **5**(1), 28–34.

Mitchell, S. W. & Radwan, E. (2006), Heuristic Prioritization of Emergency Evacuations Staging to Reduce Clearance Time, in ‘Proceedings of the 85th Annual Meeting Transportation Research Board’, Washington, DC, USA.

Moon, M. (2018), ‘Dubai tests a passenger drone for its flying taxi service’.

URL: <https://www.engadget.com/2017/09/26/dubai-volocopter-passenger-drone-test/>

Morrison, D. D., Riley, J. D. & Zancanaro, J. F. (1962), ‘Multiple shooting method for two-point boundary value problems’, *Communications of the ACM* **5**(12), 613–614.

Murray, C. C. & Chu, A. G. (2015), ‘The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery’, *Transportation Research Part C: Emerging Technologies* **54**, 86–109.

Nagel, K. & Flötteröd, G. (2012), Agent-Based Traffic Assignment: Going From Trips to Behavioral Travelers, in R. M. Pendyala & C. R. Bhat, eds, ‘Travel Behaviour Research in an Evolving World’, International Association for Travel Behaviour Research (IATBR), pp. 261–293.

- Naidu, B. (2019), ‘Spot Speed Survey & Analysis – A Case Study on Jalandhar-Ludhiana Road, National Highway-1, India’, *International Journal of Engineering and Technology* **10**(1).
- Nasiri, M. M., Shahmoradi Moghadam, H. & Torabi, S. A. (2018), ‘A hub covering flow network design resilient to major disruptions and supply/demand uncertainties’, *International Journal of Business Continuity and Risk Management* **8**(4), 319.
- Nedjati, A., Vizvari, B. & Izbirak, G. (2015), ‘Post-earthquake response by small UAV helicopters’, *Natural Hazards* **80**, 1669–1688.
- Nguyen, D. T., Ofli, F., Imran, M. & Mitra, P. (2017), Damage assessment from social media imagery data during disasters, in ‘Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2017’, pp. 569–576.
- Nohadani, O. & Sharma, K. (2008), ‘Optimization under uncertainty’, *Springer Optimization and Its Applications* **22**, 125–178.
- Noyan, N. (2012), ‘Risk-averse two-stage stochastic programming with an application to disaster management’, *Computers and Operations Research* **39**(3), 541–559.
- OHCHR (2019), ‘The role of OHCHR’.
- URL:** <https://www.ohchr.org/EN/Issues/HIV/Pages/RoleOHCHR.aspx>
- Ostler, J. N., Bowman, W. J., Snyder, D. O. & McLain, T. W. (2009), ‘Performance Flight Testing of Small, Electric Powered Unmanned Aerial Vehicles’, *International Journal of Micro Air Vehicles* **1**(3), 155–171.
- Ousingsawat, J. & Campbell, M. E. (2004), Establishing Trajectories for Multi-Vehicle Reconnaissance, in ‘AIAA Guidance, Navigation, and Control Conference and Exhibit’, Providence, RI, USA, pp. 1–12.
- URL:** <http://arc.aiaa.org/doi/pdf/10.2514/6.2004-5224>
- Özdamar, L. (2011), ‘Planning helicopter logistics in disaster relief’, *OR Spectrum* **33**(3), 655–672.

- Ozdamar, L., Aksu, D. T., Yasa, E. & Ergunes, B. (2018), ‘Disaster relief routing in limited capacity road networks with heterogeneous flows’, *Journal of Industrial and Management Optimization* **14**(4), 1367–1380.
- Özdamar, L., Ekinici, E. & Küçükyazici, B. (2004), ‘Emergency Logistics Planning in Natural Disasters’, *Annals of Operations Research* **129**(1-4), 217–245.
- Özdamar, L. & Ertem, M. A. (2015), ‘Models, solutions and enabling technologies in humanitarian logistics’, *European Journal of Operational Research* **244**(1), 55–65.
- Özdamar, L., Tüzün Aksu, D. & Ergünes, B. (2014), ‘Coordinating debris cleanup operations in post disaster road networks’, *Socio-Economic Planning Sciences* **48**(4), 249–262.
- Pacheco, G. G. & Batta, R. (2016), ‘Forecast-driven model for prepositioning supplies in preparation for a foreseen hurricane’, *Journal of the Operational Research Society* **67**(1), 98–113.
- PAHO & WHO (2001), Humanitarian Supply Management and Logistics in the Health Sector, Technical report, Pan American Health Organization, Washington, DC.
- Pannequin, J., Bayen, A., Mitchell, I., Chung, H. & Sastry, S. (2007), Multiple Aircraft Deconflicted Path Planning with Weather Avoidance Constraints, in ‘AIAA Guidance, Navigation and Control Conference and Exhibit’, Hilton Head, SC, USA, pp. 1–16.
- Park, M. W. & Kim, Y. D. (1998), ‘A systematic procedure for setting parameters in simulated annealing algorithms’, *Computers and Operations Research* **25**(3), 207–217.
- Park, S. (2015), Air force institute of technology, PhD thesis, Air Force Institute of Technology.
- Paterlini, S. & Krink, T. (2006), ‘Differential evolution and particle swarm optimisation in partitional clustering’, *Computational Statistics and Data Analysis* **50**(5), 1220–1247.
- Paulmann, J. (2013), ‘Conjunctures in the History of International Humanitarian Aid during the Twentieth Century’, *Humanity: An International Journal of Human Rights, Humanitarianism, and Development* **4**(2), 215–238.

- Perl, J. & Daskin, M. S. (1985), ‘A warehouse location-routing problem’, *Transportation Research Part B* **19**(5), 381–396.
- Pharpatara, P., Herisse, B. & Bestaoui, Y. (2017), ‘3-D Trajectory Planning of Aerial Vehicles Using RRT ’, *IEEE Transactions on Control Systems Technology* **25**(3), 1116–1123.
- Pillac, V., Van Hentenryck, P. & Even, C. (2016), ‘A conflict-based path-generation heuristic for evacuation planning’, *Transportation Research Part B: Methodological* **83**, 136–150.
- Pisinger, D. & Ropke, S. (2010), Large neighborhood search, in M. Gendreau & J.-Y. Potvin, eds, ‘Handbook of Metaheuristics’, 2 edn, Vol. 146, Springer, Boston, MA, pp. 399–420.
- Poli, R., Kennedy, J. & Blackwell, T. (2007), ‘Particle swarm optimization’, *Swarm Intelligence* **1**(1), 33–57.
URL: <http://link.springer.com/10.1007/s11721-007-0002-0>
- Ponda, S. (2008), Trajectory Optimization for Target Localization Using Small Unmanned Aerial Vehicles, in ‘AIAA Guidance, Navigation, and Control Conference’, Chicago, IL, USA.
URL: <http://arc.aiaa.org/doi/pdf/10.2514/6.2009-6015%5C>
- Puri, A. (2005), ‘A Survey of Unmanned Aerial Vehicles (UAV) for Traffic Surveillance’.
- QuestUAV (2017), ‘QuestUAV Q-Pod Technology’.
URL: https://www.geosystems.fr/images/PDF/20140613_Brochure_QPod_complete.pdf
- Raghunathan, A. U., Gopal, V., Subramanian, D., Biegler, L. T. & Samad, T. (2004), ‘Dynamic optimization strategies for three-dimensional conflict resolution of multiple aircraft’, *Journal of Guidance, Control, and Dynamics* **27**(4), 586–594.
- Ransikarbum, K. & Mason, S. J. (2016), ‘Goal programming-based post-disaster decision making for integrated relief distribution and early-stage network restoration’, *International Journal of Production Economics* **182**, 324–341.
- Raptopoulos, A. (2013), ‘No roads? There is a drone for that’.
URL: https://www.ted.com/talks/andreas_raptopoulos_no_roads_there_s_a_drone_for_that

- Rawls, C. G. & Turnquist, M. A. (2010), 'Pre-positioning of emergency supplies for disaster response', *Transportation Research Part B: Methodological* **44**(4), 521–534.
URL: <http://dx.doi.org/10.1016/j.trb.2009.08.003>
- Rawls, C. G. & Turnquist, M. A. (2011), 'Pre-positioning planning for emergency response with service quality constraints', *OR Spectrum* **33**(3), 481–498.
- Rawls, C. G. & Turnquist, M. A. (2012), 'Pre-positioning and dynamic delivery planning for short-term response following a natural disaster', *Socio-Economic Planning Sciences* **46**(1), 46–54.
- Reagan, J. (2018), 'Drone Delivery Canada Teams Up with Toyota Subsidiary'.
URL: <https://dronelife.com/2018/05/29/drone-delivery-canada-teams-up-with-toyota-subsidiary/>
- Ren, L., Castillo-Effen, M., Yu, H., Johnson, E., Yoon, Y., Takuma, N. & Ippolito, C. A. (2017), 'Small unmanned aircraft system (sUAS) categorization framework for low altitude traffic services', *AIAA/IEEE Digital Avionics Systems Conference - Proceedings* **2017-Septe.**
- Ren, L., Castillo-Effen, M., Yu, H., Yoon, Y., Nakamura, T., Johnson, E. N. & Ippolito, C. A. (2017), Small Unmanned Aircraft System (sUAS) Trajectory Modeling in Support of UAS Traffic Management (UTM), in '17th AIAA Aviation Technology, Integration, and Operations Conference', Denver, CO, USA.
URL: <https://arc.aiaa.org/doi/10.2514/6.2017-4268>
- Rennemo, S. J., Rø, K. F., Hvattum, L. M. & Tirado, G. (2014), 'A three-stage stochastic facility routing model for disaster response planning', *Transportation Research Part E: Logistics and Transportation Review* **62**, 116–135.
URL: <http://dx.doi.org/10.1016/j.tre.2013.12.006>
- Rezaei-Malek, M., Tavakkoli-Moghaddam, R., Zahiri, B. & Bozorgi-Amiri, A. (2016), 'An interactive approach for designing a robust disaster relief logistics network with perishable commodities', *Computers and Industrial Engineering* **94**, 201–215.
URL: <http://dx.doi.org/10.1016/j.cie.2016.01.014>

- Richards, A. & How, J. (2002), ‘Aircraft trajectory planning with collision avoidance using mixed integer linear programming’, *American Control Conference* **3**(2), 1936–1941 vol.3.
URL: <http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=1023918>
- Rios, J. L., Martin, L. & Mercer, J. (2019), Use of UAS Reports (UREPs) during TCL3 Field Testing, Technical report, NASA Langley Research Center, Hampton, VA.
- Rios, J. L., Mulfinger, D. G., Smith, I. S., Smith, D. R., Baskaran, V. & Wang, L. (2017), UTM Data Working Group Demonstration 1 Final Report, Technical report, NASA, Moffett Field, CAA, USA.
URL: <https://utm.arc.nasa.gov/docs/Rios-NASA-Tech-Memo-2017-219494v2.pdf>
- Robinson, P. (2000), ‘The Policy-Media Interaction Model: Measuring Media Power During Humanitarian Crisis’, *Journal of Peace Research* **37**(5), 613–633.
- Rochon, E., Grogan, J., Cooper, D. & Avant, E. (2018), Submitting Operational Waiver Requests, Technical report, FAA, Baltimore, United States.
URL: https://www.faa.gov/uas/resources/event_archive/2018_uas_symposium/media/Submitting-Operational-Waiver-Requests.pdf
- Rosenkrantz, D. J., Stearns, R. E. & Lewis II, P. M. (2009), An analysis of several heuristics for the traveling salesman problem, in S. S. Ravi & S. K. Shukla, eds, ‘Fundamental Problems in Computing’, Springer Science & Business Media, Dordrecht.
- Rothfeld, R. L. (2018), Agent-based Simulation of Urban Air Mobility, in ‘AIAA AVIATION Forum. 2018 MOdeling and Simulation Technologies Conference’, Atlanta Georgia, pp. 1–10.
- Ruiz, S. & Soler, M. (2015), ‘Conflict pattern analysis under the consideration of optimal trajectories in the European ATM - seminar’, *ATM R&D Seminar (ATM 2015)* **15**, 1–10.
- Rysaback-Smith, H. (2015), ‘History and principles of humanitarian action’, *Turkish Journal of Emergency Medicine* **15**(Suppl 1), 5–7.
URL: <http://dx.doi.org/10.5505/1304.7361.2015.52207>

Sahin, H., Kara, B. Y. & Karasan, O. E. (2016), ‘Debris removal during disaster response: A case for Turkey’, *Socio-Economic Planning Sciences* **53**, 49–59.

URL: <http://dx.doi.org/10.1016/j.seps.2015.10.003>

Sarıççek, I. & Akkus, Y. (2013), ‘Unmanned Aerial Vehicle hub-location and routing for monitoring geographic borders’, *Applied Mathematical Modelling* **39**, 3939–3953.

Sathiyaraj, B. M., Jain, L. C., Finn, A. & Drake, S. (2008), ‘Multiple UAVs path planning algorithms: A comparative study’, *Fuzzy Optimization and Decision Making* **7**(3), 257–267.

Sbayti, H. & Mahmassani, H. (2006), ‘Optimal Scheduling of Evacuation Operations’, *Transportation Research Record* **1964**(1964), 238–246.

Scott, J. E. & Scott, C. H. (2017), Delivery Models for Healthcare, in ‘Proceedings of the 50th Hawaii International Conference on System Sciences’, pp. 3297–3304.

Seekircher, K. (2017), Evacuation Planning under Selfish Evacuation Routing, Doctorate thesis, Duisburg-Essen University.

Sensefly (2014), ‘User Manual eBee and eBee Ag’.

URL: [http://95.110.228.56/documentUAV/drone manual/%5BENG%5D-2014_Extended_User_Manual_eBee_and_eBee_Ag-v12.1.pdf](http://95.110.228.56/documentUAV/drone%20manual/%5BENG%5D-2014_Extended_User_Manual_eBee_and_eBee_Ag-v12.1.pdf)

Seo, S. H., Choi, J. I. & Song, J. (2017), ‘Secure utilization of beacons and UAVs in emergency response systems for building fire hazard’, *Sensors (Switzerland)* **17**(10).

Sergeeva, M. (2017), Automated Airspace Sectorization by Genetic Algorithm, PhD thesis, Université Paul Sabatier.

SESAR (2016), European Drones Outlook Study, Technical report, SESAR, Brussels, Belgium.

URL: http://www.sesarju.eu/sites/default/files/documents/reports/European_Drones_Outlook_Study_2016.pdf

SESAR (2019), Intermediate Concept of Operations for U-Space, Technical report, Eurocontrol, Rome, Italy.

- Shavarani, S. M. (2019), 'Multi-level facility location-allocation problem for post-disaster humanitarian relief distribution: A case study', *Journal of Humanitarian Logistics and Supply Chain Management* **9**(1), 70–81.
- Shavarani, S. M., Nejad, M. G., Rismanchian, F. & Izbirak, G. (2018), 'Application of hierarchical facility location problem for optimization of a drone delivery system: a case study of Amazon prime air in the city of San Francisco', *International Journal of Advanced Manufacturing Technology* **95**(9-12), 3141–3153.
- Shaw, P. (1998), Using constraint programming and local search methods to solve vehicle routing problems, in 'Lecture Notes in Computer Science volume 1520', chapter Fourth Int, pp. 417–431.
- Sheu, J. B. (2007), 'An emergency logistics distribution approach for quick response to urgent relief demand in disasters', *Transportation Research Part E: Logistics and Transportation Review* **43**(6), 687–709.
- Sheu, J. B. (2010), 'Dynamic relief-demand management for emergency logistics operations under large-scale disasters', *Transportation Research Part E: Logistics and Transportation Review* **46**(1), 1–17.
- Shippey, B. & Subbarao, K. (2008), 'Trajectory design using collocation and genetic algorithms: Aircraft turning maneuver', *Proceedings of the IEEE International Conference on Control Applications* pp. 905–911.
- Simmons, D. (2016), 'Rwanda begins Zipline commercial drone deliveries'.
URL: <http://www.bbc.co.uk/news/technology-37646474>
- Soesilo, D. & Sandvik, K. B. (2016), Drones in Humanitarian Action – A survey on perceptions and applications, Technical report, FSD, Geneva, Switzerland.
URL: [https://reliefweb.int/sites/reliefweb.int/files/resources/Drones in Humanitarian Action Survey-Analysis-FINAL2.pdf](https://reliefweb.int/sites/reliefweb.int/files/resources/Drones%20in%20Humanitarian%20Action%20Survey-Analysis-FINAL2.pdf)

- Song, B. D., Park, K. & Kim, J. (2018), 'Persistent UAV delivery logistics: MILP formulation and efficient heuristic', *Computers and Industrial Engineering* **120**(May), 418–428.
- Song, Y. & Yan, X. (2016), 'A method for formulizing disaster evacuation demand curves based on SI model', *International Journal of Environmental Research and Public Health* **13**(10).
- Sorkin, A. R. (2018), 'Larry Page s Flying Taxis, Now Exiting Stealth Mode'.
URL: <https://www.nytimes.com/2018/03/12/business/dealbook/flying-taxis-larry-page.html>
- Steets, J., Grünewald, F., Binder, A., De Geoffroy, V., Kauffmann, D., Krüger, S., Meier, C. & Sokpoh, B. (2010), Cluster Approach Evaluation 2 Synthesis Report, Technical Report April, Global Public Policy Institute, Berlin, Germany.
URL: [https://www.humanitarianresponse.info/system/files/documents/files/Cluster Approach Evaluation 2.pdf](https://www.humanitarianresponse.info/system/files/documents/files/Cluster%20Approach%20Evaluation%20.pdf)
- Stirk, C. (2014), Humanitarian Assistance from Non-State Donors: What is it worth?, Technical Report April, Global Humanitarian Assistance, Bristol, UK.
URL: <http://www.globalhumanitarianassistance.org/report/humanitarian-assistance-non-state-donors>
- Stöcker, C., Bennett, R., Nex, F., Gerke, M. & Zevenbergen, J. (2017), 'Review of the current state of UAV regulations', *Remote Sensing* **9**(5), 33–35.
- Stoney, R. B. (1993), Design, fabrication and test of a vertical attitude takeoff and landing unmanned air vehicle., Master thesis, Naval Postgraduate School, Monterrey, CA, United States.
- Sumaylo, K. K. (2017), Local humanitarian action in practice, Technical report, Humanitarian Leadership Academy, London, UK.
- Sun, R., Zhang, Y., Ye, B. & Ochieng, W. (2018), A Required Navigation Performance Based Approach to Monitor the Accuracy and Integrity Performance of UAVs for Delivery Applications, in G. S. Sun J., Yang C., ed., 'China Satellite Navigation Conference (CSNC) 2018 Proceedings. CSNC 2018. Lecture Notes in Electrical Engineering, vol 497.', Springer,

Singapore.

URL: <http://link.springer.com/10.1007/978-981-13-7759-4>

Suzuki, K. A., Kemper Filho, P. & Morrison, J. R. (2012), ‘Automatic battery replacement system for UAVs: Analysis and design’, *Journal of Intelligent and Robotic Systems: Theory and Applications* **65**(1-4), 563–586.

Szeto, W. Y. & Lo, H. K. (2004), ‘A cell-based simultaneous route and departure time choice model with elastic demand’, *Transportation Research Part B: Methodological* **38**(7), 593–612.

Tang, J., Alam, S., Lokan, C. & Abbass, H. A. (2012), ‘A multi-objective approach for Dynamic Airspace Sectorization using agent based and geometric models’, *Transportation Research Part C: Emerging Technologies* **21**(1), 89–121.

URL: <http://dx.doi.org/10.1016/j.trc.2011.08.008>

Tatham, P. (2009), ‘An Investigation into the Suitability of the Use of Unmanned Aerial Vehicle Systems (UAVs) to Support the Initial Needs Assessment Process in Rapid Onset Humanitarian Disasters’, *International Journal of Risk Assessment and Management* **13**(1), 1–20.

Tatham, P. (2011), Improving the Civil-Military Dimension of Disaster-Related Humanitarian Logistics, Technical report, Griffith Business School.

Thippavong, D. P., Apaza, R., Barmore, B., Battiste, V., Belcastro, C. M., Burian, B., Dao, Q., Feary, M., Go, S., Goodrich, K. H., Homola, J., Idris, H. R., Kopardekar, P. H., Lachter, J. B., Neogi, N. A., Ng, H. K., Oseguera-Lohr, R. M., Patterson, M. D. & Verma, S. A. (2018), ‘Urban Air Mobility Airspace Integration Concepts and Considerations’, *2018 Aviation Technology, Integration, and Operations Conference* .

URL: <https://arc.aiaa.org/doi/10.2514/6.2018-3676>

Thomas, A. & Kopczak, L. (2007), Life-Saving Supply Chains, in H. L. Lee & C.-Y. Lee, eds, ‘Building Supply Chain Excellence in Emerging Economies. International Series in Operations Research & Management Science, vol 98.’, Springer, Boston, MA, USA, pp. 93–111.

- Thomas, A. S. & Kopczak, L. R. (2005), From Logistics to Supply Chain Management: The Path Forward in the Humanitarian Sector, Technical report, Fritz Institute, San Francisco, CA.
- Timperio, G., Panchal, G. B., Samvedi, A., Goh, M. & De Souza, R. (2017), ‘Decision support framework for location selection and disaster relief network design’, *Journal of Humanitarian Logistics and Supply Chain Management* **7**(3), 222–245.
- Tofighi, S., Torabi, S. A. & Mansouri, S. A. (2016), ‘Humanitarian logistics network design under mixed uncertainty’, *European Journal of Operational Research* **250**(1), 239–250.
URL: <http://dx.doi.org/10.1016/j.ejor.2015.08.059>
- Tomasini, R. & Van Wassenhove, L. (2009), *Humanitarian Logistics*, INSEAD Business Press, Palgrave Macmillan UK.
URL: <https://books.google.co.uk/books?id=eeEXTdQHKDgC>
- Trelea, I. C. (2003), ‘The particle swarm optimization algorithm: Convergence analysis and parameter selection’, *Information Processing Letters* **85**(6), 317–325.
- UAV Systems International (2017), ‘X8 Long Range Cargo Drone’.
URL: <http://www.uavsystemsinternational.com/product/x8-long-range-cargo-drone/>
- UAViators (2015), ‘Humanitarian UAV Code of Conduct & Guidelines’.
URL: <https://humanitariandronecode.files.wordpress.com/2017/12/uaviators-code-and-guidelines.pdf>
- UK Government (2014), ‘Military low flying’.
URL: <https://www.gov.uk/guidance/military-low-flying>
- Ülkü, M. A., Bell, K. M. & Wilson, S. G. (2015), ‘Modeling the impact of donor behavior on humanitarian aid operations’, *Annals of Operations Research* **230**(1), 153–168.
URL: <http://dx.doi.org/10.1007/s10479-014-1623-5>

UN General Assembly (2015), 'Transforming our world : the 2030 Agenda for Sustainable Development'.

URL: <https://www.refworld.org/docid/57b6e3e44.html>

UNICEF (2015), UNICEF Strategic Framework on Environmental Sustainability for Children, Technical report, Division of Data, Research and Policy, New York, NY.

URL: https://www.unicef.org/environment/files/Framework_on_ES_Dec15.pdf

UNICEF (2016), 'Malawi tests first unmanned aerial vehicle flights for HIV early infant diagnosis'.

URL: https://www.unicef.org/media/media_90462.html?p=printme

Unifly (2019), 'Unifly's Unmanned Traffic Management Platform'.

URL: <https://www.unifly.aero/en/unifly-utm>

UNISDR (2009), Terminology on Disaster Risk Reduction, Technical report, UNISDR, Geneva, Switzerland.

URL: www.unisdr.org/publications

United Nations (1949), 'Geneva Convention Relative To the Protection of Civilian Persons'.

United Nations (1977), 'Protocol II to the Geneva Conventions: Protection of Victims of Non-International armed Conflicts'.

UNOCHA (2012), OCHA on Message : General Assembly resolution 46 / 182, Technical report, UN OCHA.

UNOCHA (2015), 'Country-based Pooled Funds'.

URL: www.reliefweb.int

UNOCHA (2019a), CERF 2018 Annual Report, Technical report, UNOCHA, Geneva, Switzerland.

UNOCHA (2019b), 'International Coordination Mechanisms'.

URL: <http://interactive.unocha.org/publication/asiadisasterresponse/InternationalHumanitarianArchitecture.html>

UNOCHA (2019c), ‘Who does what ?’.

Urbina, E. & Wolshon, B. (2003), ‘National review of hurricane evacuation plans and policies: A comparison and contrast of state practices’, *Transportation Research Part A: Policy and Practice* **37**(3), 257–275.

Urmson, C. & Simmons, R. (2003), Approaches for heuristically biasing RRT growth, in ‘Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)’, Vol. 2, pp. 1178–1183.

URL: <http://ieeexplore.ieee.org/document/1248805/>

USAID (2010), USG Humanitarian Assistance To Haiti for the Earthquake, Technical report, USAID.

USAID (2016), Ecuador - Earthquake: Fact sheet 5, Technical report, USAID.

URL: https://www.usaid.gov/sites/default/files/documents/1866/ecuador_eq_fs05_05-06-2016.pdf

Vahdani, B., Veysmoradi, D., Noori, F. & Mansour, F. (2018a), ‘Two-stage multi-objective location-routing-inventory model for humanitarian logistics network design under uncertainty’, *International Journal of Disaster Risk Reduction* **27**(October 2017), 290–306.

URL: <https://doi.org/10.1016/j.ijdrr.2017.10.015>

Vahdani, B., Veysmoradi, D., Noori, F. & Mansour, F. (2018b), ‘Two-stage multi-objective location-routing-inventory model for humanitarian logistics network design under uncertainty’, *International Journal of Disaster Risk Reduction* **27**(October 2017), 290–306.

URL: <https://doi.org/10.1016/j.ijdrr.2017.10.015>

Van Hentenryck, P., Bent, R. & Coffrin, C. (2010), ‘Strategic planning for disaster recovery with stochastic last mile distribution’, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **6140 LNCS**, 318–333.

- Van Wassenhove, L. N. (2006), ‘Blackett memorial lecture humanitarian aid logistics: Supply chain management in high gear’, *Journal of the Operational Research Society* **57**(5), 475–489.
- Van Wassenhove, L. N. & Pedraza Martinez, A. J. (2012), ‘Using OR to adapt supply chain management best practices to humanitarian logistics’, *International Transactions in Operational Research* **19**(1-2), 307–322.
- Vascik, P. D., Balakrishnan, H. & Hansman, R. J. (2018), Assessment of Air Traffic Control for Urban Air Mobility and Unmanned Systems, *in* ‘8th International Conference on Research in Air Transportation’, Barcelona, Spain.
- Vayanos, P., Kuhn, D. & Rustem, B. (2011), ‘Decision rules for information discovery in multi-stage stochastic programming’, *Proceedings of the IEEE Conference on Decision and Control* pp. 7368–7373.
- Vecere, A., Monteiro, R., Ammann, W. J., Giovinazzi, S. & Melo Santos, R. H. (2017), ‘Predictive models for post disaster shelter needs assessment’, *International Journal of Disaster Risk Reduction* **21**(November 2016), 44–62.
URL: <http://dx.doi.org/10.1016/j.ijdr.2016.11.010>
- Vertical Technologies (2017), ‘DeltaQuad Specifications’.
URL: https://www.verticaltechnologies.com/img/cms/DeltaQuad_EU_specs_V2.pdf
- Victoria, J. F., Afsar, H. M. & Prins, C. (2016), Vehicle Routing Problem with Time-Dependent Demand in humanitarian logistics, *in* ‘Proceedings of 2015 International Conference on Industrial Engineering and Systems Management, IEEE IESM 2015’, Seville, Spain, pp. 686–694.
- Vornic, A. (2017), ‘Drones take flight to help end hunger’.
URL: <https://insight.wfp.org/drones-take-flight-to-help-end-hunger-20222948e01>
- Weimin, D. (2000), Chi-Chi, Taiwan Earthquake Event Report, Technical report, Risk Management Solutions, Menlo Park, CA.
URL: <http://forms2.rms.com/rs/729-DJX-565/images/eq-chi-chi-taiwan.eq.pdf>

WFP (2002), Emergency Field Operations Pocketbook, Technical report, WFP, Rome, Italy.

URL: http://www.unicef.org/emerg/files/WFP_manual.pdf

WFP (2017a), ‘The WFP food basket’.

URL: <https://www.wfp.org/food-assistance/kind-food-assistance/wfp-food-basket>

WFP (2017b), The Year in Review 2016, Technical report, WFP, Rome, Italy.

URL: https://docs.wfp.org/api/documents/WFP-0000019183/download/?_ga=2.245347461.1698952705.1512920441-1272578713.1499352257

WFP (2018a), ‘72 Hours Emergency Assessment’.

URL: <https://www.wfp.org/72-hours-emergency-assessment>

WFP (2018b), ‘Immediate Response Account’.

URL: <https://www.wfp.org/funding/immediate-response-account>

WFP (2018c), The 72-hour Assessment Approach: A guide for vulnerability and spatial analysis in sudden-onset disasters, Technical Report June, WFP, Rome, Italy.

WFP (2019), ‘Logistics Cluster’.

URL: <https://www.wfp.org/logistics-cluster>

WFP Aviation (2012), Annual Review 2011, Technical report, WFP Aviation, Rome, Italy.

URL: http://documents.wfp.org/stellent/groups/public/documents/communications/wfp_247741.pdf?_ga=2.8805693.940050920.1519661951-2078292071.1519661951

WFP Aviation (2016), WFP Aviation in 2015, Technical report, World Food Programme, Rome, Italy.

URL: http://documents.wfp.org/stellent/groups/public/documents/communications/wfp_284702.pdf?_ga=2.101328514.2089474537.1519309844-1504303459.1517938603

Wisetjindawat, W., Ito, H., Fujita, M. & Eizo, H. (2014), ‘Planning Disaster Relief Operations’, *Procedia - Social and Behavioral Sciences* **125**, 412–421.

URL: <http://dx.doi.org/10.1016/j.sbspro.2014.01.1484>

- Wolfson, S. & Wright, N. (1995), A UNHCR Handbook for the Military on Humanitarian Operations, Technical Report January, UNHCR, Geneva, Switzerland.
- Wolshon, B. (2001), ‘“One-Way-Out”: Contraflow Freeway Operation for Hurricane Evacuation’, *Natural Hazards Review* **2**(3), 105–112.
- Wu, T. H., Low, C. & Bai, J. W. (2002), ‘Heuristic solutions to multi-depot location-routing problems’, *Computers and Operations Research* **29**(10), 1393–1415.
- Wuerl, A., Crain, T. & Braden, E. (2003), ‘Genetic algorithm and calculus of variations-based trajectory optimization technique’, *Journal of Spacecraft and Rockets* **40**(6), 882–888.
- X Development (2018), ‘X - Wing’.
- Xie, C., Lin, D. Y. & Travis Waller, S. (2010), ‘A dynamic evacuation network optimization problem with lane reversal and crossing elimination strategies’, *Transportation Research Part E: Logistics and Transportation Review* **46**(3), 295–316.
- Xie, J., Li, Q., Wan, Q. & Li, X. (2014), ‘Near optimal allocation strategy for making a staged evacuation plan with multiple exits’, *Annals of GIS* **20**(3), 159–168.
URL: <http://dx.doi.org/10.1080/19475683.2014.942363>
- Xiong, M., Zeng, D., Yao, H. & Li, Y. (2016), ‘A crowd simulation based uav control architecture for industrial disaster evacuation’, *IEEE Vehicular Technology Conference* **2016-July**, 1–5.
- Xue, M. (2009), ‘Airspace Sector Redesign Based on Voronoi Diagrams’, *Journal of Aerospace Computing Information and Communication* **6**(12), 624–634.
- Yakici, E. (2016a), ‘Generalization of a UAV Location and Routing Problem by Time Windows’, *Journal of Naval Sciences and Engineering* **12**(2), 67–78.
- Yakici, E. (2016b), ‘Solving location and routing problem for UAVs’, *Computers and Industrial Engineering* **102**, 294–301.

- Yan, F., Liu, Y. S. & Xiao, J. Z. (2013), ‘Path planning in complex 3D environments using a probabilistic roadmap method’, *International Journal of Automation and Computing* **10**(6), 525–533.
- Yan, S. & Shih, Y. L. (2009), ‘Optimal scheduling of emergency roadway repair and subsequent relief distribution’, *Computers and Operations Research* **36**(6), 2049–2065.
- Yang, L., Qi, J., Xiao, J. & Yong, X. (2015), A literature review of UAV 3D path planning, in ‘Proceedings of the World Congress on Intelligent Control and Automation (WCICA)’, IEEE, pp. 2376–2381.
- Yang, X., Ding, M., Zhou, C., Cai, C., Yu, Q. & Shao, S. (2009), ‘Fast on-ship route planning using improved sparse A-star algorithm for UAVs’, *Proceedings of SPIE* **7497**, 749705–749705–8.
URL: <http://link.aip.org/link/PSISDG/v7497/i1/p749705/s1?Agg=doi>
- Yi, W. & Kumar, A. (2007), ‘Ant colony optimization for disaster relief operations’, *Transportation Research Part E: Logistics and Transportation Review* **43**(6), 660–672.
- Yi, W. & Özdamar, L. (2007), ‘A dynamic logistics coordination model for evacuation and support in disaster response activities’, *European Journal of Operational Research* **179**(3), 1177–1193.
- Yılmaz, O., Yakıcı, E. & Karatas, M. (2018), ‘A UAV location and routing problem with spatio-temporal synchronization constraints solved by ant colony optimization’, *Journal of Heuristics* **25**(4), 673–701.
URL: <https://doi.org/10.1007/s10732-018-9389-6>
- Yokoyama, N. & Suzuki, S. (2005), ‘Modified Genetic Algorithm for Constrained Trajectory Optimization’, *Journal of Guidance, Control, and Dynamics* **28**(1), 139–144.
URL: <http://arc.aiaa.org/doi/abs/10.2514/1.3042>
- Zhang, W., Wang, N. & Nicholson, C. (2017), ‘Resilience-based post-disaster recovery strategies for road-bridge networks’, *Structure and Infrastructure Engineering* **13**(11), 1404–1413.
URL: <http://dx.doi.org/10.1080/15732479.2016.1271813>

- Zhou, M. (2013), ‘Trajectory Optimization for Fuel Cell Powered Uavs Trajectory Optimization for’.
- Zhou, X. & Taylor, J. (2014), ‘DTAlite: A queue-based mesoscopic traffic simulator for fast model evaluation and calibration’, *Cogent Engineering* **1**(1), 1–19.
- Zhu, B.-J., Hou, Z.-X. & Ouyang, H.-J. (2016), ‘Trajectory optimization of unmanned aerial vehicle in dynamic soaring’, *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* **0**(0), 1–15.
URL: <http://pig.sagepub.com/lookup/doi/10.1177/0954410016664912>
- Ziliaskopoulos, A. K. (2000), ‘A Linear Programming Model for the Single Destination System Optimum Dynamic Traffic Assignment Problem’, *Transportation Science* **34**(1), 37–49.

Appendix A

Humanitarian Operations Literature Table

Table A.1: Summary of literature reviewed in Section 2. Symbols used in this table are defined in Table A.2.

Author	Solution	Application	Objectives	Features	Size
Hub Location Problem					
Balcik et al. (2008)	MILP	Relief	S	A,B,CD,MS	167 nodes
Bozorgi-Amiri et al. (2012)	PSO	Relief	C	A,CD,MC,STC,STD,	35 nodes
Bozorgi-Amiri et al. (2013)	MILP	Relief	C,S	A,CD,MC,RO,STC,STD	15 nodes
Golabi et al. (2017)	GA	Evacuation	D	A,AC,CD,LR	33 nodes
Hong et al. (2018)	MILP	Delivery	S	LR,RC	10 nodes
Hu et al. (2012)	PSO,SA	Evacuation	C	A,CD	18 nodes
Hu et al. (2017)	GA	Evacuation	C,S,SF	A,CD,MT	28 nodes
Nasiri et al. (2018)	MILP	Relief	C	CD,A,B,MS,RO	10 nodes

Table A.1 – continued from previous page

Author	Solution	Application	Objectives	Features	Size
Noyan (2012)	2SSP	Relief	C,S,T	A,CD,MC,STN	50 nodes
Rawls & Turnquist (2010)	OH	Relief	C,S	A,CD,MC,STD	30 nodes
Rawls & Turnquist (2011)	MILP	Relief	C,S	A,CD,MS,STD	30 nodes
Rawls & Turnquist (2012)	MILP	Relief	C,S	A,CD,MC,MS,MT,STD	50 nodes
Scott & Scott (2017)	MILP	Healthcare	DT	B,LR	9 nodes
Shavarani (2019)	GA	Delivery	D	A,LR,RC	182 nodes
Tofighi et al. (2016)	2SSP	Relief	C,T,S	A,STD,STT	9 nodes
Vehicle Routing Problem					
Akbari & Salman (2017)	OH	Repair	T,R	AC,CV,LB,SC	250 nodes
Barbarosoğlu et al. (2002)	OH	Relief	C,T	CV,MC,PD,SD	6 nodes
Coelho et al. (2017)	GA	Delivery	D,E,T	CV,LR,RC	33 nodes
De Angelis et al. (2007)	MIP	Relief	D	CV,TW	8 nodes
Dorling et al. (2017)	MIP,SA	Delivery	D	B,LR,PE,RC	500 nodes
Duque & Sörensen (2011)	VNS	Repair	T,R	CD,CV,SC	216 nodes
Garrido et al. (2015)	2SSP	Relief	C,S,T	STN,CV,AC	10 nodes
Haidari et al. (2016)	MIP	Vaccine	C,S	CV,LR	123 nodes
Kim et al. (2018)	AC	Repair	R,T	AC,CV,LD,SC	23 nodes
Li & Teo (2018)	GA	Relief/Repair	C,DT,T	AC,MT	1,000 nodes
Murray & Chu (2015)	MIP,NN	Delivery	T	CV,LR	9 nodes
Nedjati et al. (2015)	MIP	Relief	M	CV,MS	44 nodes
Özdamar (2011)	MIP,OH	Relief	T	CV,LS,MC,PD,RC,SD	68 nodes

Table A.1 – continued from previous page

Author	Solution	Application	Objectives	Features	Size
Özdamar et al. (2004)	LR	Relief	U	CV,LS,MC,SD	12 nodes
Pacheco & Batta (2016)	OSO	Relief	C,S,T	A,CV,F,STD	455 nodes
Ransikarbum & Mason (2016)	GP	Relief/Repair	C,DT,T	PR,SC	8,057 nodes
Sahin et al. (2016)	OH	Repair	R,T	AC,CV,LB,SC	45 nodes
Sheu (2007)	MIP	Relief	C,D	STD	6 nodes
Sheu (2010)	MIP	Relief	S	STD	24 nodes
Song et al. (2018)	MIP	Healthcare	D,S	CV,LR,PE,RC,SC	10 nodes
Aksu & Özdamar (2014)	MIP	Repair	R,T	AC,CV,LB,SC	386 nodes
Yan & Shih (2009)	MIP	Relief/Repair	DT,T	AC,MC,SC	30,579 nodes
Location Routing Problem					
Abounacer et al. (2014)	MILP	Relief	C,DT,S	CV,MC,MS	49 nodes
Bozorgi-Amiri & Khorsi (2016)	MILP	Relief	C,DT,U	CV,CD,MC	6 nodes
Mete & Zabinsky (2010)	2SSP	Relief	C,S,T	CD,CV,STD	15 nodes
Rezaei-Malek et al. (2016)	MILP	Relief	C,U	CD,MT,RO,STD,STT	19 nodes
Sarıççek & Akkus (2013)	MILP	Surveillance	C,SF	LR,MS	34 nodes
Vahdani et al. (2018 <i>b</i>)	GA	Relief	C	CD,MT,SD,STD	50 nodes
Van Hentenryck et al. (2010)	2SSP	Relief	C,S,T	A,CV,STD	1,000 nodes
Yakici (2016 <i>a</i>)	MILP	Military	V	CV,LR,TW	60 nodes
Yakici (2016 <i>b</i>)	OH	Military	V	CV,LR	150 nodes
Yi & Özdamar (2007)	MILP	Relief	U	CV,PD	20 nodes
Yılmaz et al. (2018)	OH	Reconnaissance	C	CV,LR	80 nodes

Table A.1 – continued from previous page

Author	Solution	Application	Objectives	Features	Size
Endogenous Stochastic Problems					
Boyles & Waller (2010)	MSP	Min Cost Flow	T	STT	5,000 nodes
Hooshmand et al. (2018)	GA	Healthcare	T	SC,STO	-
Khaligh & Mirhassani (2016)	MSP	Delivery	C	CV,STD	10 nodes
Nohadani & Sharma (2008)	SO	Shortest Path	D	STT	300 nodes
Vayanos et al. (2011)	MSP	Offshore	C	STO	-
Evacuation					
Bish et al. (2014)	SG	Generic	NC,S	CG,DM,HE,SO	Mesoscopic
Bretschneider & Kimms (2012)	OH	Generic	NC	CG,DM	Macroscopic
Chen & Zhan (2008)	SG	Generic	T	CG,DM,NG,SO,	Microscopic
Chien & Korikanthimath (2007)	SG	Generic	T	CG,DL,DM,SO	Mesoscopic
Chiu et al. (2007)	SG	No-Notice	T	CG,DM,SO	Mesoscopic
Even et al. (2015)	MILP	Generic	NC	CG,DM,PG	Macroscopic
Hobeika & Kim (1998)	SG	Power Plant	E	CG,LN,UE	Macroscopic
Kimms & Maiwald (2017)	MILP	Generic	SF	CF,CG,DM	Mesoscopic
Kimms & Maiwald (2018)	MILP	Generic	SF	CG,DM,RE	Mesoscopic
Li et al. (2012)	SP,OH	Generic	T	CG,DM,SO	Macroscopic
Li et al. (2018)	OH	Generic	T,W	A,CG,DM	Macroscopic
Liu et al. (2006)	SG	Generic	T,W	CG,DM,SO	Mesoscopic
Mitchell & Radwan (2006)	OH	Hurricane	NC	CG,DM,SO	Microscopic
Pillac et al. (2016)	OH,SA	Generic	E	CG,DM,PG	Macroscopic

Table A.1 – continued from previous page

Author	Solution	Application	Objectives	Features	Size
Shayti & Mahmassani (2006)	SG	Generic	T	CG,DM,SO	Mesoscopic
Seekircher (2017)	SG	Generic	NC	CG,DM,LB	Mesoscopic
Xie et al. (2014)	OH	Generic	NC	A,CG,DM,SO	Macroscopic

Table A.2: Symbol Glossary.

Solution		Objectives		Features	
Symbol	Description	Symbol	Description	Symbol	Description
AC	Ant Colony	C	Cost	A	Demand Allocation
GA	Genetic Algorithm	D	Distance Travelled	AC	Accessibility
GP	Goal Programming	DT	Delivery Time	B	Budget
LR	Lagrangian Relaxation	E	Total Evacuations	CD	Capacitated Depots
MIP	Mixed Integer Programming	M	Mission number	CF	Contra-flow
NN	Neural Networks	NC	Network Clearance	CG	Congestion
OH	Own Heuristic	S	Satisfied Demand	CV	Capacitated Vehicles
OSO	Other Stochastic Optimisation	SF	Safety	DL	Delay
PSO	Particle Swarm Optimisation	R	Repair	DM	Demand Management
SA	Simulated Annealing	T	Operation Time	HE	Heterogenous Agents
SG	Stackelberg Game	U	Unsatisfied Demand	LB	Link Blockage
SP	Shortest Path Algorithm	V	Value	LN	Link Number
VNS	Variable Neighbourhood Search	W	Wait Time	LR	Limited Vehicle Range

Table A.2 – continued from previous page

Symbol	Description	Symbol	Description	Symbol	Description
2SSP	2-Stage Stochastic Optimisation	-	-	LS	Limited Supply
Features					
MC	Multi-Commodity	MS	Minimum Service	MT	Multiple Time Steps
PG	Path Generation	PR	Partial Repair	RC	Refuel/Recharge
RE	Resilience	RO	Robustness	SC	Schedulling
SD	Split Delivery	SO	System-Optimal Assignment	STD	Stochastic Demand
STN	Stochastic Network State	STT	Stochastic Travel Time	TW	Time Windows
STO	Stochastic Operating Time	UE	Multi-Commodity	ZC	Zone Clearance

Appendix B

Drone Operations Literature Table

Table B.1: Summary of literature reviewed in Section 3. Symbols used in this table are defined in Table B.2.

Author	Solution	Application	Objectives	Features	Size
Arrieta-Camacho et al. (2007)	DC,IPOPT	Control	M	WP,3DM,DT	2 aircraft
Bellingham et al. (2002)	MILP	Avoidance	T	2DM,RH,SO	-
Chakrabarty et al. (2019)	RKT	Generic	M	3DM,SO,DO	500 m flight
Forsmo (2012)	MILP,GA	Generic	T	DT,2/3DM,SO	2 km flight
Gammell et al. (2014)	RRT	Generic	D	2DM,SO	-
Gao et al. (2014)	GSM,PSO	Design	A,T	2DM,GL	24 hour flight
Geiger (2009)	DC,NN	Surveillance	T	2DM,DT,W	-
Hosseini & Mesbahi (2013)	NLP	Generic	E	3DM,VR	24 hour flight
Hrabar (2008)	PRM,DL	Generic	D	2DM,SO	50 m flight

Table B.1 – continued from previous page

Author	Solution	Application	Objectives	Features	Size
Kothari et al. (2009)	RRT	Generic	D	2DM,SO	1 km flight
Kuwata & How (2011)	MILP	Cooperation	T	2DM,RH	2 vehicles
Langelan (2007)	DC	Soaring	E, T	2DM,DS	60 km flight
Li et al. (2014)	PRM,A*	Military	D	3DM,SO	600 m flight
Lin & Saripalli (2014)	RRT	Generic	D	3DM,DB,DO	1 km flight
Lin & Saripalli (2017)	RRT	Generic	D	3DM,SO,DO	1 km flight
Pannequin et al. (2007)	DC,NPSOL	Avoidance	C,T	2DM,DT,W	20 aircraft
Pharpatara et al. (2017)	RRT	Generic	D	3DM,SO	1 km flight
Raghunathan et al. (2004)	DC,IPOPT	Avoidance	M	3DM	8 aircraft
Richards & How (2002)	MILP	Avoidance	T	2DM,SO,DO	3 aircraft
Ruiz & Soler (2015)	HS,IPOPT	Conflict Analysis	T	3DM,DT	4000 trajectories
Shippey & Subbarao (2008)	DC,GA	Generic	T	3DM	300 m flight
Urmson & Simmons (2003)	RRT	Generic	D	2DM,SO	-
Wuerl et al. (2003)	GA	Low-thrust	F	-	-
Yan et al. (2013)	PRM,A*	Generic	D	3DM,SO	40 m flight
Yokoyama & Suzuki (2005)	DC,GA	Generic	CR	-	-
Zhou (2013)	PS	Generic	D,F,T	3DM,WP,VR	3 km flight
Zhu et al. (2016)	GSM,SNOPT	Soaring	D	3DM,W	110 m flight

Table B.2: Symbol Glossary.

Solution		Objectives		Features	
Symbol	Description	Symbol	Description	Symbol	Description
A*	A* Heuristic	A	Altitude Gain	DB	Dubins Curves
DC	Direct Collocation	C	Cost	DO	Dynamic Obstacles
DL	D*-Lite	CR	Conflict Resolution	DS	Discretised Space
GA	Genetic Algorithm	D	Distance Travelled	DT	Discretised Time
GSM	Gauss Pseudo-Spectral Method	E	Energy	GL	Gliding Kinematics
HS	Hermite-Simpson Collocation	F	Fuel	RH	Receding Horizon
IPOPT	Interior Point Optimisation	M	Manouvering	RO	Robustness
MIP	Mixed Integer Programming	T	Mission Time	SO	Static Obstacles
NN	Neural Networks	-	-	VR	Voltage Reduction
PRM	Probabilistic Road Maps	-	-	W	Wind
PS	Other Pseudo-Spectral Method	-	-	WP	Waypoints
PSO	Particle Swarm Optimisation	-	-	2/3DM	2/3 Dimensional Model
Solution					
RKT	Receding Kinematic Trees	RRT	Rapidly-Growing Random Trees	SNOPT	Sparse Nonlinear Optimisation

Appendix C

Optimal Relief Appendix

C.1 Flight Optimal Control Problem

Based on the flight dynamics derived in Section 4.1 the resulting Optimal Control Problem in three dimensions is described as follows:

Objective:

$$\text{Minimise } 1 - SOC_v(t_f) \quad \forall v \in \mathcal{V} \quad (\text{C.1})$$

Subject to:

$$dx_v/dt = U_v(t) \cos(\Psi_v(t)) \cos(\gamma_v(t)) \quad \forall v \in \mathcal{V} \quad (\text{C.2})$$

$$dy_v/dt = U_v(t) \sin(\Psi_v(t)) \cos(\gamma_v(t)) \quad \forall v \in \mathcal{V} \quad (\text{C.3})$$

$$dz_v/dt = U_v \sin(\gamma_v(t)) \quad \forall v \in \mathcal{V} \quad (\text{C.4})$$

$$dU_v/dt = (T_v(t) - D_v(t) + L_v(t) \sin(\gamma_v(t)))/m_v \quad \forall v \in \mathcal{V} \quad (\text{C.5})$$

$$D_v(t) = qSC_{D_v}(t) \quad \forall v \in \mathcal{V} \quad (\text{C.6})$$

$$C_{D_v} = C_{D0} + KC_{L_v}^2 \quad \forall v \in \mathcal{V} \quad (\text{C.7})$$

$$C_{L_v} = L_v(t) + KC_{L_v}^2 \quad \forall v \in \mathcal{V} \quad (\text{C.8})$$

$$L_v = m_v g / \cos(\gamma_v(t)) \quad \forall v \in \mathcal{V} \quad (\text{C.9})$$

$$dSOC_v/dt = -T_v(t)U_v(t)/\mu BV \quad \forall v \in \mathcal{V} \quad (\text{C.10})$$

$$|\gamma_v| < \frac{\pi}{2}, \quad 0 \leq T_v \leq T_{max}, \quad U_v = U_{cr}, \quad SOC_v \leq 1 \quad \forall v \in \mathcal{V} \quad (\text{C.11})$$

$$f(x, y) + z_{min} \leq z_v \leq f(x, y) + z_{max}$$

The following boundary conditions are assumed:

$$\text{Initial: } x_v(t_0) = x_{0_v}, \quad y_v(t_0) = y_{0_v}, \quad z_v(t_0) = z_{0_v}, \quad SOC_v(t_0) = 1, \quad \gamma_v(t_0) = \gamma_{0_v} \quad \forall v \in \mathcal{V} \quad (\text{C.12})$$

$$\text{Final: } x_v(t_f) = x_{fv}, \quad y_v(t_f) = y_{fv}, \quad z_v(t_f) = z_{fv}, \quad SOC_v(t_f) \text{ free}, \quad \gamma_v(t_f) \text{ free} \quad \forall v \in \mathcal{V} \quad (\text{C.13})$$

C.2 Route Assignment Algorithm

We present the pseudocode describing the route assignment process in Section 5.3.3:

Algorithm 8 Route Continuity

```

1: function ROUTECONTINUITY
2:   Input: vehicle flows  $x_{ij}^*$ , hub set  $H$ , network  $N$ 
3:   Output: routes  $N_r$ 
4:   for  $i$  in  $H$  do
5:      $N_r = \sum_j x_{ij}^*$ 
6:   end for
7:   for  $i$  in  $N$  do
8:     for  $j$  in  $N$  do
9:       imbalanced flow  $x_{bij} = x_{ij}^* - x_{ji}^*$ 
10:      if  $x_{bij} < 0$  then
11:         $x_{bij} = 0$ 
12:      end if
13:    end for
14:  end for
15:  imbalanced routes  $N_{rb} = \text{CreateRoutes}(x_{bij}, H, x_{bij})$ 
16:  balanced routes  $N_{ra} = \text{CreateRoutes}(x_{bij} - x_{bij}, H, x_{bij})$ 
17:  return  $N_{ra}, N_{rb}$ 
18: end function

```

C.3 Detailed Results

Table C.1: Percentage battery depletion with no payload.

Nodes	1	2	3	4	5	6	7	8	9	10	11	12	13
1	-	0.11	0.19	0.35	0.29	0.26	0.27	0.10	0.28	0.31	0.13	0.58	0.55
2	0.11	-	0.23	0.29	0.39	0.17	0.36	0.21	0.30	0.29	0.13	0.50	0.58
3	0.18	0.23	-	0.31	0.20	0.28	0.13	0.14	0.11	0.18	0.11	0.56	0.37
4	0.35	0.29	0.31	-	0.51	0.14	0.42	0.40	0.27	0.17	0.25	0.23	0.46
5	0.28	0.39	0.19	0.50	-	0.45	0.11	0.18	0.25	0.36	0.28	0.69	0.40
6	0.25	0.18	0.25	0.12	0.50	-	0.39	0.32	0.28	0.20	0.17	0.33	0.51
7	0.27	0.35	0.13	0.41	0.11	0.38	-	0.20	0.16	0.26	0.24	0.60	0.32
8	0.12	0.20	0.14	0.42	0.19	0.32	0.22	-	0.24	0.31	0.15	0.61	0.49
9	0.28	0.31	0.11	0.28	0.25	0.27	0.16	0.24	-	0.11	0.17	0.49	0.28
10	0.31	0.30	0.18	0.17	0.36	0.20	0.26	0.31	0.11	-	0.19	0.36	0.31
11	0.12	0.15	0.11	0.25	0.28	0.17	0.23	0.15	0.17	0.19	-	0.49	0.45
12	0.57	0.50	0.51	0.25	0.80	0.33	0.59	0.61	0.48	0.33	0.47	-	0.52
13	0.55	0.59	0.42	0.47	0.41	0.58	0.32	0.49	0.28	0.31	0.45	0.52	-

Table C.2: Percentage battery depletion with 1kg payload.

Nodes	1	2	3	4	5	6	7	8	9	10	11	12	13
1	-	0.12	0.21	0.40	0.32	0.29	0.31	0.12	0.32	0.35	0.14	0.65	0.62
2	0.12	-	0.26	0.32	0.43	0.20	0.39	0.23	0.36	0.33	0.17	0.57	0.68
3	0.20	0.26	-	0.35	0.22	0.30	0.14	0.15	0.13	0.21	0.12	0.57	0.42
4	0.40	0.33	0.35	-	0.57	0.15	0.46	0.47	0.30	0.19	0.28	0.25	0.52
5	0.32	0.46	0.23	0.56	-	0.51	0.12	0.21	0.29	0.41	0.32	0.78	0.46
6	0.29	0.19	0.29	0.13	0.52	-	0.44	0.36	0.33	0.22	0.20	0.39	0.58
7	0.31	0.39	0.14	0.46	0.12	0.43	-	0.22	0.19	0.29	0.26	0.67	0.36
8	0.12	0.23	0.15	0.51	0.24	0.38	0.22	-	0.31	0.38	0.18	0.69	0.55
9	0.31	0.34	0.13	0.30	0.29	0.30	0.17	0.28	-	0.12	0.20	0.49	0.35
10	0.35	0.33	0.21	0.19	0.40	0.22	0.29	0.35	0.13	-	0.21	0.37	0.35
11	0.14	0.15	0.12	0.28	0.32	0.19	0.26	0.17	0.20	0.21	-	0.52	0.51
12	0.65	0.63	0.57	0.26	0.84	0.38	0.67	0.74	0.50	0.37	0.53	-	0.59
13	0.64	0.67	0.42	0.57	0.46	0.58	0.36	0.55	0.31	0.35	0.51	0.59	-

Algorithm 9 Create Routes

```
1: function CREATEROUTES
2:   Input: vehicle flows  $x_{ij}$ , hub set  $H$ , imbalanced flows  $x_{b_{ij}}$ 
3:   Output: routes  $N_r$ 
4:   for route  $r_v$  in route set  $N_r$  do
5:     origin  $i = \text{SelectHub}(H)$ 
6:     destinations  $d = \text{FindAll}(j, x_{ij} > 0)$ 
7:     destination  $j = \text{Random}(d)$ 
8:      $r_v = i, j$ 
9:      $x_{ij} = x_{ij} - 1$ 
10:    if  $x_{ij} = x_{b_{ij}}$  then
11:      while  $j \neq \text{hub}$  do
12:         $d = \text{FindAll}(k, x_{jk} > 0)$ 
13:        if  $\text{IsAnyConnectedToHub}(d)$  then
14:          destinations  $d_h = \text{FindConnectedToHub}(d)$ 
15:           $k = \text{Random}(d_h)$ 
16:        else
17:           $k = \text{Random}(d)$ 
18:        end if
19:         $x_{jk} = x_{jk} - 1$ 
20:         $r_v = \text{AddToRoute}(r_v, k)$ 
21:         $j = k$ 
22:      end while
23:    else
24:      while  $j \neq i$  do
25:         $d = \text{FindAll}(k, x_{jk} > 0)$ 
26:        if  $\text{IsAnyConnectedTo}(d, i)$  then
27:          destinations  $d_h = \text{FindConnectedToHub}(d)$ 
28:           $k = \text{Random}(d_h)$ 
29:        else
30:           $k = \text{Random}(d)$ 
31:        end if
32:         $x_{jk} = x_{jk} - 1$ 
33:         $r_v = \text{AddToRoute}(r_v, k)$ 
34:         $j = k$ 
35:      end while
36:    end if
37:  end for
38:  return  $N_r$ 
39: end function
```

Algorithm 10 Cargo Assignment

```

1: function CARGOASSIGNMENT
2:   Input: routes  $N_r$ , flows  $c_{ij}^*$ , demand  $A_i$ , payload capacity  $Q$ 
3:   Output: cargo assignments  $N_{rc}$ 
4:   for node  $i$  in  $A$  and node  $j$  in  $A$  do
5:     cargo transported  $S_i = S_i - c_{ij}^*$ 
6:      $S_j = S_j - c_{ij}^*$ 
7:   end for
8:   for  $r_v$  in  $N_r$  and node  $n$  in  $r_v$  do
9:     cargo assigned  $r_c(n) = \min(c_{ij}^*, Q, A_i)$ 
10:     $N_{rc} = r_c$ 
11:  end for
12:  for  $r_v$  in  $N_r$  and node  $n$  in  $r_v$  do
13:    if  $A_i \neq \text{Sum}(r_c(r_v(n) = i))$  then
14:       $N_{rc} = \text{ReassignCargo}(N_{rc}, c_{ij}^*, S_i, Q)$ 
15:    end if
16:  end for
17:  return  $N_{rc}$ 
18: end function

```

Algorithm 11 Cargo Reassignment

```

1: function REASSIGNCARGO
2:   Input: routes  $N_r$ , flows  $c_{ij}^*$ , demand  $S_i$ , payload capacity  $Q$ 
3:   Output: cargo assignments  $N_{rc}$ 
4:   for node  $i$  in  $S$  do
5:     while  $S_i > \text{Sum}(N_{rc}(n) = i)$  do
6:        $n = \text{Find}(N_{rc} < Q)$ 
7:        $N_{rc}(n) + = 1$ 
8:     end while
9:     while  $S_i < \text{Sum}(N_{rc}(n) = i)$  do
10:       $n = \text{Find}(N_{rc} > 0)$ 
11:       $N_{rc}(n) - = 1$ 
12:    end while
13:  end for
14:  return  $N_{rc}$ 
15: end function

```

Algorithm 12 Energy Monitoring

```

1: function ENERGYMONITORING
2:   Input: routes  $N_r$ , battery consumption  $e_{ij}$ , hub set  $H$ 
3:   Output: feasible routes  $N_b$ 
4:   for vehicle route  $r_v$  in  $N_r$  do
5:     battery in route  $r_b = 0$ 
6:     for node  $n$  in  $r_v$  do
7:       origin  $i = r_v(n)$ 
8:       destination  $j = r_v(n + 1)$ 
9:       battery level  $b = r(n - 1) + e_{ij}(r_c(n))$ 
10:       $r_b = \text{AddToRoute}(r_b, b)$ 
11:    end for
12:     $N_{rb} \leftarrow r_b$ 
13:  end for
14:  for  $r_b$  in  $N_{rb}$  and  $n$  in  $r_b$  do
15:    while  $r_b(n) > 1$  do
16:       $i = r_v(n - 1)$ 
17:       $j = r_v(n)$ 
18:      hub  $h = \text{ClosesHub}(i, j, H)$ 
19:      cargo  $c = r_c(n - 2)$ 
20:      battery to hub  $b_{h-} = r_b(n - 1) + e_{ih}(c)$ 
21:      battery after hub  $b_{h+} = r_b(\text{end}) + e_{hj}(c) - b_{h-}$ 
22:      if  $b_{h-} < 1$  and  $b_{h+} < 1$  then
23:         $r_v = \text{InsertAt}(h, n - 1)$ 
24:         $r_c = \text{InsertAt}(c, n - 2)$ 
25:         $r_b = \text{UpdateBattery}(r_v, r_c)$ 
26:      end if
27:      if  $n < 0$  then
28:        penalty  $p = M$ 
29:        break
30:      end if
31:       $n = n - 1$ 
32:    end while
33:  end for
34:  return  $N_b$ 
35: end function

```

Algorithm 13 Vehicle Assignment

```

1: function VEHICLEASSIGNMENT
2:   Input: feasible routes  $N_b$ , time matrix  $T_{ij}$ , drones  $D$ 
3:   Output: route for each UAV set  $N_d$ 
4:   for route  $r$  in  $N_b$  do
5:     route flight time  $t_r = 0$ 
6:     for  $n$  in  $r$  do
7:        $t_r += t_{n,n+1}$ 
8:     end for
9:     route flight time set  $N_t \leftarrow t_r$ 
10:  end for
11:  drone mission duration set  $T_d = \text{InitialiseAtZero}(D)$ 
12:  while  $\text{IsEmpty}(N_b)$  do
13:    drone route  $r_d = \text{FindMin}(T_d)$ 
14:    current hub  $i = r_d(\text{end})$ 
15:    for  $t_r$  in  $N_t$  do
16:      relocation hub  $j = r(\text{start})$ 
17:      relocation time  $t_L = t_{ij}$ 
18:       $t_r += t_L$ 
19:    end for
20:     $t_{\min} = \min(t_r)$ 
21:     $r_d \leftarrow t_r$ 
22:    drone mission duration  $t_d += t_r$ 
23:  end while
24:   $N_d \leftarrow r_d$ 
25:  return  $N_d$ 
26: end function
    
```

Table C.3: Percentage battery depletion with 2kg payload.

Nodes	1	2	3	4	5	6	7	8	9	10	11	12	13
1	-	0.14	0.23	0.46	0.36	0.34	0.35	0.13	0.35	0.39	0.16	0.74	0.70
2	0.15	-	0.30	0.37	0.50	0.23	0.45	0.27	0.42	0.43	0.25	0.65	0.74
3	0.23	0.29	-	0.39	0.25	0.33	0.16	0.20	0.14	0.23	0.14	0.65	0.48
4	0.45	0.37	0.40	-	0.64	0.15	0.54	0.51	0.34	0.21	0.32	0.28	0.60
5	0.37	0.48	0.25	0.64	-	0.59	0.16	0.23	0.33	0.46	0.36	0.88	0.52
6	0.33	0.22	0.33	0.15	0.62	-	0.50	0.41	0.34	0.25	0.22	0.44	0.65
7	0.35	0.46	0.16	0.53	0.15	0.53	-	0.24	0.20	0.33	0.29	0.85	0.41
8	0.13	0.26	0.17	0.52	0.26	0.41	0.25	-	0.31	0.41	0.20	0.79	0.64
9	0.36	0.39	0.14	0.34	0.33	0.34	0.20	0.31	-	0.14	0.23	0.58	0.36
10	0.40	0.38	0.23	0.21	0.46	0.25	0.33	0.39	0.14	-	0.24	0.43	0.40
11	0.16	0.17	0.13	0.32	0.36	0.22	0.30	0.19	0.22	0.24	-	0.60	0.57
12	0.74	0.70	0.65	0.32	0.88	0.47	0.75	0.78	0.56	0.43	0.60	-	0.67
13	0.70	0.74	0.47	0.59	0.52	0.65	0.45	0.64	0.36	0.40	0.60	0.67	-

Algorithm 14 Battery Assignment

```

1: function BATTERYASSIGNMENT
2:   Input: battery consumption  $e_{ij}$ , time matrix  $T_{ij}$ , drones  $D$ , hub set  $H$ 
3:   Output: battery assignment  $B$ 
4:    $t = 0$ 
5:   drone route step  $d_s = 1$ 
6:   while MissionIsNotFinished( $D$ ) do
7:     for drone  $d$  in  $D$  do
8:       if current drone time step  $d_t \leq t$  then
9:         drone departing  $d_{dep} = True$ 
10:        current origin  $d_o = d_{route}(d_s)$ 
11:       end if
12:     end for
13:      $B = \text{BatteriesToNodes}(D, H, B)$ 
14:     for  $d$  in  $D$  do
15:       if  $d_{dep} = True$  then
16:         current destination  $d_d = d_{route}(d_s + 1)$ 
17:         travel time  $d_{tv} = T_{d_o, d_d}$ 
18:         expected battery  $d_b = e_{d_o, d_d}$ 
19:       end if
20:     end for
21:      $B = \text{AssignBatteriesToVehicles}(D, H, B, e_{ij})$ 
22:     for  $d$  in  $D$  do
23:       if  $d_{dep} = True$  then
24:          $d_t = d_t + d_{tv}$ 
25:          $d_s = d_s + 1$ 
26:       end if
27:     end for
28:      $B = \text{ChargeBatteries}(H, B)$ 
29:   end while
30:    $t = t + 1$ 
31:   return  $B$ 
32: end function

```

Algorithm 15 Battery Assignment Operators

```

1: function BATTERYTONODES
2:   Input: hub set  $H$ , drones  $D$ , battery assignment  $B$ 
3:   Output: battery assignment  $B$ 
4:   for drone  $d$  in  $D$  do
5:     if MissionIsFinished() and drone battery  $d_b$  is not empty then
6:        $B \leftarrow \text{AddBatteryToNode}(d_b, \text{destination } d_d)$ 
7:     end if
8:     if drone departing  $d_{dep} = \text{True}$  and  $d_b$  is not empty and  $d_{step} > 0$  then
9:        $B \leftarrow \text{AddBatteryToNode}(d_b, d_d)$ 
10:       $B \leftarrow \text{RemoveBattery}(d_b)$ 
11:    end if
12:  end for
13:  return  $B$ 
14: end function
15: function BATTERYTOVEHICLES
16:   Input: hub set  $H$ , drones  $D$ , battery assignment  $B$ , battery consumption  $e_{ij}$ 
17:   Output: battery assignment  $B$ 
18:   for drone  $d$  in  $D$  do
19:     if drone departing  $d_{dep} = \text{True}$  and drone battery  $d_b$  is not empty then
20:       for  $b$  in  $B$  do
21:         if battery charge  $b_c \leq e_{d_o, d_d}$  then
22:            $d \leftarrow \text{AddBatteryToNode}(b, \text{destination } d_d)$ 
23:            $b_c - = e_{d_o, d_d}$ 
24:            $d \leftarrow \text{RemoveBattery}(d_b)$ 
25:           break
26:         end if
27:       end for
28:     else
29:        $d \leftarrow \text{AddBattery}(\text{new battery } b)$ 
30:        $b_c - = e_{d_o, d_d}$ 
31:     end if
32:   end for
33:   return  $B$ 
34: end function
35: function CHARGEBATTERIES
36:   Input: hub set  $H$ , battery assignment  $B$ 
37:   Output: battery assignment  $B$ 
38:   for  $b$  in  $B$  do
39:     if battery location  $b_{loc}$  is hub  $H$  then
40:        $b_{charge} + = \text{chargeRate}$ 
41:     end if
42:   end for
43:   return  $B$ 
44: end function

```

Table C.4: Percentage battery depletion with 3kg payload.

Nodes	1	2	3	4	5	6	7	8	9	10	11	12	13
1	-	0.16	0.32	0.58	0.42	0.37	0.41	0.15	0.40	0.45	0.23	0.89	0.80
2	0.16	-	0.33	0.43	0.55	0.25	0.51	0.37	0.45	0.43	0.30	0.74	0.84
3	0.26	0.33	-	0.45	0.28	0.38	0.18	0.22	0.19	0.29	0.15	0.74	0.54
4	0.52	0.42	0.45	-	0.74	0.19	0.81	0.58	0.39	0.24	0.38	0.32	0.67
5	0.41	0.61	0.28	0.73	-	0.66	0.19	0.32	0.37	0.53	0.50	1.01	0.59
6	0.40	0.25	0.47	0.17	0.74	-	0.56	0.49	0.39	0.29	0.25	0.49	0.76
7	0.41	0.52	0.21	0.62	0.15	0.57	-	0.33	0.23	0.38	0.33	0.86	0.46
8	0.15	0.30	0.20	0.64	0.31	0.47	0.28	-	0.35	0.45	0.22	0.89	0.72
9	0.44	0.45	0.16	0.47	0.37	0.39	0.22	0.37	-	0.15	0.26	0.64	0.40
10	0.45	0.43	0.26	0.24	0.54	0.29	0.38	0.45	0.16	-	0.27	0.53	0.45
11	0.20	0.22	0.15	0.36	0.41	0.25	0.34	0.23	0.25	0.27	-	0.75	0.66
12	0.84	0.74	0.74	0.34	1	0.48	0.86	0.91	0.64	0.49	0.68	-	0.90
13	0.80	0.84	0.54	0.70	0.64	0.74	0.46	0.72	0.40	0.46	0.89	0.76	-

Table C.5: Parameter Tuning – fitness deviation examines the relative difference in optimal solution with respect to $c = 0.9999$ iteration.

Parameters								Results	
N	a	c	α_r	β_r	α_d	β_d		Fitness Deviation (%)	Runtime (min)
100	0.01	0.9999	0.2	0.6	0.2	0.6		0.00%	4600
100	0.001	0.9995	0.2	0.6	0.2	0.6		0.00%	1,089
100	0.01	0.9995	0.1	0.55	0.2	0.6		0.00%	938
100	0.01	0.9995	0.1	0.7	0.2	0.6		0.00%	1086
100	0.01	0.9995	0.2	0.6	0.1	0.55		0.00%	927
100	0.01	0.9995	0.2	0.6	0.2	0.6		0.00%	956
150	0.01	0.9995	0.2	0.6	0.2	0.6		0.00%	1012
100	0.01	0.9995	0.2	0.6	0.2	0.8		0.50%	927
100	0.01	0.9995	0.2	0.8	0.2	0.6		0.90%	925
100	0.01	0.9995	0.1	0.4	0.2	0.6		1.00%	925
100	0.01	0.9995	0.2	0.6	0.1	0.4		1.00%	984
100	0.01	0.9995	0.2	0.6	0.1	0.7		1.00%	982
100	0.01	0.9995	0.2	0.6	0.4	0.8		1.20%	886
100	0.01	0.9995	0.2	0.4	0.2	0.6		14.30%	855
100	0.01	0.9995	0.2	0.6	0.2	0.4		42.10%	909
100	0.01	0.9995	0.2	0.6	0.4	0.65		49.00%	895
50	0.01	0.9995	0.2	0.6	0.2	0.6		49.60%	842
100	0.01	0.999	0.2	0.6	0.2	0.6		51.70%	666
100	0.1	0.9995	0.2	0.6	0.2	0.6		52.00%	768
100	0.01	0.9995	0.2	0.6	0.4	0.5		56.30%	1117
100	0.01	0.9995	0.4	0.5	0.2	0.6		56.60%	1115
100	0.01	0.9995	0.4	0.65	0.2	0.6		61.30%	894
100	0.01	0.995	0.2	0.6	0.2	0.6		67.90%	122
100	0.01	0.9995	0.4	0.8	0.2	0.6		67.90%	812
100	0.01	0.99	0.2	0.6	0.2	0.6		130.20%	48

Table C.6: Detailed relief distribution outputs per township (without balancing).

Township	Day 1		Day 2	
	Cargo Delivered (kg)	Demand Satisfaction (%)	Cargo Delivered (kg)	Demand Satisfaction (%)
40 UAVs				
Nanto City	-	-	-	-
Tsao Tun	2,373	100	2,373	100
Ji Jii	1,767	20	6,808	100
Pu Li	159	99	159	99
Chu Shan	0	0	0	0
Guo Hsin	0	0	240	42
Lu Gu	0	0	0	0
Ming Jian	1,125	99	1,126	100
Sui Li	288	100	288	100
Yu Chi	-	-	-	-
Chong Liao	3,888	99	3,892	100
Ren Ai	0	0	0	0
Hsin Yi	0	0	0	0
60 UAVs				
Nanto City	0	0	215	100
Tsao Tun	0	0	2,373	100
Ji Jii	8,655	99	8,655	99
Pu Li	159	99	159	99
Chu Shan	0	0	0	0
Guo Hsin	0	0	567	100
Lu Gu	0	0	1,266	99
Ming Jian	0	0	474	42
Sui Li	288	100	288	100
Yu Chi	-	-	-	-
Chong Liao	1,407	36	3,892	100
Ren Ai	0	0	0	0
Hsin Yi	0	0	0	0
90 UAVs				
Nanto City	0	0	215	100
Tsao Tun	549	23	2,373	100
Ji Jii	8,655	99	8,656	100
Pu Li	159	99	161	100
Chu Shan	0	0	342	100
Guo Hsin	567	100	567	100
Lu Gu	1266	99	1,267	100
Ming Jian	0	0	1,126	100
Sui Li	288	100	288	100
Yu Chi	-	-	-	-
Chong Liao	3,891	99	3,892	100
Ren Ai	0	0	298	100
Hsin Yi	0	0	314	100

Table C.7: Detailed relief distribution outputs per township (with balancing).

Township	Day 1		Day 2	
	Cargo Delivered (kg)	Demand Satisfaction (%)	Cargo Delivered (kg)	Demand Satisfaction (%)
40 UAVs				
Nanto City	-	-	-	-
Tsao Tun	756	32	1,437	61
Ji Jii	2,800	32	5,356	62
Pu Li	50	31	101	63
Chu Shan	111	32	204	60
Guo Hsin	192	33	336	59
Lu Gu	304	33	571	62
Ming Jian	358	31	688	61
Sui Li	90	31	174	60
Yu Chi	-	-	-	-
Chong Liao	1,276	32	2,404	62
Ren Ai	97	32	178	60
Hsin Yi	95	30	191	61
60 UAVs				
Nanto City	95	44	197	92
Tsao Tun	1,092	46	2,190	92
Ji Jii	4,051	47	8,050	93
Pu Li	77	48	146	91
Chu Shan	150	44	306	89
Guo Hsin	267	47	534	94
Lu Gu	424	33	856	68
Ming Jian	526	47	1,024	91
Sui Li	129	45	267	93
Yu Chi	-	-	-	-
Chong Liao	1,798	46	3,610	93
Ren Ai	139	47	268	90
Hsin Yi	146	46	293	93
90 UAVs				
Nanto City	146	68	215	100
Tsao Tun	1,629	69	2,373	100
Ji Jii	5,929	68	8,656	100
Pu Li	110	68	161	100
Chu Shan	219	64	342	100
Guo Hsin	384	68	567	100
Lu Gu	637	50	1,267	100
Ming Jian	760	67	1,126	100
Sui Li	201	70	288	100
Yu Chi	-	-	-	-
Chong Liao	2,659	68	3,892	100
Ren Ai	199	67	298	100
Hsin Yi	206	66	314	100

Appendix D

Trajectory Optimisation Appendix

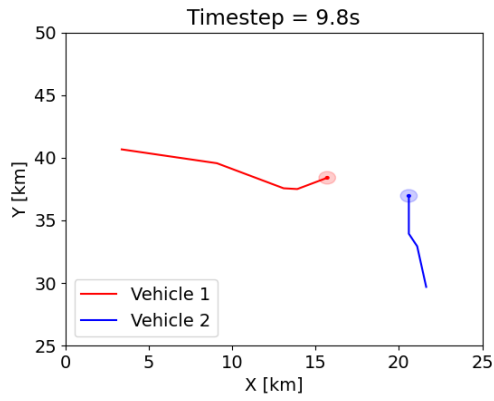
Table D.1: Particle Swarm Optimisation tuning results - averaged over 20 iterations.

Parameters			Crossings	Aircraft Count		Flight Time	Fitness
p	s	n	(-)	Avg (-)	Dev (-)	Deviation (min)	(-)
0.4	0.4	0.4	8673	1899	1512	2.984	22.5
0.4	0.4	0.6	9157	1953	1450	2.721	22.5
0.4	0.4	0.8	9820	2026	1406	2.532	23.1
0.4	0.4	1	9578	1999	1477	2.906	23.4
0.4	0.6	0.4	8419	1871	1569	3.303	23.1
0.4	0.6	0.6	8061	1831	1539	3.374	22.1
0.4	0.6	0.8	9378	1977	1581	3.185	24.1
0.4	0.6	1	9834	2028	1589	3.162	24.8
0.4	0.8	0.4	9314	1970	1526	2.987	23.7
0.4	0.8	0.6	8959	1931	1558	3.176	23.6
0.4	0.8	0.8	9357	1975	1536	3.082	23.8
0.4	0.8	1	10038	2051	1494	2.746	24.1
0.4	1.2	0.4	8889	1923	1447	2.819	22.2
0.4	1.2	0.6	7573	1777	1674	3.815	22.7
0.4	1.2	0.8	9361	1975	1543	3.014	23.8
0.4	1.2	1	8339	1862	1639	3.810	23.2
0.4	1.5	0.4	9536	1995	1473	2.867	23.4
0.4	1.5	0.6	9052	1941	1467	2.937	22.7
0.4	1.5	0.8	8829	1916	1569	3.241	23.3
0.4	1.5	1	9474	1988	1555	3.163	23.8
0.4	1	0.4	8921	1926	1485	2.935	22.7
0.4	1	0.6	8867	1920	1504	3.032	22.8
0.4	1	0.8	8445	1873	1525	3.310	22.3
0.4	1	1	7844	1807	1652	3.938	22.7
0.6	0.4	0.4	8915	1926	1426	2.745	22.1
0.6	0.4	0.6	9065	1942	1467	2.901	22.7
0.6	0.4	0.8	8897	1924	1568	3.247	23.6
0.6	0.4	1	10387	2089	1575	3.130	25.0

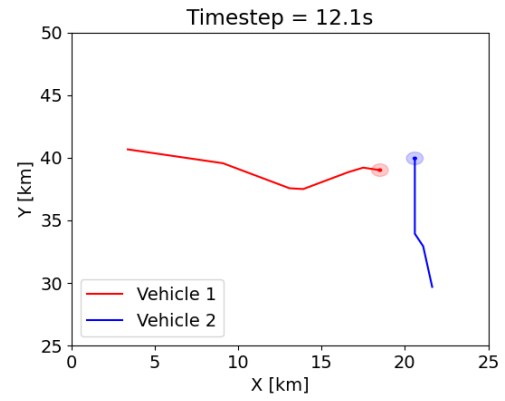
0.6	0.6	0.4	8547	1885	1550	3.210	22.6
0.6	0.6	0.6	8759	1908	1545	3.143	23.0
0.6	0.6	0.8	10228	2072	1381	2.479	23.3
0.6	0.6	1	9316	1970	1558	3.289	23.7
0.6	0.8	0.4	9599	2002	1451	2.733	23.3
0.6	0.8	0.6	8725	1905	1488	3.067	21.9
0.6	0.8	0.8	9458	1986	1585	3.124	24.2
0.6	0.8	1	10172	2065	1462	2.725	23.7
0.6	1.2	0.4	7775	1799	1672	3.653	23.1
0.6	1.2	0.6	8270	1854	1556	3.398	22.5
0.6	1.2	0.8	9788	2023	1503	2.912	23.7
0.6	1.2	1	9426	1982	1495	3.035	23.3
0.6	1.5	0.4	9870	2032	1403	2.606	23.2
0.6	1.5	0.6	10032	2050	1420	2.648	23.3
0.6	1.5	0.8	9002	1935	1447	2.890	22.4
0.6	1.5	1	9549	1996	1481	2.810	23.4
0.6	1	0.4	8404	1869	1486	3.269	21.7
0.6	1	0.6	8797	1913	1486	3.091	22.0
0.6	1	0.8	8840	1917	1523	3.211	23.0
0.6	1	1	9658	2008	1498	3.025	23.3
0.8	0.4	0.4	8804	1913	1543	3.137	23.0
0.8	0.4	0.6	8361	1864	1540	3.320	22.3
0.8	0.4	0.8	9519	1993	1508	3.011	23.6
0.8	0.4	1	9063	1942	1566	3.470	23.3
0.8	0.6	0.4	9251	1963	1438	2.845	22.7
0.8	0.6	0.6	10013	2048	1422	2.648	23.5
0.8	0.6	0.8	9025	1938	1543	3.113	23.3
0.8	0.6	1	9629	2005	1630	3.456	24.8
0.8	0.8	0.4	9125	1949	1444	2.861	22.6
0.8	0.8	0.6	8254	1852	1553	3.421	22.5
0.8	0.8	0.8	9376	1977	1527	3.059	23.7
0.8	0.8	1	9708	2014	1590	3.041	24.5
0.8	1.2	0.4	8343	1862	1514	3.316	22.1
0.8	1.2	0.6	8918	1926	1521	3.126	23.1
0.8	1.2	0.8	8921	1926	1481	3.079	22.8
0.8	1.2	1	9360	1975	1496	3.080	23.3
0.8	1.5	0.4	8467	1876	1456	3.052	21.7
0.8	1.5	0.6	8603	1891	1539	3.164	22.9
0.8	1.5	0.8	9145	1951	1421	2.858	22.2
0.8	1.5	1	9562	1998	1487	2.873	23.4
0.8	1	0.4	8448	1874	1547	3.172	22.8
0.8	1	0.6	8527	1883	1518	3.134	22.6
0.8	1	0.8	10315	2081	1476	2.740	24.2
0.8	1	1	10491	2101	1500	2.750	24.6
1.2	0.4	0.4	8489	1878	1554	3.257	23.0
1.2	0.4	0.6	9965	2042	1493	2.798	24.0
1.2	0.4	0.8	9044	1940	1542	3.133	23.3
1.2	0.4	1	9885	2034	1493	2.910	23.9

1.2	0.6	0.4	8859	1920	1502	3.230	22.5
1.2	0.6	0.6	9445	1985	1508	2.943	23.5
1.2	0.6	0.8	9523	1993	1397	2.632	22.5
1.2	0.6	1	8978	1933	1596	3.354	23.7
1.2	0.8	0.4	9063	1942	1456	2.834	22.6
1.2	0.8	0.6	9238	1962	1537	3.100	23.5
1.2	0.8	0.8	9337	1973	1487	2.941	23.4
1.2	0.8	1	10479	2099	1458	2.697	24.2
1.2	1.2	0.4	8469	1876	1514	3.116	22.4
1.2	1.2	0.6	8355	1863	1563	3.227	22.7
1.2	1.2	0.8	9438	1984	1480	2.892	23.0
1.2	1.2	1	9172	1954	1421	2.872	22.2
1.2	1.5	0.4	8471	1876	1517	3.085	22.4
1.2	1.5	0.6	9128	1949	1463	2.954	22.8
1.2	1.5	0.8	9630	2005	1422	2.750	22.8
1.2	1.5	1	10171	2065	1433	2.764	23.4
1.2	1	0.4	8951	1930	1483	2.961	22.4
1.2	1	0.6	8816	1915	1613	3.385	23.6
1.2	1	0.8	8403	1869	1580	3.557	22.5
1.2	1	1	9311	1970	1533	3.251	23.5
1.5	0.4	0.4	8906	1925	1496	3.076	22.8
1.5	0.4	0.6	9517	1993	1473	2.788	23.2
1.5	0.4	0.8	8531	1883	1558	3.287	22.8
1.5	0.4	1	10600	2113	1440	2.813	23.7
1.5	0.6	0.4	9617	2004	1333	2.420	22.2
1.5	0.6	0.6	9402	1980	1517	2.970	23.5
1.5	0.6	0.8	10714	2126	1412	2.422	24.1
1.5	0.6	1	9051	1941	1579	3.254	23.1
1.5	0.8	0.4	9011	1936	1411	2.818	22.0
1.5	0.8	0.6	8054	1830	1617	3.571	22.7
1.5	0.8	0.8	8675	1899	1549	3.399	22.6
1.5	0.8	1	10354	2086	1447	2.733	24.2
1.5	1.2	0.4	8918	1926	1511	3.033	23.0
1.5	1.2	0.6	9335	1972	1498	3.016	23.0
1.5	1.2	0.8	9773	2021	1541	3.311	23.8
1.5	1.2	1	9822	2027	1525	3.021	24.1
1.5	1.5	0.4	9402	1980	1390	2.551	22.2
1.5	1.5	0.6	8483	1878	1541	3.252	22.8
1.5	1.5	0.8	9880	2033	1458	2.879	23.3
1.5	1.5	1	9986	2045	1507	3.111	23.7
1.5	1	0.4	8842	1918	1505	3.022	22.9
1.5	1	0.6	9207	1958	1396	2.663	22.2
1.5	1	0.8	9764	2020	1509	3.064	23.7
1.5	1	1	10248	2074	1483	3.029	24.3
1	0.4	0.4	9188	1956	1431	2.819	22.7
1	0.4	0.6	9693	2012	1406	2.659	22.8
1	0.4	0.8	8678	1899	1593	3.435	23.6
1	0.4	1	9934	2039	1586	3.202	24.7

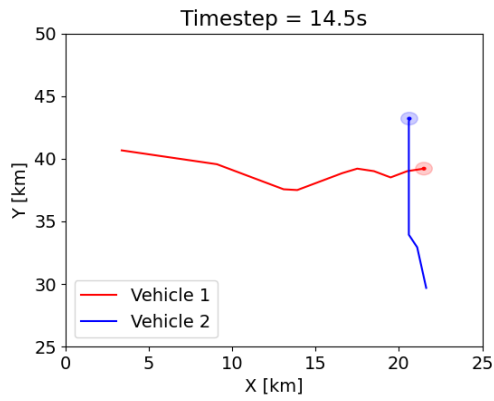
1	0.6	0.4	8519	1882	1552	3.166	23.0
1	0.6	0.6	9504	1991	1436	2.674	23.0
1	0.6	0.8	10097	2057	1443	2.662	23.6
1	0.6	1	9904	2036	1592	3.167	24.5
1	0.8	0.4	8213	1848	1571	3.302	22.7
1	0.8	0.6	8442	1873	1617	3.452	23.2
1	0.8	0.8	10006	2047	1526	3.059	24.2
1	0.8	1	10487	2100	1475	3.007	24.0
1	1.2	0.4	9208	1958	1372	2.539	21.9
1	1.2	0.6	9182	1955	1471	2.847	22.8
1	1.2	0.8	8861	1920	1558	3.247	23.1
1	1.2	1	10473	2099	1514	2.990	24.7
1	1.5	0.4	8969	1932	1437	2.749	22.1
1	1.5	0.6	9192	1956	1465	2.932	22.8
1	1.5	0.8	9782	2022	1488	2.749	23.6
1	1.5	1	8584	1889	1626	3.843	23.6
1	1	0.4	9430	1983	1404	2.776	22.5
1	1	0.6	8138	1839	1657	3.518	23.2
1	1	0.8	10631	2116	1400	2.690	23.9
1	1	1	8920	1926	1623	3.453	23.6



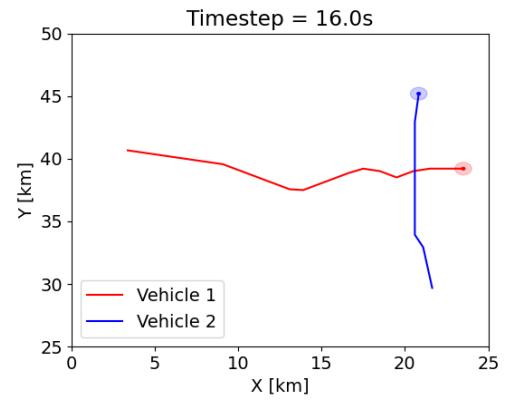
(a) Timestep = 9.8s



(b) Timestep = 12.1s

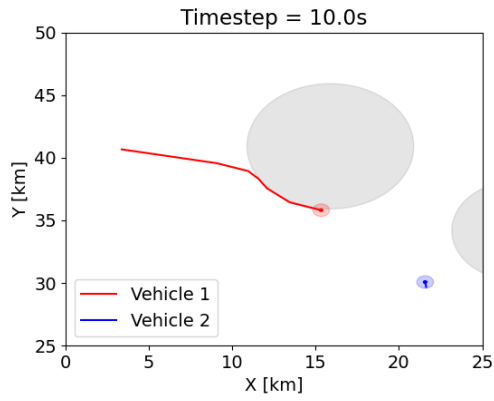


(c) Timestep = 14.5s

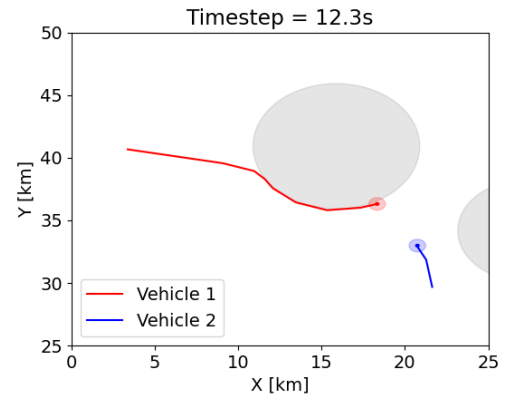


(d) Timestep = 16.0s

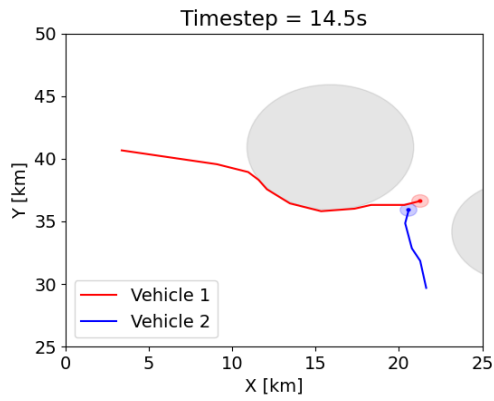
Figure D.1: Dynamic obstacle rapid tree simulation results



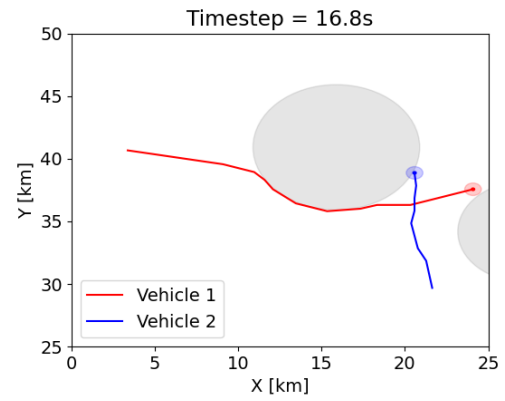
(a) Timestep = 10.0s



(b) Timestep = 12.3s



(c) Timestep = 14.5s



(d) Timestep = 16.8s

Figure D.2: Static and dynamic obstacle rapid tree simulation results

Appendix E

Endogenous Stochastic Research

Appendix

E.1 Formulation Linearisation

The proposed problem contains various non-linear relationships that prevent the use of linear solution methods. In this section, we introduce a series of auxiliary variables to linearise these relationships.

Variable $m_{ij}^{s,t}$ is introduced in equations (E.1.1-E.1.3) to ensure that the objective function (7.1.1) remains linear. The variable relates scenario costs to the explored node by calculating the product $u_{ij}^t E_\xi Z$.

$$Z = \sum_{t \in T} (Z_{uav}^t + Z_{exp}^t + \sum_{s \in S} p^s \sum_{ij \in E} \gamma_{ij}^s m_{ij}^{s,t}) \quad (\text{E.1.1})$$

$$m_{ij}^{s,t} \leq u_{ij}^t M \quad \forall ij \in E, \quad \forall t \in T, \quad \forall s \in S \quad (\text{E.1.2})$$

$$m_{ij}^{s,t} \leq Z_{veh}^{t,s} + Z_{dmg}^{t,s} + Z_{unc}^{t,s} + Z_{pen}^{t,s} \quad \forall ij \in E, \quad \forall t \in T, \quad \forall s \in S \quad (\text{E.1.3})$$

$$m_{ij}^{s,t} \geq Z_{veh}^{t,s} + Z_{dmg}^{t,s} + Z_{unc}^{t,s} + Z_{pen}^{t,s} - (1 - u_{ij}^t)M \quad \forall ij \in E, \quad \forall t \in T, \quad \forall s \in S \quad (\text{E.1.4})$$

$$m_{ij}^{s,t} \geq 0 \quad \forall ij \in E, \quad \forall t \in T, \quad \forall s \in S \quad (\text{E.1.5})$$

Further components of the objective function require modification. In particular, equations (7.1.6) and (7.1.7) rely on the non linear terms $u_{ij}^t y_{ijk}^{s,t}$ and $(1 - u_{ij}^t) y_{ijk}^{s,t}$ to calculate deterministic and uncertain damage costs, respectively. Variables $z_{ijk}^{s,t}$ and $v_{ijk}^{s,t}$ are introduced to capture the relationship, with the linearisation constraints presented in equation sets (E.2.1-E.2.5) and (E.3.1-E.3.5).

$$Z_{dmg}^{t,s} = \sum_{ij \in E} z_{ijk}^{s,t+1} d_{ij}^s c \quad \forall t \in T, \quad \forall s \in S \quad (\text{E.2.1})$$

$$z_{ijk}^{s,t+1} \leq y_{ijk}^{s,t+1} \quad \forall ij \in E, \quad \forall k \in K_v, \quad \forall t \in T, \quad \forall s \in S \quad (\text{E.2.2})$$

$$z_{ijk}^{s,t+1} \leq b_{ij}^{t+1} \quad \forall ij \in E, \quad \forall k \in K_v, \quad \forall t \in T, \quad \forall s \in S \quad (\text{E.2.3})$$

$$z_{ijk}^{s,t+1} \geq y_{ijk}^{s,t+1} + b_{ij}^{t+1} \quad \forall ij \in E, \quad \forall k \in K_v, \quad \forall t \in T, \quad \forall s \in S \quad (\text{E.2.4})$$

$$z_{ijk}^{s,t} = \{0, 1\} \quad \forall ij \in E, \quad \forall k \in K_v, \quad \forall t \in T, \quad \forall s \in S \quad (\text{E.2.5})$$

$$Z_{unc}^{t,s} = \sum_{ij \in E} y_{ijk}^{s,t+1} \hat{d}_{ij}^s c \quad \forall t \in T, \quad \forall s \in S \quad (\text{E.3.1})$$

$$v_{ijk}^{s,t+1} \leq y_{ijk}^{s,t+1} \quad \forall ij \in E, \quad \forall k \in K_v, \quad \forall t \in T, \quad \forall s \in S \quad (\text{E.3.2})$$

$$v_{ijk}^{s,t+1} \leq 1 - b_{ij}^{t+1} \quad \forall ij \in E, \quad \forall k \in K_v, \quad \forall t \in T, \quad \forall s \in S \quad (\text{E.3.3})$$

$$v_{ijk}^{s,t+1} \geq y_{ijk}^{s,t+1} - b_{ij}^{t+1} \quad \forall ij \in E, \quad \forall k \in K_v, \quad \forall t \in T, \quad \forall s \in S \quad (\text{E.3.4})$$

$$v_{ijk}^{s,t} = \{0, 1\} \quad \forall ij \in E, \quad \forall k \in K_v, \quad \forall t \in T, \quad \forall s \in S \quad (\text{E.3.5})$$

Finally, the battery relationship is linearised by the introduction of n_{ijk}^t in (E.4.1-E.4.6), which monitors battery level to the previous node visited. The constraint set presented below modifies the equations (7.2.6) and (7.2.7).

$$o_{jk}^t = \begin{cases} 0 & \text{if } j \in D \\ \sum_{i \in V} (x_{ijk}^t q_{ij} + n_{ijk}^t) & \text{if } j \notin D \end{cases} \quad \forall j \in V, \quad \forall t \in T, \quad \forall k \in K_d \quad (\text{E.4.1})$$

$$1 \geq \sum_{i \in V} (x_{ijk}^t q_{ij} + n_{ijk}^t) \quad \forall j \in V, \forall t \in T, \forall k \in K_d \quad (\text{E.4.2})$$

$$n_{ijk}^t \leq x_{ijk}^t \quad \forall i, j \in V, \forall t \in T, \forall k \in K_d \quad (\text{E.4.3})$$

$$n_{ijk}^t \leq o_{jk}^t \quad \forall i, j \in V, \forall t \in T, \forall k \in K_d \quad (\text{E.4.4})$$

$$n_{ijk}^t \geq o_{jk}^t - (1 - x_{ijk}^t) \quad \forall i, j \in V, \forall t \in T, \forall k \in K_d \quad (\text{E.4.5})$$

$$0 \leq n_{ijk}^t \leq 1 \quad \forall ij \in E, \forall k \in K_v, \forall t \in T \quad (\text{E.4.6})$$

E.2 Genetic Algorithm Tuning

Table E.1: Manual Genetic Algorithm Tuning - each experiment is repeated 1000 times.

Parameters			Results		
z	p_c	p_m	Travel Time [h]	Energy [KWh]	Runtime [h]
50	0.6	0.1	16.67	828.5	1.657
100	0.6	0.1	16.42	806.3	2.640
100	0.5	0.1	16.48	799.5	2.554
100	0.7	0.1	16.67	813.5	2.772
100	0.6	0.15	16.66	810.3	2.840
100	0.5	0.15	16.66	810.3	2.458
100	0.7	0.15	16.48	824.8	2.698
100	0.6	0.05	16.64	756.5	2.510
100	0.5	0.05	16.65	818.5	2.456
100	0.7	0.05	16.40	811.9	2.780
150	0.6	0.1	16.63	770.3	4.251
200	0.6	0.1	16.55	776.1	5.765